EFFECTS OF FIELD INDEPENDENCE, COMPUTER EXPERIENCE, AND DYNAMIC PICTORIAL ONLINE HELP PRESENTATIONS ON LEARNING APPLICATION FUNCTION IN A GRAPHICAL USER INTERFACE

By

JOHN GORDON TYLER

A DISSERTATION PRESENTED TO THE GRADUATE SCHOOL
OF THE UNIVERSITY OF FLORIDA IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

UNIVERSITY OF FLORIDA

1993

To my daughter Jessica Ann, for her sweet and joyful love.

May her desire to learn always brightly shine and guide her.

ACKNOWLEDGMENTS

This research project was the culmination of more than a decade of graduate studies and professional endeavors. During this time I received kind and expert guidance from many people, without whose gifts of time and talent this goal would never have been realized. First, I express my deep appreciation to Dr. Lee Mullally for serving as the chair of my supervisory committee, and for his guidance throughout this research project. His reassuring support and constructive criticism empowered me to achieve goals beyond my expectations.

I also wish to thank the other graduate faculty on my supervisory committee, Drs. Roy Bolduc, Doug Dankel, and Jeff Hurt, who provided insightful recommendations and thoughtful questioning during each phase of this research project. They were always willing to contribute their expertise to help refine the methods and materials applied in this study. I also want to thank Dr. James Algina for his assistance with the development and review of the statistical analyses for this study.

I want to express my gratitude to the management team at the IBM Personal Systems Programming Center in Boca Raton, Flordia. In particular I want to thank Mark Tempelmeyer, who actively promoted my return to the University of Florida. His constant optimism and wry sense of humor helped to inspire and motivate me during my long sabbatical from the IBM lab. In addition, I want to thank Dr. Frances Palacio, Oscar Fleckner, Marty Voss, Janis Walkow, and Tim Shortley for their support during my leave.

Thanks also go to my friends and colleagues at IBM who contributed to this project: Jack Reese, Jeff Baker, Ray Voigt, Larry Kyralla, Doug Bloch, Larry Mallett, Tom Greaves, Chris Freeman, Kerry Ortega, Carol Righi, Jim Lewis, and many others who supported or participated in this study. I especially want to thank Dr. Bob Kamper, for his thoughtful reviews of this manuscript and his spirited e-mail during this study.

I also thank professors Grandon Gill, Paul Hart, and Randy Coyner of the College of Business at Florida Atlantic University for consulting on this research and for providing access to students in their classes. Their support was essential and greatly appreciated.

Finally, I wish to express humble gratitude to my family, who continually gave encouragement and supported me in every conceivable manner. I want to thank my father, Dr. Leslie J. Tyler, whose career served as a wonderful example and who instilled in me the desire to become a research professional. I also thank my mother, Pat, who reminded me at times to "Stop and smell the roses", and to not be too deeply immersed in books and computers. Above everyone else, I'm deeply thankful for my wife, Jo, whose devotion and kindness overcame the many hardships. She gave me her strength and hope whenever my own were flagging.

TABLE OF CONTENTS

Page

Abstract of Dissertation Presented to the Graduate School
of the University of Florida in Partial Fulfillment of the
Requirements for the Degree of Doctor of Philosophy

EFFECTS OF FIELD INDEPENDENCE, COMPUTER EXPERIENCE,
AND DYNAMIC PICTORIAL ONLINE HELP PRESENTATIONS ON
LEARNING APPLICATION FUNCTION IN A GRAPHICAL USER INTERFACE

By
John Gordon Tyler

December 1993

Chairman: Lee J. Mullally
Major Department: Instruction and Curriculum

The graphical user interfaces of modern computer applications use dynamic pictorial elements to represent application functions. Online help messages assist users learning to operate those functions. Online help, however, rarely incorporates pictorial elements. Instructional message design and visual learning theories suggest that pictorially encoded messages should result in greater learning than purely verbal help messages. H. A. Witkin's theory of cognitive style suggests that learners with greater field independence will perform better in complex visual environments, such as those found in graphical user interfaces. Some researchers suggest that prior computer experience is the most important determinant of performance in an unfamiliar human-computer interface.

This study was conducted to examine the effects of individuals' level of field independence and prior computer experience on application task performance in a graphical user interface. This study also investigated Aptitude x Treatment interaction effects between computer users' cognitive style (field dependence-independence) or their level of computer experience and the use of dynamic pictorial message elements in online help messages.

From a population of university business college students, 38 volunteer subjects were randomly assigned to one of two online help treatments: text-only help or text-with-motion-video help. The two help treatments were identical except for the addition of digital motion video segments. Field independence was measured using the Group Embedded Figures Test. Computer experience was assessed using the Computer Experience and Competence Survey. Time in help was measured as the total time online help messages were displayed during training. These variables were applied as covariates in a multiple covariance analysis. Performance on computer application tasks was the dependent variable. Subjects completed an individualized computer-based training regimen including a pretest, twelve lessons covering system and spreadsheet application functions, and a posttest.

The results showed that subjects with higher field independence had significantly higher task performance scores than subjects with lower field independence. Also, subjects with more computer experience had significantly higher performance scores than those with less prior experience. No significant differences in performance on application tasks resulted from the addition of dynamic pictorial message elements in online help. The results of this study may contribute to the design of adaptive human-computer interfaces and online help systems.

CHAPTER 1
INTRODUCTION

The goal of this study was to investigate the effects of aptitude differences among individual computer users and the effects of dynamic pictorial message presentations in computer-based instruction (e.g., online help). Two distinct characteristics of computer users, cognitive style (measured as field independence) and computer experience, were examined to determine whether relationships exist between these characteristics, the presence of pictorial message content in online help, and performance on computer tasks. Specific research questions were raised to examine the effects of computer users' cognitive styles and levels of computer experience on their performance on application tasks in a graphical user interface (GUI). In addition, this study was designed to determine whether Aptitude x Treatment interaction (ATI) effects occurred between either cognitive style or prior computer experience and the presence of dynamic pictorial message elements in online help.

Statement of the Problem

The human-computer interface (HCI) provides an environment for the interaction between a human user and the dynamic operations of a computer. A graphical user interface, one class of HCI, uses a variety of pictorial message elements in addition to verbal elements (text) to represent computer states and functions (Horton, 1990; Shneiderman, 1983). A computer state, or processing condition, may be verbally described using text message elements, or it may be visually presented by encoding with static or dynamic pictorial elements.

Learning to recognize computer states and manipulate computer functions often requires learning from information displayed online, in the form of system or application help messages. Online help messages typically consist of text messages that explain a function, state, or procedure. Although computer states and functions are routinely represented in a GUI using dynamic, pictorial elements, online help messages rarely incorporate pictorial message elements.

Instructional message design principles suggest that the modality of message elements employed during instruction should be the same as the modality found in performance situations (Fleming & Levie, 1978). Instructional message design theory also suggests that the better a symbol system conveys the critical features of a concept or task, the more appropriate it is during instruction (Salomon, 1978). Reinforcing information presented verbally with appropriate visuals has been shown to result in significantly greater learning over presenting verbal messages alone (Fleming, 1987).

Research on the cognitive style dimension field dependence-independence has shown that subjects who tend to be more field-independent identify and distinguish objects in a complex visual field more readily than less field-independent subjects (Witkin, Moore, Goodenough, & Cox, 1977). Additional studies by Witkin et al. (1977) have shown that a learner's verbal abilities are only marginally correlated with his or her level of field independence. These findings suggest that when using an unfamiliar graphical interface, users with higher field independence should perform better than users with lower field independence. According to Witkin's theory, this would result from increased comprehension of the complex visual environment by individuals with higher field independence.

Comparative expertise research has shown that when introduced to new computer systems or applications, computer users differ widely in initial performance, depending primarily on the extent of their prior computer experience, and to a lesser degree on the difference between the type of interface encountered previously and the one being introduced (Whiteside, Jones, Levy, & Wixon, 1985). These findings suggest that with increased prior experience, where the potential for computer operation skill transfer exists, initial performance will be higher than without prior experience.

The problem addressed by this study was the lack of experimental evidence to demonstrate theoretically anticipated effects of field independence and computer experience on learning application functions in a GUI. In addition, there was a lack of evidence regarding what effects the use of dynamic pictorial message elements in online help information would have on learning application functions in a GUI.

Prior research has not provided clear answers to the following instructional design and human-computer interface design questions: Does either the user's level of field independence or level of computer experience moderate the effectiveness of specific message designs for online help? In particular, does a user's level of field independence interact with varying levels of pictorial content in online help to influence performance on tasks in a GUI? Would the extent of a user's prior computer experience interact with varying levels of pictorial content in online help to influence performance on application tasks? Would certain combinations of field independence and computer experience interact in unique ways with the presence of dynamic pictorial elements in online help to influence performance on tasks? This study was designed and conducted to answer these questions.

Need for the Study

Graphical User Interfaces and Visual Learning

Pictorial or graphical user interfaces have been rapidly replacing textual interfaces as the primary means for manipulating computer system and application functions. By representing state and function using nonverbal graphical elements (both static and dynamic), graphical user interfaces have achieved greater expressive power than text-based user interfaces (Horton, 1990). By improving the fidelity of representation for computer state and function, graphical user interfaces may improve computer users' ability to learn and control a wider range of functions than would be expected when text-based user interfaces are employed.

Online help systems display documentation for computer operations and application functions via video display devices. Online help messages consist primarily of textual (verbal-digital) message elements that verbally describe the help topic. Online help messages rarely incorporate pictorial (visual-iconic) elements that visually illustrate the help topic. This would be an appropriate design for help messages in a primarily textual user interface. Instructional message design principles suggest, however, that online help messages describing the use and manipulation of visual-iconic elements in a GUI should also contain visual-iconic elements that refer to and depict the computer operations and application functions. Specifically, Fleming and Levie (1978) stated: "In general, the modality used in the final testing or application situation should be the modality employed during instruction" (p. 106). Affirming the utility of this principle applied to the design of online help messages was one goal of this study.

As computer users interact with their systems using graphical, direct-manipulation techniques, learning computer functions shifts from a primarily verbal paradigm to an increasingly visual one. Strong verbal language skills cease to be the sole aptitude required for successful learning and competent task completion. Visual, nonverbal cognitive skills should take on added value and have greater influence on concept attainment. This expectation follows from Paivio's dual-coding theory and has been supported by research on the use of graphics in human-computer interfaces (Rieber, 1990; Rieber & Kini, 1991).

Two aspects of dual-coding theory are important when considering online help message design. First, verbal and visual cognitive mechanisms are interrelated; their learning effects are symbiotic. That is, information that is coded in both verbal and visual modes is remembered more readily and more accurately than information encoded in only one mode (Fleming, 1987). Dual-coding theory also suggests that visual stimuli are encoded more frequently in both modes than are verbal stimuli, thus strengthening the value of visual, nonverbal stimuli for instruction.

Visual learning theory differentiates between presentations using static and dynamic graphical presentations. Graphics have been shown to be effective attention-gaining devices (Rieber & Kini, 1991). When appropriately designed, graphics may enhance learning during computer-based instruction. Animated, or dynamic, graphical images are fundamentally different from static graphics. Animation in computer-based instruction involves rapidly updated computer screen displays, presenting an illusion of motion. Because computer states and functions are themselves dynamic, their pictorial or iconic representations in graphical user interfaces may be animated at appropriate times. Just as online help presentations should incorporate the pictorial elements found in graphical user interfaces, these elements should be appropriately animated to provide a more effective representation of the computer state and function (Rieber, 1990).

Individual Characteristics and Performance in HCI

Prior research has documented significant variability of performance when users are first introduced to unfamiliar computer systems and applications (Pocius, 1991; Whiteside et al., 1985). Although many potential sources exist for this variance, and while most sources have not been reliably measured, human-computer interaction research has become increasingly focused on identifying the factors involved. For example, HCI researchers have reported effects of general intelligence, prior computer experience, cognitive style, academic background, and age on computer-based learning outcomes (Pocius, 1991). Among the individual characteristics found to influence performance during computer-based instruction, two were selected for further examination in this study: (a) cognitive style and (b) computer experience.

Cognitive style. In a review of field dependence-independence research (Witkin et al., 1977), four essential characteristics of cognitive style were described. Cognitive style (a) refers to individual differences in how people perceive, solve problems, and learn; (b) is a pervasive dimension that influences one's personality, not only one's cognitive processes; (c) is stable over time, although it may be manipulated or altered over time; and (d) is a bipolar dimension, where "...each pole has adaptive value under specified circumstances, and so may be judged positively in relation to those circumstances" (p. 16).

Witkin's research has identified field dependence-independence as the cognitive style dimension most widely investigated and systematically applied to educational problems. His theory of cognitive style was not intended to narrowly categorize individuals as "field-dependent" or "field-independent." Rather, these terms have been used as convenient, if somewhat misleading, labels for extremes of performance on perception tests (e.g., the Group Embedded Figures Test). In defining field dependence-independence, Witkin et al. (1977) made the following assertion:

> Because scores from any test of field dependence-independence form a continuous distribution, these labels reflect a tendency, in varying degrees of strength, toward one mode of perception or the other. There is no implication that there exist two distinct types of human beings. (p. 7)

In describing individuals' cognitive styles, Witkin et al. consistently referred to relative, rather than absolute, characteristics. For example, "relatively field-independent persons have been found more likely to impose structure spontaneously on stimulus material which lacks it" (1977, p. 9), and "the relatively field-dependent person tends to adhere to the organization of the field as given" (p. 9). An individual's ability to perceive structure in a complex graphical computer application and the ability to correctly interact with that application could be expected to correlate positively with that individual's degree of field independence.

Computer experience. A user's prior computer experience can dramatically influence performance in a new, unfamiliar human-computer interface. In comparative expertise research, performance differences between novices and experts have been analyzed with regard to how they performed in problem solving situations (Lesgold, Gabrys, & Magone, 1990), or on other cognitive operations such as memory and perception (Aster & Clark, 1985). Whiteside et al. (1985) demonstrated consistent differences between expert and novice computer users' performance on tasks in familiar and unfamiliar user interfaces. They observed that as users' familiarity with the type of interface increased, the higher their performance tended to be. User knowledge of computers, in most cases directly derived from experiences using them, has been cited as the most significant factor affecting performance on computer tasks (Moran, 1981).

Applying ATI Research to Online Help Design

Aptitude x Treatment interaction research is appropriate where the instructional design problem involves determining how the elements of an instructional message might affect learning for certain individuals under certain task conditions (Clark & Salomon, 1986). In the design of online help messages, for users differing along dimensions of cognitive style and computer experience, an instructional designer must determine the level of information abstraction and the combination of media attributes to apply to maximize user performance.

The general and specific effects that aptitudes such as field independence have across a variety of instructional treatments need to be better understood. ATI research techniques may be usefully applied to investigate such effects. Snow and Lohman (1984) described the goal of theories of instructional treatment design: "There was a clear prescriptive goal for such a theory. It was the design of an adaptive instructional system . . . [providing] alternative instructional treatments to fit the major differences in aptitude profiles among students" (p. 350).

One of the advantages of computer-based instruction, including online help information, is its potential to adapt each presentation to the aptitude and ability characteristics of the individual learner. Where these characteristics can be reliably measured and are clearly understood, instructional design principles may be applied to adjust the presentation to optimize the fit between the learner and the lesson. Before proceeding to design and validate adaptive instructional systems; however, principles defining relationships between learner characteristics and specific instructional treatment variables must first be identified and their reliability established. This has been a fundamental research objective of instructional designers developing intelligent tutoring systems (Perez & Seidel, 1990; Wiggs & Perez, 1988). This study was designed to identify and measure specific ATI effects and to contribute toward the design of future adaptive instructional systems.

Another goal of this research was to test the validity of instructional message design principles based on Paivio's dual-coding theory applied to the design of online help in a GUI. In addition, this research attempted to determine whether Aptitude x Treatment interaction effects exist between an individual user's cognitive style or level of computer experience and the use of pictorial messages in online help. Evidence relating to such interaction effects has been systematically collected and examined in this study.

Definition of Terms

The term graphical user interface implies more than the use of a graphics display terminal to present a human-computer interface. There are four key aspects to the design and operation of a GUI: (a) the technique of direct-manipulation is broadly applied; (b) the state of data and program objects is consistently represented using nonverbal, pictorial (iconic) symbols; (c) changes in system states of interest to the user are visually perceivable when they occur; and (d) functions that cannot be controlled using pictorial symbols are accessible using textual menus of consistent structure and organization (Shneiderman, 1983).

The term field dependence-independence has been applied by Witkin et al. (1977) and many other researchers to designate a dimension of cognitive style. Witkin et al. described this trait as "the extent to which the person perceives part of a field as discrete from the surrounding field as a whole, rather than embedded in the field . . . or, to put it in everyday terminology, the extent to which the person perceives analytically" (p. 7). Further, they described field dependence-independence as "a broad dimension of individual differences that extends across both perceptual and intellectual activities" (p. 10).

In general terms, experience has been defined as knowledge or skill gained through activity or practice. The term computer experience has been used in this study to refer to both the extent of prior computer interaction activities and the types of prior computer interaction experience (e.g., interaction with different types of user interfaces, computer applications, and systems). Furthermore, computer experience is operationally defined in this study as the score obtained on the experience scale of the Computer Experience and Competence Survey.

Pictorial message elements are the structured components of an information display, comprised of organized visual-iconic symbols, and designed to convey specific meaning. Pictorial (visual-iconic) symbols also are differentiated from textual (verbal-digital) symbols. Pictorial message elements may be either static or dynamic. Dynamic pictorial elements periodically or continuously change in appearance, whereas static pictorial elements have fixed visual appearance. In this study, digital motion video playback sequences were used to operationalize dynamic pictorial message elements within the instructional treatment.

Hypotheses

This study was designed to answer several research questions relating to a learning situation in which computer-based instruction was implemented using online help displayed in a GUI, where the GUI was unfamiliar to the users, and where the instruction was systematically varied by adding dynamic pictorial elements to text-based help displays. The research questions were stated in the form of the following null hypotheses:

1. No significant differences in application task performance result from a three-way interaction among field dependence-independence, prior computer experience, and the presence of dynamic pictorial message content in online help.

2. No significant differences in application task performance result from an interaction between prior computer experience and the presence of dynamic pictorial message content in online help.

3. No significant differences in application task performance result from an interaction between field dependence-independence and the presence of dynamic pictorial message content in online help.

4. No significant differences in performance on computer application tasks exist between subjects viewing text-only online help and subjects viewing online help containing text and dynamic pictorial elements.

5. No significant relationship exists between prior computer experience and a computer user's performance on computer application tasks in an unfamiliar GUI.

6. No significant relationship exists between field dependence-independence and a computer user's performance on computer application tasks in an unfamiliar GUI.

Assumptions and Limitations

The hypotheses given above state the core questions of this research. In attempting to find answers to these research questions, a number of assumptions were made and certain limitations were accepted which constrained the research problem and the generalizations which might be made regarding the results. These assumptions and limitations are discussed in detail below.

Variance of Treatment Duration

The time that subjects spent using online help messages was another, potentially confounding variable in this study. The use of online help, and the selection and display of dynamic pictorial elements, was entirely subject to individual user control and discretion during task completion. The use of online help by subjects participating in this study was assumed to be representative of their use of help in similar computer application learning tasks. The between subjects variance of the use of online help was statistically controlled by treating time in help as a concomitant variable in the analysis of covariance.

Population Sampled

The subjects for the experiment were sampled from an adult population of primarily undergraduate university students enrolled as business college majors. This population was expected to exhibit a unique and characteristic distribution of cognitive style and computer expertise. The generalization of results in this study has therefore been restricted to this population. Caution should be used in generalizing any results from this study to other populations.

Task Motivation

The computer application and the nature of application tasks were selected to be meaningful and relevant to the sample population. Individuals from the sampled population (business majors at a university) were required to demonstrate competencies in computer operations, specifically spreadsheet applications. In addition, tasks were arranged in a sequence such that the completion of each task was one step toward a project goal (e.g., creating and printing a graphical representation of a small company's annual balance sheet data). The tasks, therefore, had intrinsic incentives that were expected to increase subjects' motivation to learn the operations of the computer system and to complete application tasks.

Novelty Effects

The computer-based instructional treatments (online help messages) were assumed to involve a degree of novelty because the sample was comprised of students with varying computer experience, but who had no prior exposure to the computer system used--IBM Operating System/2 [1] version 2.0 (OS/2)--and its direct-manipulation GUI. At the time of this study, this version of OS/2 was a new product with many new features, particularly with respect to the design and operation of its GUI. Novelty effects may have also derived from the use of digital motion video technology to present the dynamic pictorial message elements in online help. The presence of these novelty effects was considered when describing and characterizing the results of this study.

Summary

Principles of instructional message design should be carefully applied to the design of online help in human-computer interfaces. Where these interfaces employ direct-manipulation techniques and rely on the use of pictorial (visual-iconic) symbols to represent computer state and function, online help should also employ similar presentation symbologies. For some individuals, learning should improve as the instructional conditions more closely resemble the criterion task performance conditions. These learning benefits, however, may be altered or limited by individual differences.

Computer user characteristics, particularly cognitive style and computer experience, may influence learning and performance in the human-computer interface. The combination of specific online help designs with these user characteristics may result in detectable Aptitude x Treatment interactions. Detailed understanding of such ATI effects may prove to be useful in the design and development of adaptive, computer-based instructional systems. This study was designed to identify and measure relationships that exist among the online help message design variables and the user aptitude variables (computer experience and field dependence-independence), and to

---

[1] Operating System/2 and OS/2 are registered trademarks of International Business Machines Corporation.

recommend directions for future research on related problems of instructional design for human-computer interfaces.

CHAPTER 2
REVIEW OF LITERATURE

Overview

This study was designed to examine the effects of field independence and computer experience on application task performance in an unfamiliar graphical user interface (GUI). This study also was designed to investigate whether Aptitude x Treatment interaction effects occur between computer users' cognitive style (field dependence-independence) or prior computer experience and the use of dynamic pictorial message elements in online help messages. The research questions addressed by this study were derived from an examination of theories contributing to research on instructional message design, human-computer interaction, cognitive styles, and computer expertise. This chapter describes these theories and where they intersect, identifies issues raised in previous research, and summarizes the body of literature relevant to the research questions addressed by this study.

Instructional Message Design for Visual Learning

Modality in message presentation refers to the sensory modes utilized to convey meaning. One principle of instructional message design states that "the modality used in the final testing or application situation should be the modality employed during instruction" (Fleming & Levie, 1978, p. 106). This principle is relevant to the design of online help where the criterion tasks require performance in a highly pictorial or graphical human-computer interface. It suggests that instruction (e.g., information presented in online help) should incorporate supporting pictorial message elements together with textual elements.

The symbol systems used to convey information during instruction differ in their capacities to support the extraction of meaning (Salomon, 1978). The better a symbol system can convey the critical features of an idea or event, the more appropriate it should be for instruction. In a direct-manipulation user interface, where tasks involve the manipulation of iconic visual symbols, the online help should directly incorporate those iconic symbols, rather than simply refer to them with verbal descriptions.

Dual-Mode Theories

Dual-mode theories suggest, and evidence supports the argument, that repeating verbal information with visuals results in significantly greater learning over verbal messages alone (Fleming, 1987). Dual-coding theory contends that two independent information encoding mechanisms exist. One stores and processes information as verbal codes while the other stores and processes information as visual images. These two modes also are referred to as analytic and analogic modes, respectively (Clark & Salomon, 1986).

Paivio's dual-coding theory of visual learning clearly suggests that to support construction of an adequate mental model of the computer system and its operations, the online information should attempt to present messages describing those operations using visual, nonverbal stimuli in addition to textual, verbal stimuli (Rieber & Kini, 1991). Both static and dynamic visual stimuli may be incorporated into the online messages. Because a direct-manipulation GUI is inherently a dynamic pictorial display, it follows that adding dynamic pictorial message elements to online help would enhance its effectiveness. This study was designed in part to examine certain effects of using dynamic pictorial message elements in online help.

Dynamic Pictorial Message Elements

Dynamic pictorial message elements may be incorporated into online help using various techniques, such as animated graphics or motion video windows. The use of animated graphics in computer-based instruction is becoming increasingly common (Horton, 1990; Rieber, 1990). Instructional message design principles suggest that where temporal or directional concepts are being taught, dynamic pictorials may be used to visually portray these concepts and this will improve learning (Rieber & Kini, 1991). Operating a computer system with a GUI requires understanding the visible motion of dynamic pictorial symbols in the user interface. It follows that the use of dynamic pictorials in online help may improve task performance in the GUI. This study was designed to measure the effects of dynamic pictorial elements by incorporating digital motion video segments into online help messages.

Instructional message design principles suggest that instruction should incorporate dynamic pictorial symbols wherever such symbols are employed in a task environment, to support the extraction of meaningful information relevant to that task environment. These principles have not been tested, however, with respect to the design of online help in direct-manipulation interfaces. One goal of this study was to evaluate the utility of the dual-coding theory as applied to learning application operations in a graphical user interface.

Human-Computer Interface Design

Direct-Manipulation and Nonverbal Literacy

Graphical user interfaces were developed to allow the computer user to more directly manipulate an interactive computer system's state, instead of relying on command language interpreters (Shneiderman, 1983). The manipulation of visual iconic symbols displayed by the computer, using visual tools (e.g., an arrow pointer), results in an immediately visible change in system state. This is the central principle of direct-manipulation interface design. It requires the real-time animation of iconic display elements that are mnemonics for system or application states and functions. These visual iconic symbols appear to the user to represent controls that operate functions of the computer which otherwise would be invisible and more difficult to understand and manipulate.

Learning to control computer functions in a GUI involves different cognitive processes than are required of a user learning the same functions in a textual, command language user interface. This follows directly from research on verbal and nonverbal literacy (Sinatra, 1986). A learner's perception and memory of the syntax and semantics of verbally encoded messages depends on verbal language skills, which are sequential and analytical in nature. On the other hand, the perception and memory of spatial-temporal manipulation of pictorial elements depends on visual and kinesthetic processes, which are holistic and analogical in nature. This contrast between analytical (verbal) and holistic (nonverbal) processes also has been presented as the fundamental determinant of cognitive style differences (Miller, 1987). Thus, a theoretical link can be proposed between dual-coding theory and cognitive style theories. This link provides a basis for this study.

Symbol Systems and Mental Models

An instructional message design that is appropriate for teaching the skills required in a verbal command language interface may be inadequate or inefficient for teaching the skills required in a predominantly iconic interface. This derives from Salomon's theory of media attributes. "The closer the match between the communicational symbol system and the content and task-specific mental representations, the easier the instructional message is to recode and comprehend" (Clark & Salomon, 1986, p. 468).

Learning to manipulate a computer system's functions requires the user to develop an internal representation, or mental model, of the system (van der Veer, 1990). A mental model may be based largely upon propositions encoded verbally, as would be expected for users of a command language interface, or it may be based predominantly on analogical images. An effective mental model should parallel the organizational metaphors depicted in a GUI. Factors that influence the development of these mental models are significant determinants in the design of user interfaces. The ease with which a user creates an adequate mental model of a computer system or application largely determines the productivity that user will be able to achieve.

The formation of mental models while learning computer functions in a graphical user interface depends more heavily on analogical, rather than analytical, information processing. Individual differences in cognitive style, particularly field dependence-independence, are characterized by differing tendencies to exercise analytical information processing. This theoretical link between cognitive style and construction of mental models provides a basis for a deeper understanding of how field independence may influence performance on tasks in a GUI.

Direct-manipulation user interfaces support formation of visual as well as verbal mental models, which computer users may construct to help them manipulate system and application functions. The ability of users to form correct mental models has been demonstrated (van der Veer & Wijk, 1990). Performance on application tasks in a GUI is believed to depend on the user's ability to form effective mental models. To the extent that online help can be designed to facilitate this ability, performance should improve.

Cognitive Style Effects

The Dimensions of Cognitive Style

Cognitive styles are psychological dimensions that represent consistent tendencies in an individual's manner of acquiring and processing information. Gregorc (1984) indicated that "stylistic characteristics are powerful indicators of deep underlying psychological forces that help guide a person's interactions with existential realities" (p. 54). Many dimensions of cognitive style have been reported. Canelos, Taylor, Dwyer, and Belland (1988) summarized nine different cognitive style dimensions.

Miller (1987) employed an information processing model of cognition in his analysis of eight dimensions of cognitive style. The most compelling and authoritative research on cognitive styles, however, has been conducted regarding the dimension of field dependence-independence, which has been extensively studied by Herman A. Witkin and his associates.

Field Dependence-Independence (FDI)

Begun in 1941, Witkin's research into the human phenomenon of field independence has been extensively reviewed, extended, and broadly applied. Witkin's early research detected significant individual differences in perceptual abilities, particularly the ability to perceive an upright object embedded in a tilted frame (Witkin et al., 1977). The concept of field dependence-independence was first described by Witkin in 1954 (Canelos et al., 1988).

The process of attention is the selective focusing of conscious mental activities. Research shows that there are both deliberate and automatic forms of attending to stimuli, and that there is evidence for individual biases toward relying on one form or the other. Individual differences in selective attention have been found using various tests to measure field dependence-independence. As measured with the Embedded Figures Test (EFT), relatively field-independent persons exhibit deliberate attention focusing and an ability to disembed an item from an organized context of distracting cues. Relatively field-dependent persons, on the other hand, exhibit a deficit in this regard or a tendency toward relying on more automatic attention processes (Miller, 1987).

When compared to more field-independent learners, relatively field-dependent learners are less able to identify discrete objects in complex visual fields, but are better able to perceive and identify patterns in complex visual fields (Witkin et al., 1977). In their work defining this dimension of cognitive style, Witkin et al. elucidated the bipolar, process-oriented, enduring, and pervasive characteristics of cognitive style.

Although other dimensions of cognitive style, such as impulsivity-reflectivity, also are being investigated with respect to performance on tasks involving computers (van Merrienboer, 1988, 1990), the FDI dimension is overwhelmingly the most frequently studied.

Cognitive Style and Computer-Based Learning

Many researchers have investigated relationships between the level of field dependence-independence of computer users and various aspects of their performance when learning with or from computers (Burwell, 1991; Canelos et al., 1988; Canino & Cicchelli, 1988; Cathcart, 1990; Cavaiani, 1989; MacGregor, Shapiro & Niemiec, 1988; Martin, 1983; Mykytyn, 1989; Post, 1987). Despite sometimes inconsistent findings, the frequency and recency of these studies indicate that compelling purposes motivate this research. Some of these studies were designed to detect Aptitude x Treatment interaction effects between learner cognitive style and instructional treatments. One objective of this study was to systematically measure relationships between cognitive style and instructional message design variables.

Learners' cognitive style can significantly affect their ability to perceive, remember, and apply declarative and procedural knowledge. Due to their greater analytic capacity, relatively field-independent learners exhibit greater skill disembedding simple visual stimuli embedded within a complex field. More field-dependent learners perform relatively poorly at such tasks, due to their greater tendency to process information in a holistic manner. By identifying an individual learner's cognitive style, computer-based instruction may adapt the presentation to the individual by appropriately varying certain instructional message design parameters under software control. Researchers believe that this may significantly improve learning from computer-based instruction (Canelos et al., 1988).

Comparative Expertise Effects on Mental Models

Expert-Novice Differences

Experts and novices approach problem solving in different ways (Aster & Clark, 1985; Lesgold et al., 1990). Some researchers have suggested that the underlying conceptual model in a GUI, evident to expert users, is undetected or misinterpreted by novice users. These expert-novice differences refer specifically to differing levels of experience, not to different levels of general intelligence (Aster & Clark, 1985; Mestre & Touger, 1989).

HCI research also suggests that cognitive processes employed by computer users early in the use of a new user interface differ from those used later. This may result from changes in the nature of tasks presented (e.g., tasks become more complex and, therefore, more difficult) or because users develop new task completion strategies over time as their expertise increases (Chiesi, Spilich, & Voss, 1979). Another explanation is that the user's mental models gradually become more complete and accurate (van der Veer & Wijk, 1990).

Whiteside et al. (1985) found that the performance of users who differ in prior computer experience was consistent across several different user interfaces. Regardless of the user interface style presented, novice users with little or no prior computer experience performed at the lowest levels. Those users with prior computer experience, but not with the particular user interface tested, scored higher than novices regardless of the interface. Those users who had extensive prior experience with the user interface being tested performed consistently better than either novice users or those with experience in a different user interface. This study was designed to examine the effects of prior computer experience on learning application function when an unfamiliar user interface is encountered.

Computer expertise is not, however, a single, monolithic dimension. At a minimum, prior experience must be evaluated both in terms of extent (depth) and range (breadth). An expert user may have considerable depth of experience, yet be limited to a single type of user interface or system and thus have limited breadth of experience. An expert user with both depth and breadth of experience has worked on a variety of systems with differing interface styles and has sufficient exposure to the operation of each to be adept at manipulating and controlling many of its functions.

An expert user's experience is, however, rarely global or comprehensive. No matter how extensive a user's experience, there are aspects of system function that may never be explored. This is partially due to the complexity of modern computer systems and partially due to limited user needs. Systems are designed to fulfill the requirements of a wide variety of users, but each individual user requires only a subset of that system's functions to perform the tasks at hand. As a computer user's tasks become more varied and complex, the range of the user's experience with an interface will increase.

In addition, user experience level is not a stable factor. Whenever a user engages in a new type of task, that user becomes more experienced (Aster & Clark, 1985; Federico, 1983). The changes that occur in the user during this progression include becoming more consistent and precise, engaging in more complex forms of task abstraction, using more automatized and internalized strategies for performance, and becoming increasingly skillful in the application and interpretation of the rules for performance.

Mental Models and HCI Metaphors

A user's mental models of a computer system are continually being modified, extended, and tested against the actual behaviors of the system (van der Veer & Wijk, 1990). Users differ in the style of representation that they apply to construct mental models, either verbal, visual, or dual-mode styles. "The learning process that leads to the mental model will be based on analogies to known situations and systems. The learning process may be facilitated by providing adequate metaphors" (p. 195). In the case of direct-manipulation interfaces, visual metaphors are used in the interface design and appear as pictorial elements which may be visually manipulated by the user. A visual metaphor that extends a conceptual model assists with perceptual processing and the formation of a correct mental model.

Comparative expertise research has demonstrated that experts and novices differ in the way they perceive and solve problems. Research on expert-novice differences has also shown that these differences are related to experience in a specific task domain and are not a measure of general intelligence. The mental models possessed by novices are marked by their simplicity and incorporation of surface features, while those possessed by experts reflect greater abstraction and organization according to fundamental principles related to the task domain. Visual metaphors presented using pictorial symbols can assist novice computer users with learning system function by supporting formation of mental models.

Measuring Computer Expertise

One of this study's objectives was to examine the relationship between computer users' prior experience and their ability to learn application functions in an unfamiliar GUI. The selection or construction of an instrument to reliably measure computer experience was therefore of primary importance.

Instruments for assessing computer literacy, experience, and knowledge have been the subject of much research. During the past decade, computers have become essential tools applied in nearly every occupation. As a result, public schools, colleges and businesses large and small have had to determine the computer skills of their students and employees. However, development of instruments to measure computer skills, knowledge, and expertise has lagged behind concurrent rapid changes in computer technology. Such tests are difficult to design and validate since the subject matter and the skills to be evaluated are continually changing as computers rapidly evolve (LaLomia & Sidowski, 1990).

For this study, computer expertise was defined as the combination of an individual's computer experience with that individual's computer knowledge or competence. Experience with computers is acquired over time, is cumulative and incremental. It relates the extent and type of computer usage the individual has engaged in, and is typically measured using a self-reporting, survey-type instrument. Knowledge of, or competence with, using computers is highly dependent on the specific computer systems and programs involved, although this knowledge may transfer more or less well from one system or program to another. Computer competence may be measured either by administration of an objective test composed of cognitive knowledge items, or it may be measured directly within the context of task-specific computer usage.

Published computer literacy, competency, and knowledge assessment instruments have become outdated by the significant and rapid changes in computers and computer applications. One area of computer skills and knowledge particularly exposed to rapid change is the use of application programs such as word processors, spreadsheets, database systems, and graphical or drawing editors. Assessing expertise in the use of such computer application programs requires measurement of performance on those aspects of application software which impinge directly on the user's ability to control the computer's function. Such control is exercised through an interaction dialog with the computer via the human-computer interface. An instrument designed to measure computer expertise must have a component that measures the ability to interact with software through a variety of user interfaces, both visual-iconic and verbal-digital in nature.

Components of Expertise

The measurement of expertise--the quality or aptitude of being an expert within a particular domain--must comprise the measurement of both experience and knowledge. Experience can be expressed in terms of frequency and duration of practice within the domain. For computer user expertise, this may be expressed as a numerical value ranging from zero, indicating no experience, to some arbitrarily high value, indicating the highest experience level among the particular sample of individual users.

An equally important aspect of expertise is cognitive knowledge regarding the functions of computer systems and applications and how the software may be operated. Knowledge about computer systems and operations is frequently referred to as computer literacy in the research literature on educational computing. In a review of six computer literacy and aptitude scales, LaLomia and Sidowski (1990) found all six had defined computer literacy and operationalized their definitions in different ways. Items testing knowledge of computer operations or applications appeared in most of these instruments.

Both experience and knowledge were measured in the Computer Competence Test developed for the 1986 National Assessment of Educational Progress (NAEP) (Educational Testing Service, 1988). In creating the NAEP Computer Competence Test, Martinez and his colleagues at ETS extensively and systematically developed 228 objective, multiple-choice items to measure computer experience and knowledge (Martinez & Mead, 1988). These included items covering general knowledge of computer systems, knowledge of four common types of computer applications, and knowledge of two computer programming languages. Items designed to assess student attitudes towards computers were also included. The test was administered to students in the third, seventh, and eleventh grades during the 1985 to 1986 school year.

All of the published and privately available computer experience and computer literacy instruments reviewed had serious content validity problems due to being published more than five years previous to this study. Thorough editing and revision would have been required to improve the validity of these instruments. The Computer Competence Test developed for the NAEP was selected for use in this study as the most comprehensive instrument. All the original NAEP test items were obtained from ETS.

The selection of items used, the editing of item text, and other modifications to the NAEP items made for this study are described in Chapter 3.

The measurement of computer literacy and expertise has become a subject for much research in the past decade. The availability of valid and reliable instruments for measuring computer expertise has been limited by the rapid changes in computer technology and applications. Computer competence tests developed prior to the widespread availability of graphical user interfaces would require substantive changes to accurately measure expertise with current computer systems. The instrument used in this study to measure computer experience was derived from the Computer Competence Test developed for the 1986 National Assessment of Educational Progress.

Summary

This study was designed to investigate an apparent link between the theory of field dependence-independence and the dual-coding theory of visual learning. This link was predicated on a characterization of the mental processes for visual learning as being more holistic-analogical than those for verbal learning. Research into the formation of mental models by users of direct-manipulation computer interfaces provided a paradigm for this investigation. Researchers have found that relatively field-independent computer users learn the operations of some computer interfaces more readily than do field-dependent users. This suggests that field-independent users may develop mental models more efficiently than field-dependent users. By manipulating the pictorial content of online help messages, this study attempted to test whether users with different levels of field independence would respond differently to varying levels of visual-iconic content.

Prior computer experience has been identified as the most significant factor contributing to successful performance in learning new human-computer interfaces. This study examined the influence of computer experience on performance in an application where information about application operations was presented using online help.

Instruments designed to measure computer experience, knowledge, literacy, and competence have not kept pace with the rapid changes in human-computer interfaces. In order to reliably determine the effects of computer experience on performance, instruments that accurately measure experience must be developed. This study employed one such instrument and attempted to demonstrate its validity and reliability.

CHAPTER 3
METHODOLOGY

Introduction

This study was conducted to examine the effects of field independence and computer experience on learning computer application functions in an unfamiliar human-computer interface. Students who were enrolled in a university-level business curriculum completed a computer training session during which their use of online help was observed and their performance on computer tasks was measured. In the training, students were randomly assigned to one of two different online help formats. Online help provided information on how to use the computer system and a spreadsheet application. One online help format displayed text-only information, while the other format displayed dynamic pictorial elements in addition to text. The display of help messages in both online help formats was under individual student control at all times. The dependent variable was performance on application tasks in the training posttest.

A multiple covariance analysis was used to examine effects of the online help format, prior computer experience, cognitive style (field dependence-independence), and the amount of time in help on application task performance. This analysis also tested for potential Aptitude x Treatment interaction (ATI) effects among field dependence-independence, computer experience, and the display of dynamic pictorial content in online help.

Before proceeding with this study a pilot study was conducted (Tyler, 1993). The objective of the pilot was to validate the instructional materials and performance measurement procedures and to collect data to support methodology decisions which could not be made based solely on existing literature. Results of the pilot study that directly influenced the design of this study are described where appropriate. This chapter presents the experimental design, the population and sample, the assessment instruments, the instructional treatments and materials employed, and the data collection methods used in this study.

Experimental Design

A fully randomized experimental design with pretest-treatment-posttest sequence was employed in this study. A random sample of 38 subjects was obtained from a population of university students enrolled in an undergraduate business management course. The subjects completed assessments for field independence and prior computer experience. Each student in the sample was then randomly assigned to one of two treatment conditions consisting of different online help formats. Both help formats displayed identical textual information, while one displayed dynamic pictorial elements in addition to the text. Each student completed individual computer training in the use of a personal computer equipped with a direct-manipulation graphical user interface (GUI) and a graphical spreadsheet application. The training consisted of a pretest, 12 training lessons, and a posttest. During the training, online help was the primary instructional resource available to students. For the students to obtain information on operating the computer system or the spreadsheet application, they had to activate online help displays. At the completion of training, students' posttest performance scores were analyzed using an analysis of covariance.

Analysis of Covariance

Data collected in this study were interpreted using a one-way analysis of covariance (ANCOVA) with multiple covariates. This analysis was performed using an ANCOVA hierarchical regression analysis technique (Cliff, 1987). This technique is equivalent to multiple covariance analysis through multiple regression (Huitema, 1980). In this multiple covariance analysis, the effects of the treatment factor (a categorical variable) and multiple covariates (interval scale concomitant variables) on the dependent variable were determined using linear regression models. The use of ANCOVA in this study had three purposes: (a) determine whether significant interactions had occurred between the online help format and any of the covariates, (b) test the effects of the two different online help formats on the dependent measure, and (c) examine the regression relationships between the concomitant variables (covariates) and the dependent measure. This approach for ANCOVA emphasized a linear regression analysis between the interval dependent and independent variables, within each level of the treatment factor.

Independent variables. There were four independent variables in this analysis. The two aptitude variables, field dependence-independence (FDI) and computer experience (CEXP), were measured as interval scale random variables. FDI was measured using the Group Embedded Figures Test (GEFT) (Witkin, Oltman, Raskin, & Karp, 1971). CEXP was operationalized as the score on the experience scale of the Computer Experience and Competence Survey, which was developed for this study using items selected from the Computer Competence Test of the 1986 National Assessment of Educational Progress (Educational Testing Service, 1988).

Time in help (THELP), a third interval scale independent variable, was measured using a computer software logging program. THELP reflected the accumulated time, in minutes, that a subject displayed online help messages during the 12 training lessons. The final independent variable was the online help treatment factor (TREAT). TREAT was a categorical variable with two levels representing the two online help formats to which subjects were randomly assigned. The first treatment level was text-only help (TOH), wherein online help displays contained only textual information. The second treatment level was text-with-motion-video help (TMVH), which displayed the same text as TOH, but also displayed--when initiated by the subject--dynamic pictorial message elements in the form of motion video windows.

Dependent variable. The dependent variable in this analysis measured the subject's performance on computer operation tasks in the direct-manipulation GUI. Application task performance was computed as the training posttest performance score (POST). The training posttest, administered at the completion of the training lessons, was composed of three application tasks involving operation of a graphical spreadsheet program. This performance measure accounted for both task completion accuracy and time-on-task.

Despite its computational complexity, ANCOVA was the most appropriate analytical procedure for this study. This was due primarily to the continuous nature of measurements used to assess both field independence and computer experience (Cliff, 1987). Since interval measures on these aptitude variables were used, rather than converting the interval scores into levels of blocking variables, greater precision was attained in determining the relationships between each aptitude variable and the dependent measure. Also, in evaluating the effect of the online help format, ANCOVA reduced any unintentional bias in the comparison of treatment groups that might have occurred as a result of experimental mortality or sampling fluctuations. Comparison of treatment groups was done on the basis of adjusted group means rather than actual group means. Adjusted means removed the within-group variance accounted for by the regression of the dependent variable on each of the covariates. The adjusted treatment mean differences could therefore be interpreted as independent of all variables used as covariates (Huitema, 1980).

Assumptions for ANCOVA

There are several assumptions for the analysis of covariance, including those which apply to analysis of variance. First, the observations must be independent. In this study, treatments were individually administered, so this assumption was met. Second, the dependent measure (POST) was normally distributed within each treatment group, satisfying the assumption of normality of $\underline{Y}$ scores. Additional assumptions apply in a unique manner to ANCOVA and are discussed in detail in the following paragraphs.

Homogeneity of regression slopes. This assumption for ANCOVA states that the regression slopes of the dependent variable on any covariate must not be significantly different between treatment groups. If the regression slopes associated with the various treatment groups were not the same, it would mean that the ANCOVA model did not fit the data. In that case, an alternative method such as the Johnson-Neyman technique would have been required (Stevens, 1990). Testing this assumption was the first step in the analysis technique applied in this study.

Homogeneity of variance. In ANCOVA, there should be no significant differences in the distributions for each of the covariates between the different levels of the treatment variable, particularly when group sizes are unequal (Stevens, 1990). In this study, subjects were randomly assigned to one of the two online help formats. After accounting for mortality effects, 18 subjects completed training in the TOH group while 20 subjects completed training in the TMVH group. Because this resulted in an unbalanced design, a test of the homogeneity of variance was required.

Independence of treatment and covariates. In ANCOVA, the treatment should not directly influence the covariate scores obtained. In this study, the aptitude variables CEXP and FDI were obtained prior to, and therefore independent of, the instructional treatment. The third covariate, THELP, was measured during training and could have been indirectly influenced by the treatment. An ANOVA on THELP was performed to determine that this covariate was independent of treatment (Huitema, 1980).

Linearity of regression. ANCOVA assumes a linear relationship between each covariate and the dependent measure. For this data set, a visual inspection of scatterplots generated when POST was plotted against CEXP, FDI, and THELP demonstrated that the assumption of linear relationships was tenable.

Selection of Covariates.

To conduct ANCOVA for this data set, each of the proposed covariates was scrutinized for appropriate use in the analysis following two recommendations by Stevens (1990). First, independent variables that were correlated with the dependent variable could be used as covariates. For this study, the relevant Pearson correlations were: (a) CEXP with POST, $r = 0.63$; (b) FDI with POST, $r = 0.42$; and (c) THELP with POST, $r = -0.51$. Given these correlations, CEXP, FDI, and THELP were appropriate for use as covariates with POST as the dependent variable.

Second, when multiple covariates are used they should not be strongly correlated with each other, so that each may contribute a unique component to the analysis. For this analysis the relevant Pearson correlations were: (a) CEXP with FDI, $r = 0.21$; (b) CEXP with THELP, $r = -0.40$; and (c) FDI with THELP, $r = -0.19$. Except for the moderate correlation between CEXP and THELP, these covariates contribute separately to the analysis.

As a final consideration for selecting covariates, Huitema (1980) recommended limiting the number of covariates used, in relation to the sample size, to prevent the adjusted means from becoming unstable. Following this recommendation, the ANCOVA model in this study included only three covariates.

Computing ANCOVA with Three Covariates

Performing analysis of covariance with three covariates is not a common statistical procedure. In general, it is an extension of a one-way ANCOVA, but with multiple covariates. Huitema (1980) referred to this as multiple covariance analysis. This method proceeded in three steps. First, a test for the assumption of homogeneous regression slopes was performed. This test determined whether there were significant interactions between online help format and any of the three covariates. Second, after confirming that no significant interactions had occurred, a test was performed to determine whether the online help format had a significant effect. Third, the analysis determined whether there were significant regression effects on the dependent measure for any of the three covariates.

For all computations described below, the sources of variance for each ANCOVA model were computed using the Statistical Analysis System (SAS)[1], release 6.07. Procedures for programming the SAS General Linear Models (GLM) procedure to perform the ANCOVA hierarchical regression analysis were derived from Cliff (1987) and from SAS User's Guide: Statistics (SAS Institute, 1985). Certain statistical tests not supported in the SAS GLM procedure were also required. The formulas for calculating these test statistics are described below.

Test for homogeneous regression slopes. In this study, ATI effects were anticipated by the research questions. The test of the ANCOVA assumption of homogeneous regression slopes was a key first step in the analysis, because it tests for interactions between the covariates (aptitude variables) and the treatment factor. Had significant interactions been found, continuing with the remainder of the multiple covariance analysis would not have been appropriate.

Testing the assumption of homogeneous regression slopes required computing the sum of squares for two linear regression models, referred to as the complete model and the reduced model (Agresti & Agresti, 1979). These two regression models were used to calculate the between-groups regression sum of squares and the within-groups regression sum of squares. An F test statistic was then computed as the ratio of the mean squares between over the mean squares within.

The general form of the complete ANCOVA model with 3 covariates is shown in Equation 1. This linear regression model yields predicted Y scores (Y_{Pred}) and includes the Y-intercept (α), Y-intercept difference parameter (δ) for the effect of

---

[1] SAS is a registered trademark of SAS Institute, Inc.

treatment (A), regression slope parameters (β_i) for each covariate (X_i), and cross-product term coefficients (γ_i).

$$\underline{Y}_{Pred} = \alpha + \delta A + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + \gamma_1(X_1 A) + \gamma_2(X_2 A) + \gamma_3(X_3 A) \qquad (1)$$

The reduced ANCOVA model, shown in Equation 2, follows the form of the complete model but eliminates the interaction terms. The reduced model calculates predicted $\underline{Y}$ scores assuming there were no interactions between treatment and the covariates.

$$\underline{Y}_{Pred} = \alpha + \delta A + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 \qquad (2)$$

The regression sum of squares and error sum of squares were computed for the complete and reduced models shown above. These values were then used to compute the $\underline{F}$ statistic to test for significant interactions between the covariates and the treatment. This $\underline{F}$ statistic, calculated using Equation 3, tests the assumption of homogeneous regression slopes (Cliff, 1987).

$$\underline{F} = \frac{(\underline{SS}_{Err}{}^{R} - \underline{SS}_{Err}{}^{C}) \;/\; p(g - 1)}{\underline{SS}_{Err}{}^{C} \;/\; (\underline{N} - pg - g)} \qquad (3)$$

In Equation 3, $\underline{SS}_{Err}{}^{R}$ is the error sum of squares for the reduced model, $\underline{SS}_{Err}{}^{C}$ is the error sum of squares for the complete model, p is the number of covariates, g is the number of groups, and $\underline{N}$ is the total number of subjects. If the resulting value of $\underline{F}$ did not exceed the critical value, $\underline{F}_{[\alpha=.05, p(g-1), \underline{N}-pg-g]}$, the ANCOVA assumption of homogeneous regression slopes would have been met.

Test for significant treatment effect. Where the assumption of homogeneous regression slopes was valid (i.e., no significant interactions between treatment and covariates had occurred), the analysis next tested the treatment effect. For this test, another $\underline{F}$ statistic was calculated to determine whether the regression lines representing the two treatment groups had different $\underline{Y}$-intercepts.

Similar to the test for homogeneous regression slopes, this test statistic was computed using the error sums of squares for a complete model and a reduced model.

To compute this $\underline{F}$ statistic, the complete model was the ANCOVA regression equation without interaction terms--the reduced model from the previous test--shown in Equation 2. The reduced model for this test was the regression equation with only terms for the covariates, shown in Equation 4.

$$\underline{Y}_{Pred} = \alpha + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 \tag{4}$$

The formula for calculating the $\underline{F}$ statistic for this test is shown in Equation 5. As in the previous test, this equation compares the error sum of squares for the complete model ($\underline{SS}_{Err}{}^{C}$) with the error sum of squares for the reduced model ($\underline{SS}_{Err}{}^{R}$). Also, p is the number of covariates, g is the number of groups, and $\underline{N}$ is the total number of subjects (Cliff, 1987).

$$\underline{F} = \frac{(\underline{SS}_{Err}{}^{R} - \underline{SS}_{Err}{}^{C}) \;/\; (g - 1)}{\underline{SS}_{Err}{}^{C} \;/\; (\underline{N} - g - p)} \tag{5}$$

This test determined whether there were significant differences between the two online help treatment groups on the dependent variable POST. Where the resulting value of $\underline{F}$ did not exceed the critical value, $\underline{F}_{[\alpha=.05, g-1, \underline{N}-g-p]}$, the null hypothesis of no significant treatment effect was not rejected.

Test for significant regression effects. The third and final step in this analysis was to test whether there were significant regression effects on task performance for each covariate. In this step, the ANCOVA resembled a multiple regression analysis and the covariates became predictor variables (X_i) in a prediction equation, as shown in Equation 4.

For each covariate, an $\underline{F}$ statistic was tested to determine whether a significant relationship existed between the covariate and the dependent variable, posttest performance. In other words, this tested whether the regression slope coefficients (β_i) in this prediction equation were nonzero. These $\underline{F}$ statistics were computed using a standard linear regression analysis procedure.

The analysis of covariance method described above was used to test all the hypotheses for this study. ANCOVA provided greater power than alternative methods, but involved greater computational complexity. Given that this study was primarily focused on examining the regression effects of the aptitude variables (covariates) on application task performance, and determining whether Aptitude x Treatment interactions had occurred, ANCOVA was the most appropriate analytical method for the data gathered in this study.

Population and Sample

The population from which subjects were drawn for this study consisted of students majoring in business curricula at a state university in southern Florida. Subjects were randomly sampled from all students enrolled in four sections of Management Information Systems, an undergraduate course required of all students majoring in a business college program at the university. A total of 129 students comprised the available population pool. In this student population, 85% were upper class undergraduates. After identifying the population and conducting initial screening, individual computer training sessions were held at a nearby corporate product evaluation center.

At the initial screening, all students were asked to volunteer for the study. In return they received free training on an advanced personal computer operating system and a graphical spreadsheet application. No other form of compensation or course credit was given for participation. Because the subjects were business majors who were required to learn computer operations, and because they were volunteers, a high level of motivation to complete the training was anticipated. Experience with similar students during the pilot study had confirmed this.

From the four course sections, all 129 students were screened using the Sign-up Form for OS/2 Training and the Computer Experience and Competence Survey. These instruments were administered during regular class sessions. The sign-up form was designed to collect demographic and computer experience data. It is included here as

Appendix A. This form was also designed to check whether students had prior experience using the computer operating system (IBM Operating System/2 version 2.0). Individuals in the population who indicated prior experience using this system were eliminated prior to selection. Because the graphical spreadsheet application, PM Chart[2], was included as a feature of the OS/2 operating system product, experience with this application was also determined in the initial screening.

Power estimate and sample size. Based on data gathered during the pilot study, a relatively large effect size ($f > 0.80$) was estimated for the effect of computer experience on task performance. A small effect size ($f < 0.50$) was anticipated for the effect of field independence on task performance. Using these effect size estimates, the sample size for the study was set at 60 subjects using Cohen's power tables (Stevens, 1990). This value was based on setting $\alpha = .05$, group size to 30, and achieving power of 0.87 for an estimated effect size of $f = 0.40$. This effect size corresponds to that observed in the pilot study for the effect of field independence on posttest performance.

Experimental mortality. From the population pool, 60 student volunteers were randomly selected and were scheduled for training appointments. Experimental mortality resulted from appointment cancellations, no-shows, and incomplete training. Of the 60 volunteers, 18 students scheduled training appointments and later either cancelled the appointments or failed to appear for their training sessions. The remaining 42 students attended training sessions as scheduled. This sample was composed of 2 Sophomores (5%), 16 Juniors (38%), 20 Seniors (48%), and 4 students (10%) who reported other academic status. There were 26 males (62%) and 16 females (38%) in this sample.

Incomplete training also contributed to mortality. Of the 42 subjects who attended training, 32 subjects completed all 12 training lessons plus the pretest and

---

[2] PM Chart is copyright 1991 by Micrografx, Inc.

posttest. Of the ten remaining subjects, four completed fewer than eight lessons and were dropped from the analysis due to incomplete data. The loss of subjects due to these mortality effects contributed to power problems in this analysis. The loss of four low-experience subjects shifted the computer experience distribution slightly toward higher scores. However, the distribution of computer experience scores for the resulting sample of 38 was not significantly different from the distribution for the initial sample of 60. Two of the four subjects dropped had scored above the mean on the GEFT, while two had scored below the mean. Finally, these four subjects were evenly divided between the two online help formats. The balanced loss of subjects indicates that mortality did not bias the results.

Summary of the sample. The resulting sample of 38 subjects exhibited diversity on both computer experience and field independence measures. All subjects completed at least 8 of the 12 lessons during the training, covering all aspects of skills required for the pretest and posttest. GEFT scores for this sample ranged from 2 to 18. These 38 subjects had normally distributed scores on the computer experience subscale (CEXP) of the CECS instrument. CEXP scores for the sample ranged from 9 to 58. Although mortality effects reduced the size of the sample, it remained representative of the target population with regard to these aptitude measures.

Random Selection and Assignment

Using the CEXP scores obtained from initial screening, a stratified random sampling technique was used. The resulting sample's distribution of computer experience was similar to the distribution in the population pool. Using this approach, 60 students were initially selected from the volunteer student population to form a representative sample to participate in the study. These students were contacted directly to confirm their willingness to participate and to schedule their training appointments.

After having scheduled their training sessions, each student was randomly assigned to one of the two online help formats, text-only help or text-with-motion-video help. From a randomized list of volunteers, each was alternately assigned to TOH or TMVH to ensure equal treatment group sizes. Random assignment to treatment minimized initial differences between treatment groups. Mortality effects (students who scheduled training sessions but failed to appear) during the study caused treatment group sizes to become unequal. In this sample ($\underline{N}$ = 38), 18 were assigned to the TOH treatment, while 20 were assigned to the TMVH treatment. The analyses performed accommodated for this unbalanced design.

Instrumentation

Instruments and methods used to measure cognitive style (field dependence-independence), prior computer experience, and computer task performance were central aspects of the design of this study. The instruments chosen from published sources or developed for this study to assess these individual characteristics are described in this section.

Volunteer Sign-up Form

129 students were initially screened using the Sign-up Form for OS/2 Training (Appendix A). Responses to questions on this form provided student identification data such as name, phone number and academic status. It also included eight questions designed to estimate the students' computer experience. This data was used when students were contacted to schedule their individual training sessions. Students who indicated very little experience were scheduled for longer training appointments. Finally, the form was signed by students as an indication of their interest in voluntarily participating in the computer training and study.

Computer Experience and Competence Survey (CECS)

During initial screening, subjects completed the CECS instrument, a multiple-choice, self-reporting survey of prior computer use. This survey was composed of three scales: user demographics (CDEM, 8 items), experience survey (CEXP, 39 items), and cognitive items (CCOG, 47 items). The experience scale items were designed to measure the amount and types of prior computer experience. The competence scale includes items designed to measure knowledge of computer systems and applications and how they operate. This instrument, like the GEFT, was designed to be conducted in a group setting such as a classroom. The 94-item CECS instrument was timed to be completed within 30 minutes. Sample items from the CECS are included here in Appendix B.

The CECS items were derived primarily from items developed for the Computer Competency test of the 1986 National Assessment of Educational Progress (NAEP) (Educational Testing Service, 1988). For many items, the original NAEP versions were used verbatim. Some items were edited to account for differences in the target population (university students versus secondary school students). Other items were modified to accommodate the significant changes in personal computers during the six years between the development of the NAEP items and this study. NAEP items designed to measure computer programming knowledge were not included because those items did not appear to relate closely to knowledge of or ability to operate computer applications. The resulting CECS instrument was designed to determine the extent of a subject's prior computer experience and knowledge of computer systems, applications, and their operation.

Due to reliability and validity problems found in the CCOG (cognitive knowledge) scale during the pilot study, this scale was not used in the data analysis. Only scores from the CEXP (experience) scale were used in the study.

CEXP ecological validity estimate. The subjects' initial computer experience was also measured using a three-task computer operations pretest. The scores on the pretest were expected to correlate strongly in a positive direction with the subjects' CEXP scores. The pilot study had confirmed this strong positive correlation (r = 0.85), which establishes a reasonable measure of ecological validity for the CEXP scale.

<u>CEXP reliability estimate.</u> An analysis of the computer experience scale revealed problems with several items. First, items 17 and 18 were not included in the reliability estimate because they were not calculated into the CEXP score (these two items provide qualitative data only). In addition, items 43 through 47 formed another qualitatively scored component of CEXP and were also excluded from the reliability computation. The resulting CEXP scale included 32 items, but initially showed only moderate reliability ($\alpha = 0.61$).

A procedure to increase the reliability of the CEXP scale was used, based on a routine incorporated into the reliability procedure of the Statistical Package for the Social Sciences (SPSS), version 4.0. This routine used an iterative approach to increase reliability by removing individual items. As each item was removed, alpha was recalculated for the remaining items. All items were finally ranked in the order of decreasing negative influence on alpha. Using this approach, Chronbach's alpha for the CEXP scale was improved to 0.85 by removing nine items (listed in order of their removal: 42, 30, 12, 11, 40, 38, 10, 31, and 39). These items had the greatest negative influence on alpha. The final CEXP scale therefore had 23 scored items with improved reliability. The CEXP scores with improved reliability are used throughout the remaining data analysis and discussion.

Improving the reliability of the CEXP subscale increased the power of the analyses of covariance where CEXP was used as a covariate. More consistent determination of computer experience among subjects within the sample allowed more accurate analysis of regression slopes. Removal of the nine items did not significantly affect the correlation between CEXP and posttest scores ($\underline{r} = 0.63$). The initially low reliability of the CEXP scale was partially explained by the variety of item types within the scale. The improved reliability CEXP scores adjusted for response inconsistencies by removing those items that least contributed to the instrument's reliability.

CEXP scores obtained. The population mean CEXP score was 30.0 (SD = 10.9). CEXP scores appeared normally distributed, ranging from 9 to 62. For the sample (N = 38) attending training sessions, CEXP scores ranged from 9 to 58, with a group mean of 31.4 (SD = 11.0).

Group Embedded Figures Test (GEFT)

All subjects who attended training sessions completed the Group Embedded Figures Test. GEFT scores were used to measure the subjects' cognitive style (degree of field independence). Scores on the group-administered scale ranged from 0 (extreme field dependence) to 18 (extreme field independence).

The GEFT is a widely used and extensively normed instrument that measures field dependence-independence, one dimension of an individual's cognitive style. It is a 20 minute test consisting of 18 items presented in three individually timed sections. Scores for college-aged men have a mean of 12.0 (SD = 4.1) and 10.8 for women (SD = 4.2) (Cavaiani, 1989). Because the range of scores is continuous between 0 and 18, it is considered incorrect to label a subject "field-independent" or "field-dependent" (Witkin et al., 1977). Using two parallel forms of the GEFT with identical time limits, an internal consistency reliability estimate of 0.82 (for both males and females) was obtained (Mykytyn, 1989). Validity measures based on comparing results on the GEFT to results on the EFT (an individually administered form of the GEFT) show correlations of 0.62 for females and 0.82 for males (Cavaiani, 1989).

GEFT scores obtained for this sample (N = 38) ranged from 2 to 18, with a group mean of 12.4 (SD = 4.1). As expected, this sample distribution was skewed revealing a larger percentage of more field-independent individuals than would be found in a general public sample of same-aged subjects (Witkin et al., 1977). This sample distribution is similar to that reported for other college student populations (Cavaiani, 1989; Witkin et al., 1971).

Task Performance Assessment

Measurements of computer task performance were taken for each subject in the training pretest and posttest. Each test was comprised of three tasks. A performance score was calculated for each task and the sum of the scores for the three tasks in each test formed the total test performance score. The computer task performance score reflected the accuracy and rate of task completion and was calculated using Equation 6.

$$PS = \frac{ET}{AT} * P * DW \qquad (6)$$

Where:

PS = Performance score

ET = Expected time to task completion

AT = Actual time to task completion

P = Percentage of task completed

DW = Difficulty weighting factor

This method of scoring performance approximately followed the calculation used by Whiteside et al. (1985). This score quantified users' performance on tasks taking into account the rate of task completion and the competence demonstrated on the task. This formula normalized time across all tasks by dividing expected task completion time (10 minutes) by the actual task completion time. In addition, task difficulty was taken into account by using a difficulty weighting factor. This factor accounts for increasing task difficulty across a series of tasks.

Expected time to task completion (ET). This was a constant (10 minutes), determined by taking the average time subjects required for completing the tasks, then adding 20%. This value was initially based on the researcher's estimate. The final value was established based on task performance data gathered during the pilot study. This also served as the time limit allowed for attempting to complete a task. This time limit prevented subjects from spending too much time attempting a task and helped limit their frustration when no progress was being made, particularly during the pretest.

Actual time to task completion (AT). This value was computed for each task by measuring the total task completion time, beginning to end. Subjects were observed during task completion and their computer interactions were logged by an observer. Regardless whether the subject was continually on-task or not, total time to task completion was used. Subjects remained seated at the computer work station during tasks. The work station environment was controlled to minimize distractions and subjects were not allowed to take breaks between tasks during the pretest and posttest. Also, subjects were not allowed to discuss their performance or other aspects of the study until after they had completed all lessons and tests.

Percentage of task completion (P). This value reflected the degree of success on the task with respect to predetermined mastery criteria. Each task was composed of five subtasks. For every subtask completed correctly, P was increased 20%. For example, if a subject completed only four of five subtasks, that task was scored 80% complete. Subjects received P = 1.0 for every fully completed task.

Task difficulty weighting factor (DW). This value represented approximate task difficulty, which was operationally defined as the minimum number of discrete operations in the user interface required to complete the task. Examples of discrete operations included moving the mouse pointer (cursor) to an object, clicking or double-clicking with the mouse button on an object, or entering keystrokes (multiple keystrokes for a single entry were counted as one discrete operation). Three DW levels were identified to organize the 12 lessons and to design representative tasks for the pretest and posttest. Low difficulty tasks required 1 to 3 discrete operations per subtask, with a total of 8 to 9 per task. Medium difficulty tasks required 2 to 5 discrete operations per subtask, with a total of 15 to 20 per task. High difficulty tasks required 3 to 7 discrete operations per subtask, with a total of 20 to 25 per task.

Three test tasks, one task from each difficulty level, comprised the training pretest and posttest. The 12 training lessons were designed with gradually increasing difficulty, with four lessons at each of these three difficulty levels. The pretest and posttest tasks were designed as equivalent forms, so each task in the pretest had the same DW value as the corresponding task in the posttest.

By adjusting for task difficulty when computing the task performance score, a subject's competence was more accurately recorded. Subjects were expected to gain competence as they proceeded from easy to more difficult tasks. Therefore, more difficult tasks were weighted more heavily than the easier tasks.

Instructional Treatment

The instruction provided each subject was individually delivered as a short computer-based training course. After screening and aptitude assessment, the instruction consisted of an introductory videotape, followed by a computer-based interactive tutorial, followed by a pretest, 12 training lessons, and a posttest. The experimental treatment (online help format) was administered during the 12 training lessons. Subjects were randomly assigned to one of two treatment groups: (a) text-only help (TOH) or (b) text-with-motion-video help (TMVH). These two online help formats were similar in all but one respect. In the TMVH group, dynamic pictorial message elements (digital motion video windows) were added to text messages in online help, while the TOH group had identical text help messages without the dynamic pictorial elements.

Setting

The computer training sessions were conducted at an International Business Machines Corporation facility in Boca Raton, Florida. The setting was a product usability evaluation center that was ideal for this type of study. Individual subjects were seated in simulated offices--rooms equipped with computer systems, desks, tables, and other accessories typically found in corporate offices. Each subject completed the training in a single session, scheduled during normal business hours or at night, according to the student's preference. During the training each subject was situated beyond sight and hearing of others, so that he or she would not be distracted while completing tasks.

Subjects completed the training using a microcomputer equipped with a high-resolution color display and dot-matrix printer. A standard keyboard and mouse were also attached as user input devices. The software installed consisted of IBM Operating System/2 version 2.0 (OS/2), an advanced personal computer operating system with a graphical user interface (GUI). The OS/2 GUI incorporated direct-manipulation, object-oriented control features whereby application functions were controlled by manipulating graphical features with the mouse pointer, rather than relying solely on verbal menus or command-line interfaces for function selection and activation. User interaction with the system and the graphical spreadsheet application required manipulation of appropriate icons, buttons, dialog panels, windows, pull-down menus, and other user interface controls.

<u>Observers</u>

During the training, subjects were monitored by an observer situated immediately outside the simulated office. The observer could view the office interior through a one-way mirror or through remote-controlled color video cameras. The display of the computer used by the subject was attached to a second color display at the observer's console, allowing the observer to closely follow the subject's progress on each task. To help minimize the level of anxiety a subject might experience from being observed, subjects were not shown the observer's console until after the training session was completed.

One of two observers was randomly assigned to monitor each subject. Both observers had conducted observations during the pilot study so they were familiar with the training protocol and comfortable with the monitoring procedures. As the observers' role included rating subjects on task performance, inter-rater reliability was a concern. A series of t tests were run, grouping subjects by observer, to determine whether differences existed that were related to the observer assigned. Dependent variables examined included pretest and posttest scores, and time in help. No significant differences attributable to observer assignment were detected.

Instructional Sequence

The computer training protocol was delivered in five stages: orientation, introductory video tape and tutorial, pretest, lessons, and posttest. A sample of 42 subjects attended individual training sessions at the corporate product evaluation center. Most training sessions were held during normal business hours although several were held in the evening. The training periods were scheduled to last three to six hours. Actual training times ranged from 2.6 to 6.8 hours, with a mean training time of 4.1 hours.

Orientation. First, subjects were escorted into the product evaluation facility housing the simulated office where they remained throughout the training. The training observer then verbally presented an overview of the purpose of the facility and the computer training lessons. Subjects then reviewed and signed informed consent forms and nondisclosure agreements.

Introductory video tape and tutorial. The introductory video tape, Working With OS/2 Version 2 [3], provided the students with a general overview of features in the operating system and its graphical user interface. This 40-minute video tape demonstrated use of system features and defined the several different object types in the system GUI. It provided illustrative scenarios for using OS/2 applications, gave non-interactive instruction to the subjects, and augmented the instruction provided by the OS/2 system tutorial.

---

[3] Working with OS/2 Version 2 is copyright 1992 by Comsell, Inc.

Each subject was then seated at the microcomputer to complete the interactive tutorial. The system tutorial program was started and ready to use when the subject was seated. Subjects were allowed as much time as needed to complete the tutorial. This typically required 30 to 40 minutes. The system tutorial provided practice using basic and slightly more advanced operations and functions of the system, focused on using the mouse to directly manipulate objects in the GUI. Observer assistance to subjects was not provided after beginning the tutorial, except in a few cases of software failures that required observer intervention.

Training pretest. A training pretest was given immediately following completion of the system tutorial, with the subject operating the computer. The pretest included three pretest tasks, designed with increasing levels of difficulty, as described previously. The five subtasks in each pretest task were selected from subtasks found in the training lessons. Thus the pretest accurately reflects the content and design of the lessons, measuring performance on operations the subject was expected to learn during training. Subjects were allowed up to 10 minutes to complete each pretest task. Many subjects completed the tasks in less time while others were unable to complete the tasks in the allowed time. Tasks in the pretest were sequenced so that subsequent tasks could be started without requiring the preceding task to be completed. The pretest tasks were scored using the task performance formula (Equation 6) and the sum of the three pretest task scores was used as the pretest performance score.

Training lessons. After the pretest, subjects completed a series of 12 lessons, during which one of the two online help formats, TOH or TMVH, was encountered. The subjects were given a tersely worded task description as they began each lesson and were encouraged to use online help whenever they experienced difficulty or were unable to proceed. Although discouraged from requesting assistance from the observer, when such assistance was requested the subject was directed to use online help to learn more about that particular operation. The instructional materials included the printed task description for each lesson, plus the online help messages. Because each subject determined the extent to which he or she would use online help in a given lesson, the amount of instructional information viewed during training varied considerably between subjects. Use of online help was tracked automatically by system software.

Subjects were given as much time as necessary to complete each lesson. However, if an impasse was reached where no progress was made for five minutes, the subject was prompted by the observer to access help for a specified help topic. If the subject completed a lesson satisfactorily, the next lesson description was immediately provided. If an entire lesson or portions of the lesson were not completed correctly, the errors were logged and the subject was allowed to proceed to the next task. Lessons were structured to facilitate smooth transitions from one lesson to the next.

When subjects reached an obvious impasse, the observer redirected the subject to first reread the task and subtask directions. Occasionally subjects required help interpreting the directions. If the subject was still unable to proceed, the observer then prompted the subject to open the online help facility. When the subject could not locate an appropriate topic, the observer directed the subject to the specific help topic for that situation. Observers did not directly instruct subjects with procedures for task completion.

The 12 training lessons were designed with gradually increasing difficulty (four lessons each in the low difficulty, medium difficulty, and high difficulty groups). The series of 12 lessons had three objectives: First, it provided practice in elementary skills for operating the system and its graphical user interface. Second, it provided practice using the online help facilities. Third, it provided training in the use of the PM Chart graphical spreadsheet application.

Each lesson included a task description, composed of five subtasks or steps. Lessons 1 to 3 covered system tasks in the user interface, such as creating a new folder, copying several files from a diskette into a folder, and moving the PM Chart program icon into the folder. Application tasks (Lessons 4 to 12) included modifying an existing file using PM Chart features, loading spreadsheet data, and creating several presentation graphics.

The lessons were also arranged as a series of tasks within a meta-task, so there was a clear project goal the subject could perceive as each lesson was completed. The directions for the 12 computer training lessons are included in Appendix C.

Training posttest. The posttest was administered immediately following completion of the 12 training lessons. The three posttest tasks were equivalent, but not identical, to the three pretest tasks. The posttest scores were used as the primary measure of learning outcomes in this study.

Gain scores (the difference between a subject's performance on the pretest and posttest) were calculated for each subject. Although commonly applied in educational research, use of gain scores has been criticized on the basis of generally poor reliability (Stevens, 1990). Specifically, when the correlation of pretest and posttest scores approaches the reliability of the test, the reliability of gain scores goes to zero. For this reason, gain scores were not used as dependent measures in the analysis.

Experimental Instructional Variable

The two treatment conditions (online help formats) were identical except for one variable: the presence of dynamic pictorial elements in the spreadsheet application online help messages. Online help provided instruction for subjects as they attempted to learn application functions during lessons 4 to 12. The two online help formats are described below, along with characteristics of online help common to both treatments.

General online help characteristics. The online help facility in the system, the information presentation facility, presented help messages displayed within windows adjacent to or overlapping the application windows. The help messages consisted of text formatted as paragraphs, sentences, and lists. Dynamic pictorial elements (digital motion video sequences) could be displayed, in addition to the text content, in the experimental treatment condition. Direct-manipulation window controls were provided so the help window could be moved, resized, and closed at any time at the discretion of the user. Figure 3-1 illustrates the OS/2 application help window interface for the graphical spreadsheet application used in this study.

All help windows incorporated standard controls that allowed the user to access additional functions, such as printing a help topic, viewing the help index, and moving forward or backward through selected help topics. The sequence of information displayed at any time in the help window was controlled in an interactive manner. The user could select a help topic and then change the topic at any time. Help messages often displayed related topic labels, called links. Links appeared as text displayed in a different color (green) than standard help text (blue). The related topics could thus

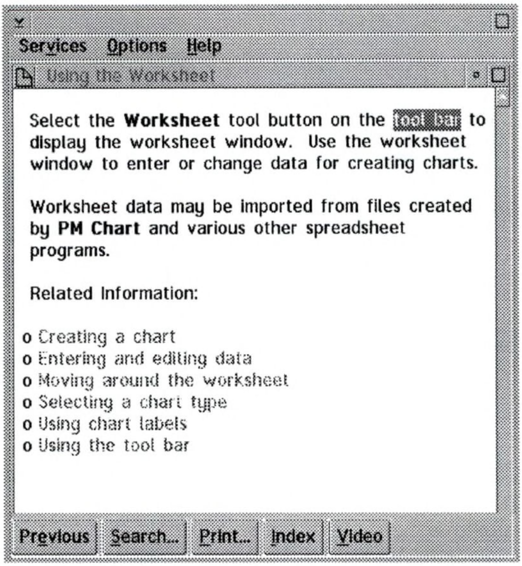

Figure 3-1. OS/2 help window interface used in PM Chart, showing video button.

easily be accessed by double-clicking on the desired link. The selected help topic would then be displayed in the help window, overlaying the previous help topic in the help window. The user could also interact with the application at any time without closing the help window. When in use, the entire help window remained visible, occasionally covering a small portion of the application window.

The information presentation facility in this system provided a hypertext implementation that supported selective access to information structured in a nonlinear manner. Subjects were able to select a series of related help topics, when desired, simply by repeatedly selecting links in the displayed help windows. When the desired help information had been viewed, the subjects closed the help window and returned to the application to complete the task at hand.

Text-only help (TOH) treatment. This treatment provided instruction in online help messages that included only text (verbal-digital) message elements relating to the selected help topic. No graphical or pictorial message elements were included in these help messages. The text contained in the help topics was carefully edited to fully describe application functions. In many help messages, the text extended beyond the window borders. For these messages, the user had to scroll within the help window to read the complete help topic text. Subjects' use of online help was automatically tracked using system software to record the total time in help and the topics displayed.

Text-with-motion-video help (TMVH) treatment. This online help format incorporated the same text messages as the TOH treatment. In addition, a Video button was added to the help window controls. An example of a help window showing the location of the Video button is shown in Figure 3-1. When a subject selected the Video button, a motion video sequence was displayed in a window overlaying the previously visible help text. This overlay technique was identical to how related help topic text windows appeared whenever text links were selected. When a motion video sequence had ended, the final video frame remained visible as a static image until that window was closed. Display of these video sequences by subjects was tracked in the same manner as the display of text help topics.

The digital motion video playback facilities employed in this study included Digital Video Interactive[4] (DVI) hardware and software features. Additional customized software was developed to provide an interface between the OS/2 information presentation facility and DVI. DVI playback at 12 frames-per-second was used to present the dynamic pictorial sequences that visually portrayed application functions described in the help text. Each video sequence, lasting from 15 to 40 seconds, was designed to match a specific help topic's text message. Each video sequence was produced using an 8mm video camera aimed directly at an active high resolution computer display. The camera's S-video output signal was connected directly into the DVI capture adapter input, so that the camera output could be captured without loss of signal quality. The camera remained stationary during each sequence, usually tightly cropping the PM Chart application window being manipulated according to the accompanying help text. This approach follows from instructional design principles based on dual-coding theory (Fleming & Levie, 1978).

Although professional grade video equipment was used to develop the video sequences, conversion to the digital display format in this system induced some loss of visual detail in the image. Several subjects commented that the images were "blurry" but maintained they could understand and follow the video sequences. Visual quality of the digital video sequences also was reduced by limiting the size of the playback window to the help topic window size. The video images thus displayed showed application features smaller than they appeared in the application interface itself. Despite these visual quality issues, subjects assigned to the TMVH treatment frequently stated their preference for the video help format.

---

[4] Digital Video Interactive and DVI are trademarks of Intel Corporation.

Data Collection

Data was gathered in several ways for this study. Scores on the aptitude measures of interest were obtained by hand-scoring the test answer forms for the GEFT and CECS instruments. During training sessions observers manually logged the actions of subjects, while at the same time the computer automatically logged online help activity. These techniques are described below.

Automated Online Help Tracking

Subjects were instructed to use online help whenever they were uncertain of how to proceed with a task. Instructions for using online help were repeated several times throughout the training lessons. Custom computer software was used to automatically collect data on the use of online help. This software consisted of a help tracking program that automatically created a log file for each subject containing accurate timing data on the subject's use of online help. Each time a help topic was opened, the time in help for that topic was recorded in the log file. The total time in help (THELP) was calculated for each subject and used as a dependent variable in the analyses of covariance described below.

The help tracking program ran in the background (not visible to the student) as the subject completed the training lessons. For each help topic opened, the log contained the topic name, the time the topic window was opened and closed, and the elapsed time for each topic. Total time in help was computed as the sum of the elapsed times. For students in the TMVH treatment, the use of video help segments was also recorded. The elapsed time for the video segments opened by the student was included in the total time in help. A sample help tracking log is included in Appendix D.

Because help windows could remain open while a subject was interacting with the application, the observer also kept records of help usage. The observer was responsible to log any occasion where a subject left a help window open when interacting with the application. Only two of the 38 subjects had used help in this manner. For these subjects, their time in help scores were adjusted to accurately reflect when they were using help and when they were interacting with the application.

Observer Event Logging

Subjects were observed by the researcher, who sat at a control panel in an adjacent room and recorded significant events in a computer database. During the training, the observer continually updated the database by adding records of actions taken by the subject and, occasionally, by the observer. Each entry in the log was automatically time-stamped to facilitate accurate timing of the subject's actions. For the three tasks in the pretest and posttest, log entries indicated success or failure for each subtask and how long it took to complete each task. The log files were later examined to calculate the pretest and posttest performance scores. A sample event log is included in Appendix E.

Summary

This experimental study of learning computer operations in graphical human-computer interfaces was designed to test the effects of university students' cognitive styles (field dependence-independence) and prior computer experience on their performance on tasks in a direct-manipulation, graphical user interface. The population pool completed assessments of field dependence-independence and prior computer experience. Volunteer subjects randomly selected from the population were scheduled for individual training sessions. The subjects were randomly assigned to one of two treatment conditions that differed only with regard to presence of dynamic pictorial elements (digital motion video) in the online help messages.

After an introductory videotape and an interactive computer-based tutorial, all subjects then completed an initial computer operations pretest to establish a performance baseline. The subjects completed a series of 12 lessons comprising tasks using the computer system's graphical user interface and a graphical spreadsheet application. Task performance was again measured in a posttest at the completion of the training lessons. A completely randomized design with pretest-treatment-posttest sequence was employed in this study. A multiple covariance analysis was used to determine whether ATI effects occurred between field independence and treatment, or between computer experience and treatment. The ANCOVA was also used to detect effects of the two aptitude measures on performance. The ANCOVA techniques were further employed to control for between subjects variance on time in help. The results of the study are described in the following chapter.

CHAPTER 4
RESULTS AND ANALYSIS

Introduction

This study was conducted to determine whether field dependence-independence or level of computer experience influenced computer users' performance on application tasks in a direct-manipulation graphical user interface (GUI). Other research questions addressed in this research concerned whether Aptitude x Treatment interactions occurred between field independence or computer experience and the presence of dynamic pictorial message elements displayed in online help. A random sample of 38 university student volunteers attended individual computer-based training sessions. The students were randomly assigned to one of two treatment groups using different online help formats. Both treatment conditions required subjects to use online help to obtain instruction for learning application functions. Text-only help was provided in one treatment level, while in the other level dynamic pictorial content (digital motion video) was displayed in addition to help text.

A fully randomized design with pretest-treatment-posttest sequence was used in the study. Data were analyzed using a multiple covariance analysis (Huitema, 1980). Field independence and computer expertise were employed as covariates in the ANCOVA. Time in help was also included as a covariate to control for between subjects variance on exposure to help messages. The dependent measure was performance on application tasks in the training posttest.

No significant interaction effects were found between any of the three covariates and the online help format. In addition, no significant effect was found for the online help format. Significant regression effects were found for both computer experience and field independence on application task performance. Increased prior computer experience and increased field independence were significantly related to improvements in application task performance on the posttest. No significant regression effect was found for time in help. These results are described in detail in this chapter.

Results

Data were collected during the experiment and analyzed as described in the preceding chapter. This section presents the ANCOVA results obtained using the Statistical Analysis System (SAS), release 6.07.

The one-way ANCOVA model for this analysis included the treatment variable and three covariates. The treatment factor (TREAT) consisted of two levels, text-only help (TOH) and text-with-motion-video help (TMVH). The covariates included computer experience (CEXP), field dependence-independence (FDI), and time in help (THELP). The single dependent variable was application task posttest performance (POST). The 38 students in the sample were randomly assigned to one of the two treatment groups. There were 18 subjects in the TOH group and 20 subjects in the TMVH group. The results of the ANCOVA are described below, beginning with a review of the null hypotheses tested.

Treatment x Covariate Interaction Effects

First, the ANCOVA assumption of homogeneous regression slopes was tested. This was also a test for interactions between the covariates and the treatment factor. Therefore, the test for homogeneous slopes tested the following null hypotheses regarding interactions: Hypothesis 1, that no significant differences in application task performance would result from a three-way interaction among field dependence-independence, prior computer experience, and the presence of dynamic pictorial message content in online help; Hypothesis 2, that no significant differences in application task performance would result from an interaction between prior computer experience and the presence of dynamic pictorial message content in online help; and Hypothesis 3, that no significant differences in application task performance would result from an interaction between field dependence-independence and the presence of dynamic pictorial message content in online help.

As described in Chapter 3, the test for homogeneous slopes required computing the error sums of squares for two linear regression models, referred to as the complete and reduced ANCOVA models. Summary source tables for the complete and reduced ANCOVA models are given in Tables 4-1 and 4-2. The $\underline{F}$ statistic to test the assumption of homogeneous regression slopes was calculated using the error sums of squares for these two models. The resulting test statistic, $\underline{F}\ (3, 30) = 1.14$, $\underline{p} > .05$, did not reach significance. The assumption of homogeneous regression slopes had been met. Therefore, the null hypotheses regarding interactions between the covariates and the treatment factor (Hypotheses 1, 2 and 3) were not rejected. No significant interaction effects on posttest performance were detected between online help format and computer experience, field dependence-independence, or time in help.

Treatment Effect

Since the assumption of homogeneous slopes was valid for this analysis, the next step in the ANCOVA was to determine whether treatment differences had a significant effect on posttest performance. This tested Hypothesis 4, that no significant differences in performance on computer application tasks would exist between subjects viewing text-only online help and subjects viewing online help containing text and dynamic pictorial elements.

Testing the treatment effect required computation of a third regression model, the reduced ANCOVA model without the treatment effect. The summary table shown in Table 4-3 identifies the sources of variance for the reduced model with the treatment effect removed. This model determined the regression effects for the three covariates, assuming the treatment variable had no effect.

Table 4-1 Summary Table for Complete ANCOVA Model Effects on Posttest Scores

Source	df	SS	F	Pr > F
Model	7	89442.35	6.25	0.0001
Error	30	61313.78		
Corrected Total	37	150756.13		
Source	df	Type III SS	F	Pr > F
TREAT	1	12.02	0.01	0.9394
CEXP	1	26451.12	12.94	0.0011
FDI	1	13391.26	6.55	0.0158
THELP	1	6608.32	3.23	0.0822
CEXP*TREAT	1	4896.28	2.40	0.1322
FDI*TREAT	1	2816.04	1.38	0.2497
THELP*TREAT	1	509.35	0.25	0.6213

Table 4-2 Summary Table for Reduced ANCOVA Model With Treatment Factor

Source	df	SS	F	Pr > F
Model	4	82449.42	9.96	0.0001
Error	33	68306.72		
Corrected Total	37	150756.13		
Source	df	Type III SS	F	Pr > F
CEXP	1	27063.47	13.07	0.0010
FDI	1	11437.64	5 .53	0.0249
THELP	1	6682.56	3.23	0.0815
TREAT	1	1266.57	0.61	0.4397

The test of significant treatment effect was performed by calculating the appropriate F statistic, supplying the error sum of squares for the reduced model without the treatment effect (from Table 4-3), and the error sum of squares for the reduced model including the treatment effect (from Table 4-2).

Table 4-3 Summary Table for Reduced Model Without Treatment Factor

Source	df	SS	F	Pr > F
Model	3	81182.84	13.22	0.0001
Error	34	69573.29		
Corrected Total	37	150756.13		
Source	df	Type III SS	F	Pr > F
CEXP	1	26849.40	13.12	0.0009
FDI	1	10192.35	4.98	0.0323
THELP	1	8541.05	4.17	0.0489

The resulting test statistic, $F\ (1, 33) = 0.61$, $p > .05$, did not reach significance. Therefore, Hypothesis 4 was not rejected. This test statistically controlled for the between-subjects variance on the covariates and determined that the adjusted group means on POST between the two online help treatments were not significantly different. The addition of dynamic pictorial elements to textual online help in this study did not significantly effect performance on application tasks in the unfamiliar GUI.

Regression Effects

After determining that the interaction and treatment effects were not significant, the analysis proceeded to test for significance of regression effects for each of the covariates. These effects were determined using the Type III sums of squares found in Table 4-2. This table shows the corresponding F statistics and probabilities for the regression effects of the covariates CEXP, FDI, and THELP.

First, the a priori assumption that time in help would have no significant effect on performance after controlling for differences on FDI and CEXP was upheld. The test statistic for regression of POST on THELP, $F\ (1, 37) = 3.23$, $p > .05$, did not reach significance. Increasing use of help, measured as the total time a user displayed help messages during training, was not significantly related to performance on posttest tasks.

Next, the regression effects of the aptitude variables, CEXP and FDI, were examined. The appropriate F statistics were examined to test the research hypotheses. Hypothesis 5 stated that no significant relationship would exist between prior computer experience and a computer user's performance on computer application tasks in an unfamiliar GUI. The test for regression effect of POST on CEXP, $F(1,37) = 13.07$, $p = 0.001$, revealed a significant effect. Therefore, the null hypothesis was rejected. Prior computer experience, as measured using the Computer Experience and Competence Survey, was significantly related to performance on the posttest tasks. As the level of prior computer experience increased, performance on application tasks was found to improve.

Hypothesis 6 stated that no significant relationship would exist between field dependence-independence and a computer user's performance on computer application tasks in an unfamiliar GUI. The test statistic for regression of POST on FDI, $F(1, 37) = 5.53$, $p = 0.025$, showed a significant effect. Therefore the null hypothesis was rejected. Field dependence-independence, as measured using the Group Embedded Figures Test, was significantly related to performance on the application posttest tasks. As field independence increased, there was a significant tendency for performance on application tasks to improve.

In this analysis, significant regression effects were found for computer experience and field dependence-independence. No significant treatment effect was found for adding dynamic pictorial elements to online help displays. In addition, no significant Aptitude x Treatment interactions were found. The following analysis of these results examines the significant regression effects as well as the character of the nonsignificant interaction and treatment effects.

Analysis

The multiple covariance analysis found no significant interaction effects between online help format and any of the covariates. Also, no significant main effect was found for online help format. Significant regression effects were found for both computer experience and field dependence-independence. This section presents an analysis of the regression slopes for the ANCOVA model as they were examined at each step in the procedure. This analysis begins with a review of the tests performed to verify that assumptions for ANCOVA had been met.

Testing ANCOVA Assumptions

Homogeneity of variance. To test this assumption, t tests were performed to determine whether the mean scores and within-group variance for CEXP, FDI, and THELP were significantly different between the two treatment levels. No significant differences ($p > .05$) were found between the treatment group means. Tests of unequal group variance for CEXP and THELP did not reach significance. A test for unequal group variance did reach significance ($p = .04$) for the FDI scores. However, homogeneity of variance is not required when the covariate is statistically independent of the treatment (Huitema, 1980). Since FDI scores were obtained prior to the training, this assumption was met for this analysis.

Independence of treatment and covariates. Since computer experience and field dependence-independence were measured prior to the instructional treatment, these covariates were measured independently. Time in help (THELP) was measured during the 12 training lessons when the online help displays were being used. An ANOVA on THELP was computed to verify that it was independent of the online help format. The resulting test statistic, $F = 1.21$, $p = 0.2796$, did not reach significance. There was no significant effect of treatment level on time in help, so this assumption was also met.

Homogeneity of regression slopes. This assumption was tested in conjunction with the test of hypotheses concerning interaction between the treatment factor and the covariates. No significant Covariate x Treatment interaction effects were found, so this assumption was valid. Additional analysis was performed concerning between-group regression slope differences, as described in the following section.

Covariate x Treatment Interactions

There were no significant Covariate x Treatment interactions. Although these interaction effects did not reach significance, the regression slopes for the two treatment groups were plotted. Cronbach and Snow (1977) recommended that even nonsignificant interactions should be examined, particularly when the number of subjects in each treatment group is much smaller than 100. In taking this position they stated: "Consistent nonsigificant results are at least as valuable to a science as are incoherent significant results" (p. 53).

The complete ANCOVA model (see Equation 1 in Chapter 3) includes a $\underline{Y}$-intercept parameter (α), regression slope parameters (β_i) for each covariate (X_i), cross-product term coefficients (γ), and a $\underline{Y}$-intercept difference parameter (δ) for the effect of treatment (A). Since there were only two treatment groups, only one δ is required for this model. The estimated values of these regression equation parameters as computed by the SAS GLM procedure are shown in Table 4-4. These regression equation parameters were used to plot regression lines to illustrate the nonsignificant ANCOVA interactions. Regression line pairs for each covariate were plotted in three separate two-dimensional graphs. These illustrations allow visual inspection of the nonsignificant two-way interaction effects. The nonsignificant three-way interaction (CEXP x FDI x TREAT) was illustrated by plotting two regression planes in a three-dimensional graph.

The nonsignificant interaction effects between treatment (online help format) and the three covariates are depicted in Figures 4-1 to 4-3. The regression of POST on computer experience for both levels of online help treatment is shown in Figure 4-1. The regression of POST on field dependence-independence for the two treatment conditions is depicted in Figure 4-2. The regression of POST on time in help for both treatment levels is shown in Figure 4-3. The regression planes formed by the intersection of CEXP and FDI regression slopes are shown in a three-dimensional graph

Table 4-4 Parameter Estimates for Complete ANCOVA Model Regression Effects

Parameter		Estimate
Y-Intercept	(α)	21.81
FDI*TREAT	(γ_1)	-5.03
FDI	(β_1)	8.00
CEXP*TREAT	(γ_2)	2.32
CEXP	(β_2)	1.54
THELP*TREAT	(γ_3)	0.93
THELP	(β_3)	-2.14
TREAT	(δ)	-6.02

in Figure 4-4. This three-dimensional illustration permits a visual inspection of the nonsignificant three-way interaction between computer experience, field dependence-independence, and online help format.

As illustrated in Figure 4-1, the nonsignificant interaction between computer experience and online help format shows a trend for subjects in the text-only help condition to perform better than subjects in the text-with-motion-video help condition. The between-groups performance difference tends to increase with increasing prior experience. This nonsignificant ATI effect reflects that in this study text help with dynamic pictorial elements were somewhat less helpful than text-only help, particularly for individuals who had more extensive computer experience.

The nonsignificant trend toward an interaction between field dependence-independence and online help format is shown in Figure 4-2. Visual inspection of the regression lines for the two treatment groups reveals that as field independence increased, subjects in the text-with-motion-video online help format tended to score

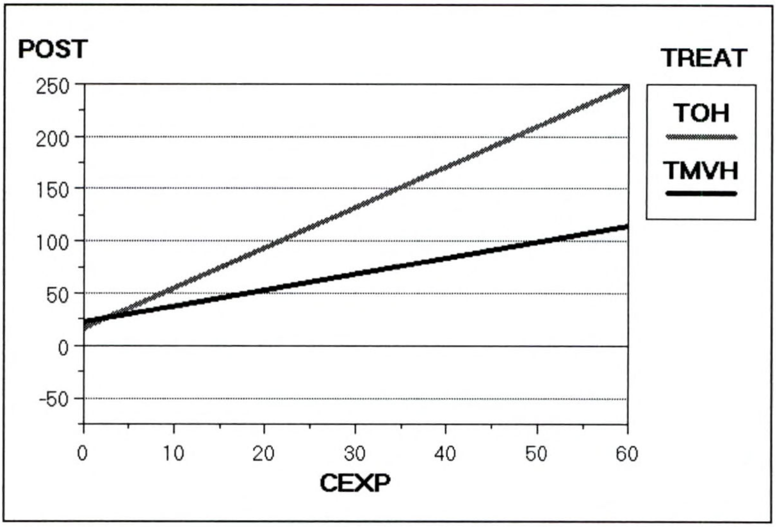

Figure 4-1. Nonsignificant CEXP x TREAT interaction.

higher on the posttest than highly field-independent subjects in the text-only group. While this interaction effect did not achieve significance, the trend indicates that in this study the performance of extremely field-independent subjects was higher when dynamic pictorial presentations appeared in the online help messages. The task performance of the most field-dependent individuals, on the other hand, appeared to be the same regardless of the online help format used.

As shown in Figure 4-3, the effect of time in help was very small (the slopes of the regression lines are approximately zero) and the slopes of the regression lines for the two treatment levels were nearly identical. There was no clear indication of a trend toward an interaction between time in help and online help format. Regardless of how much time subjects spent using help, the posttest performance score difference between the two treatment groups remained very small.

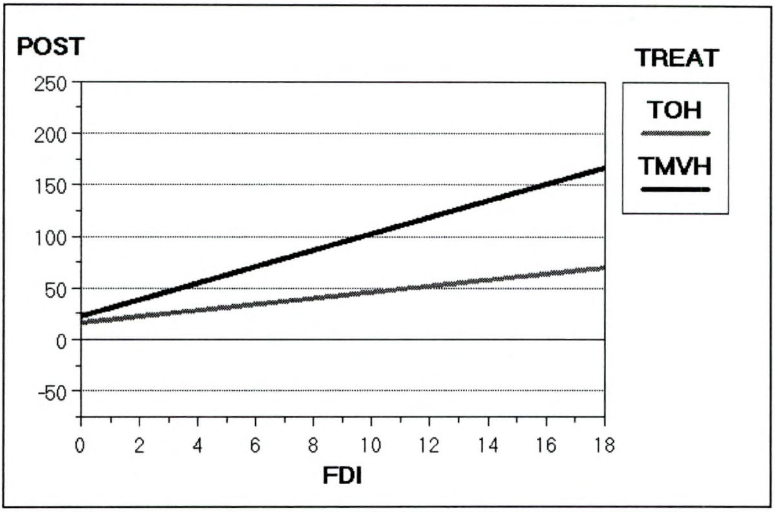

Figure 4-2. Nonsignificant FDI x TREAT interaction.

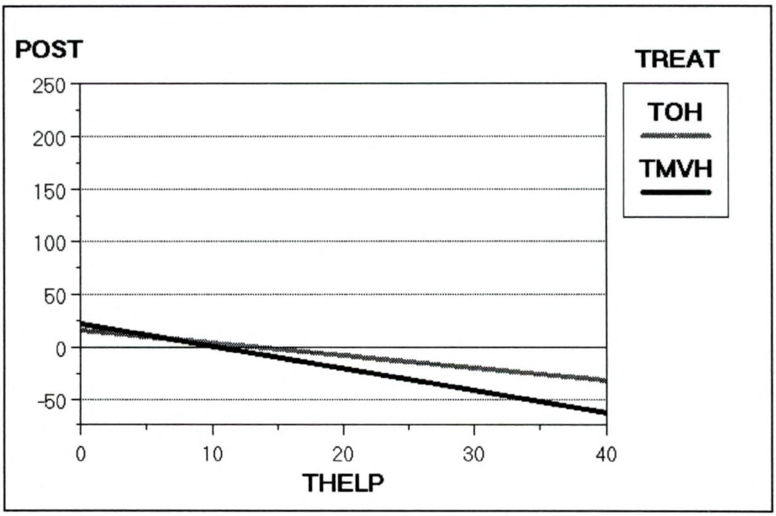

Figure 4-3. Nonsignificant THELP x TREAT interaction.

The nonsignificant three-way interaction between computer experience, field dependence-independence, and online help format is depicted in Figure 4-4. This three-dimensional plot of two regression planes reveals a trend towards an interaction. The lightly shaded plane, representing the predicted posttest scores for the text-with-motion-video help (TMVH) treatment group, shows the combined effects of increasing computer experience and increasing field independence. Among the very field-independent computer experts (FDI = 18, CEXP = 60) in this group, the predicted posttest score was 279.8. For extremely field-dependent computer experts (FDI = 0, CEXP = 60) in the TMVH group, the predicted posttest score was 114.0.

The more heavily shaded plane in Figure 4-4 represents the predicted posttest scores for the text-only help (TOH) treatment group. In this treatment group, very field-independent computer experts (FDI = 18, CEXP = 60) had predicted posttest scores of 300.6, only slightly higher than in the TMVH group. However, for extremely field-dependent computer experts (FDI = 0, CEXP = 60) the predicted posttest score for the TOH group was 247.1, much higher than the predicted score for the TMVH group, 114.0. This contrast shows that the presence of dynamic pictorials in online help was related to a negative effect on performance for highly field-dependent computer experts. For field-independent computer experts, however, there appeared to be no significant application task performance difference between the two online help treatment groups.

The two regression planes depicted in Figure 4-4 reveal another aspect of the trend towards a CEXP x FDI x TREAT interaction. For computer novices with high field-independence (CEXP = 0, FDI = 18), the predicted posttest score for the TMVH group was 165.8, while for similar individuals in the TOH group, the predicted score was 69.3. For field-independent novices, therefore, the presence of dynamic pictorials was associated with an increase in performance on GUI application tasks.

In the regression plots shown in Figures 4-1 and 4-2, there is visible evidence of trends towards both a CEXP x TREAT interaction and a FDI x TREAT interaction.

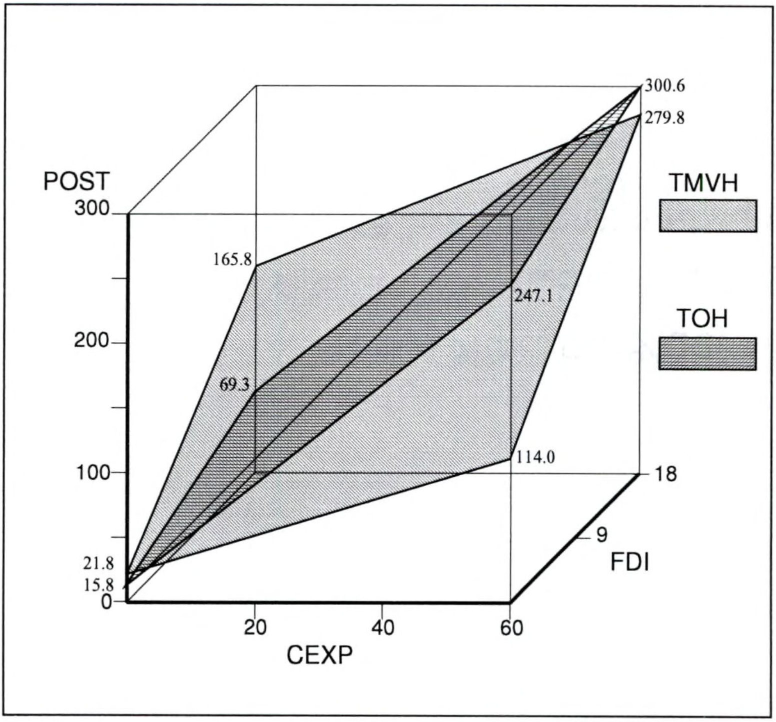

Figure 4-4. Three-dimensional plot of two regression planes showing nonsignificant CEXP x FDI x TREAT interaction.

In addition, as shown in Figure 4-4, there was visible evidence of a trend towards a CEXP x FDI x TREAT interaction. However, these Aptitude x Treatment interactions were nonsignificant. Therefore, these interaction effects were ignored in the remaining steps of the multiple covariance analysis. In the following discussion of treatment effects, the regression slopes for the two treatment groups were assumed to be equal.

Treatment Effects

Since there were no significant interactions and the assumption of homogeneous regression slopes was valid, the analysis of treatment effects examined the adjusted posttest performance means for the two treatment groups. The between-groups difference on POST adjusted means was characterized by the difference on the Y-intercept (δ), calculated using the reduced ANCOVA model. This reduced model included the covariates and the treatment factor but eliminates the interaction terms.

The regression coefficient estimates calculated using the reduced ANCOVA model are given in Table 4-5. The between-groups POST adjusted means difference attributable to the online help treatment is 12.31. The range of scores on POST was from a minimum of 53.2 to a maximum of 273.7 (SD = 63.83). Given this distribution, the between-groups difference on POST--independent of all covariates--was remarkably small.

Controlling for the effects of the covariates, the reduced ANCOVA model produced an estimate of the treatment effect on POST that was nonsignificant at the .05 alpha level. The very small between-groups difference on POST ($\delta = 12.31$, 0.19 SD) indicated that--independent of computer experience, field dependence-independence, and time in help--the effect of online help format on application task performance in this study was negligible.

Covariate Regression Effects

The final step in the ANCOVA hierarchical regression analysis was to examine the regression of the dependent variable on the covariates. In the prior steps, interaction and treatment effects were found to be nonsignificant. Removing the interaction and treatment terms from the ANCOVA model resulted in a multiple regression prediction equation. The regression coefficient estimates for this equation are given in Table 4-6.

Table 4-5 Parameter Estimates for Reduced ANCOVA Model Regression Effects with Nonsignificant Treatment

Parameter		Estimate
Y-Intercept	(α)	23.92
FDI	(β_1)	4.67
CEXP	(β_2)	2.72
THELP	(β_3)	-1.67
TREAT	(δ)	12.31

Using these estimates, predicted posttest performance scores were obtained. The coefficient of multiple correlation for this regression equation, R = .733, was reasonably high. About 53.9 percent of the variance on posttest performance was accounted for by these three covariates.

The regression effects of POST on CEXP, and of POST on FDI, were found to be significant. The regression slope of POST on CEXP is shown in Figure 4-5. Examining only the effect of computer experience, the predicted posttest performance scores ranged from 37.40 (CEXP = 0) to 194.71 (CEXP = 58). The regression effect of CEXP, expressed in terms of the sample variance on POST, was 2.46 SD.

Table 4-6 Regression Equation Parameter Estimates for Reduced ANCOVA Model

Parameter		Estimate
Y-Intercept	(α)	37.40
FDI	(β_1)	4.21
CEXP	(β_2)	2.71
THELP	(β_3)	-1.84

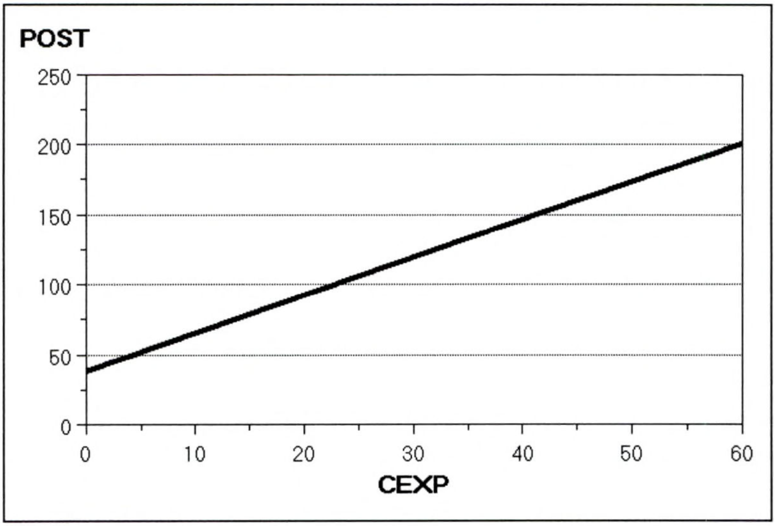

Figure 4-5. Regression effect of posttest task performance on computer experience.

The regression slope of POST on FDI is shown in Figure 4-6. The predicted POST scores ranged from 37.40 (FDI = 0) to 113.24 (FDI = 18). The regression effect of FDI, again standardized on the sample variance on POST, was 1.19 SD.

By comparing these regression contributions, or beta weights as they are often referred to in multiple regression analyses, prior computer experience accounted for more than twice the posttest variance that was accounted for by field dependence-independence. Time in help accounted for slightly less posttest score variance, with a beta weight of only 1.16 SD. With either increasing computer experience or higher field independence, subjects' posttest task performance improved significantly, regardless of online help format. Increased time in help was related to a performance decline on the posttest. Computer experience accounted for more than twice the effect of field dependence-independence.

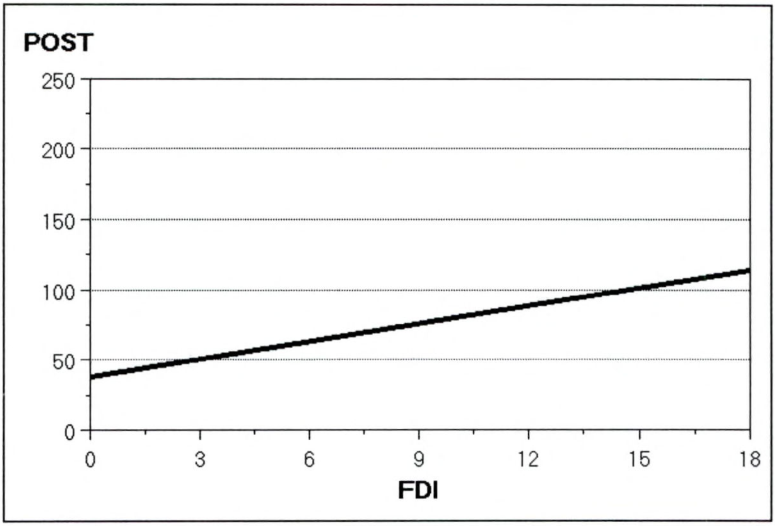

Figure 4-6. Regression effect of posttest task performance on field dependence-independence.

Summary

This study was conducted using a fully randomized experimental design with pretest-treatment-posttest sequence. The sample of 38 subjects were randomly assigned to one of two online help treatment groups. A multiple covariance analysis was applied with computer experience, field dependence-independence, and time in help as three covariates and posttest task performance as the dependent measure.

No significant Covariate x Treatment interactions were detected. Although the interaction between field independence and online help format was not significant, a trend towards an interaction was revealed. Highly field-independent subjects had higher task performance in the text-with-motion-video help treatment. A trend towards a Computer Experience x Treatment interaction was also observed. Subjects with very high computer experience performed better in the text-only help treatment. Finally, a trend towards a three-way interaction was also revealed. Very field-independent computer novices had higher task performance in the text-with-motion-video help format, while highly field-dependent computer experts performed better in the text-only help treatment.

There was no significant effect resulting solely from the online help format. When considered independent of computer experience, field dependence-independence, and time in help, the use of dynamic pictorial elements in online help messages had a negligible effect on application task performance.

Increased computer experience and greater field independence were significantly and independently related to improved performance on application tasks in the unfamiliar graphical user interface. Time in help was not significantly related to application task performance, when controlling for differences in computer experience and field independence.

The results of this study may contribute to our understanding of how computer users learn to operate advanced graphical applications. When such learning involves presentation of information via online help, knowledge of these aptitude effects can help designers improve the design of online help information, particularly with respect to the use of pictorial message elements. The implications of this study for the design of online help systems, and for future research in this area, are discussed in the next chapter.

CHAPTER 5
DISCUSSION AND RECOMMENDATIONS

Introduction

The purpose of this study was to examine the effects of field independence and computer experience on computer users learning application functions in a graphical user interface (GUI). Data were analyzed using a multiple covariance analysis. Analysis of application task performance detected significant regression effects for field independence and for computer experience. No significant Aptitude x Treatment interaction (ATI) effects were detected, although trends towards such interactions were evident. Specific results of this study are discussed below with respect to the research questions addressed. The significance of the results are then described within the context of the theories upon which this study was founded.

The implications of these results for future research and development of online help systems and other aspects of human-computer interface design are also presented here. Emphasis has been placed on recommendations for improving the design of online help systems to accommodate differences in users' cognitive styles and prior computer experience. The application of these findings to the design of adaptive computer-based instructional systems is discussed, and several recommendations for additional studies are presented.

Discussion of Findings

Each of the research questions posed for this study are examined in this section based on the results and analysis presented in the preceding chapter, and subject to the limitations within which this study was conducted. The limitations for this study included the composition of the target population, the sample size obtained, the nature of the computer system and spreadsheet application involved, characteristics of the dynamic pictorial elements used in the experimental treatment, and the instructional design employed.

The null hypotheses tested in this study are examined below citing the significant findings of the analysis and relevant aspects of the data collected. These hypotheses were stated pertaining to a learning situation where computer application online help messages were displayed in a GUI, where the GUI was unfamiliar to the users, and where the instruction was systematically varied by adding dynamic pictorial elements to text-based help displays.

Hypothesis 1. No significant differences in application task performance result from a three-way interaction among field dependence-independence, prior computer experience, and the presence of dynamic pictorial message content in online help. This hypothesis was not rejected.

The ANCOVA did not detect a significant three-way interaction effect. A trend towards an interaction was evident, however, from a visual inspection of a plot of the regression planes (see Figure 4-4). The presence of dynamic pictorials in online help was related to a decline in application task performance for the most field-dependent computer experts, while for field-independent computer novices the presence of dynamic pictorials was associated with an increase in performance.

Limited power in the analysis inhibited detection of significant interaction effects. The sample size obtained for this study was small ($N = 38$). Cronbach and Snow (1977) recommended that ATI studies incorporate samples with at least 100 subjects per group. The lack of significance of the interaction effects in this study should be viewed in light of the small sample size. The trends toward interactions observed in this analysis are evidence of effects that require further study.

Hypothesis 2. No significant differences in application task performance result from an interaction between prior computer experience and the presence of dynamic pictorial message content in online help. This null hypothesis was not rejected.

A test for an interaction effect between computer experience and the instructional treatment on task performance--when controlling for individual differences on field independence and time in help--did not reach significance. However, when the predicted posttest performance scores were plotted for the two treatment groups against computer experience, a trend towards an interaction effect was evident (see Figure 4-1). An examination of this interaction plot revealed that students with the most computer experience tended to perform better in the text-only help treatment than in the text-with-motion-video help treatment. For students with low levels of computer experience there was negligible performance difference between the treatment groups.

This interaction trend, although not significant, indicates that highly experienced users may perform better on GUI application tasks when provided with text-only online help than when help also includes dynamic pictorials. This assertion, however, should be viewed in the most tentative manner. This trend towards an interaction effect requires further investigation.

Hypothesis 3. No significant differences in application task performance result from an interaction between field dependence-independence and the presence of dynamic pictorial message content in online help. This hypothesis was not rejected.

The ANCOVA failed to detect a significant interaction effect between field independence and treatment on posttest performance scores. An illustration of this nonsignificant interaction effect, depicted in Figure 4-2, revealed different regression slopes on posttest performance for the two treatment groups as field dependence-independence varied. First, performance increased as field independence increased, regardless of treatment. The increase in performance was greater when help messages included dynamic pictorial elements than when help messages were text-only. This interaction trend indicated that as field independence increased, greater benefit was gained from the motion video content. Although nonsignificant, this trend toward an interaction was noteworthy in this study, particularly in light of the very small sample size obtained.

This interaction trend is consistent with Witkin's field independence theory. Individuals with higher field independence are better able to internalize and comprehend the structure of visually complex stimuli (Witkin et al., 1971). Since the motion video help displays used in this study were visually complex, the students who were most field-independent were able to benefit most from them, while the more field-dependent students benefited less. This interaction trend, while not significant, warrants further investigation.

Hypothesis 4. No significant differences in performance on computer application tasks exist between subjects viewing text-only online help and subjects viewing online help containing text and dynamic pictorial elements. This hypothesis was not rejected.

There was no significant difference between treatment groups on posttest performance when controlling for between-subjects variance on computer experience, field dependence-independence and time in help. The addition of dynamic pictorial message elements to online help had no detectable effect on learning the spreadsheet application tasks.

The absence of a treatment main effect was not unexpected in this study. The lack of a treatment effect appeared to contradict Paivio's dual-coding theory applied to online help message design. There appeared to have been no positive impact on application task performance resulting from the addition of motion video displays in help. There were, however, several mitigating experimental design factors that may have diminished the instructional benefits of dynamic pictorial elements. First, a primary instructional design characteristic of online help is that it is learner-controlled.

Computer-based instructional designs that are learner-controlled have been found to be less effective than program-controlled designs (McNeil & Nelson, 1991). Because the subjects in this study controlled their viewing of online help and dynamic pictorials, control of exposure to the instructional treatments was limited. This is a problem inherent to all studies involving learner-controlled computer-based instruction. Second, subjects in both treatment groups selected and viewed an average of about nine help topics. On average, students displayed help messages for less than five percent of the total training time. The exposure to the experimental treatment was therefore relatively brief. Third, the clarity of the motion video images was reduced by the compression-decompression methods inherent in the digital video technology employed. This may also have reduced the effectiveness of the pictorial sequences. Finally, since the GUI and the application were unfamiliar, and the video segments presented images captured from that interface, the pictorial displays contained visually unfamiliar--and perhaps unrecognizable--interface features. These four methodology factors may have diminished the potential instructional benefits of the dynamic pictorial message content. The potential for dynamic pictorial elements to contribute to learning from online help should not be entirely disregarded on the basis of this nonsigificant finding.

Hypothesis 5. No significant relationship exists between prior computer experience and a computer user's performance on computer application tasks in an unfamiliar GUI. This null hypothesis was rejected.

A significant regression effect for computer experience (CEXP) on posttest performance (POST) was detected. This effect demonstrated the significant relationship between computer experience and application task performance when individual differences of field dependence-independence and time in help were controlled by using these variables as covariates.

Computer experience, as measured using the Computer Experience and Competence Survey, proved to be a useful predictor of success for the computer-based training implemented in this study. When learning graphical application functions in an unfamiliar GUI, where instruction was provided in an online help environment, individuals with extensive computer experience would be expected to perform significantly better on application tasks than would individuals who had little prior computer experience. Importantly, the precise nature of prior computer experience need not be determined. The CEXP score, a general measure of prior experience, was found to be significantly related to application task performance in this study.

Hypothesis 6. No significant relationship exists between field dependence-independence and a computer user's performance on computer application tasks in an unfamiliar GUI. This hypothesis was rejected.

A significant regression effect was detected for field dependence-independence (FDI) on posttest performance (POST), when differences in computer experience (CEXP) and time in help (THELP) were controlled by using these concomitant variables as additional covariates. This relationship indicated that students with higher field independence performed more accurately, more rapidly, or both on task-based performance measures when learning application functions in the unfamiliar direct-manipulation GUI.

Individuals who were highly field-independent, who had demonstrated strong visual disembedding skills on the Group Embedded Figures Test (GEFT), were better able to interpret and manipulate the complex visual environment of the graphical spreadsheet application in this study. Because the GUI for the PM Chart application was relatively complex--compared with other direct-manipulation application interfaces--this effect may have been amplified. The relationship between field independence and application task performance might not be detected in studies of applications having simpler user interfaces.

Summary

Effects on application task performance. Significant effects were found for both computer experience and cognitive style on application task performance. Performance improved as computer experience increased and as field independence increased. The relationship between field independence and performance was weaker than that found between prior experience and performance. No significant Aptitude x Treatment interactions were detected related to performance on application tasks, although trends toward such interactions were observed. The small sample size obtained for this study reduced the power of this experimental design to detect significant ATI effects. Future studies examining these questions should be conducted with much larger samples.

Caution regarding generalizations. These results are interpreted here only with respect to the population sampled for this study. Caution must be exercised when attempting to generalize these findings to other populations. In addition, the effects of cognitive style and computer experience on application performance and online help usage must be understood in relation to how these variables were operationally defined and measured in this study. In particular (a) cognitive style referred to field independence as measured with the Group Embedded Figures Test; (b) computer experience was measured using the Computer Experience and Competence Survey; (c) time in help was measured as the total time that application help was displayed during the training lessons; and (d) application task performance was measured in the PM Chart graphical spreadsheet application. Generalization of these results to different populations or other instructional conditions is not recommended.

Recommendations for Future Research

The results of this study may be applied to improving the design of the human-computer interface (HCI), particularly with respect to online help systems. In addition, these findings may influence the design of future intelligent tutoring systems; computer-based instructional systems that can automatically sense and immediately adjust to salient characteristics of learners while they are learning. Finally, this study can be used as an example of applied research where theoretical problems of instructional design may be investigated while significant progress is also made in the development of advanced instructional systems software. These recommendations are presented to prompt other researchers to conduct additional research regarding similar instructional design problems.

Improving HCI Design

The results of this study may lead to improvements in the design of online help and other interface features. Both computer experience and field independence were found to be significantly related to performance on graphical application tasks. There was a trend toward an interaction between field independence and the use of motion video affected task performance. Similarly, there was a trend toward an interaction between computer experience and the presentation of dynamic pictorials in online help that also affected task performance. Each of these results should be considered when designing features of graphical user interfaces.

Sensitivity to user experience. As other research on learning in human-computer interfaces has shown, students' prior experience with computers had a significant effect on their performance in the OS/2 graphical spreadsheet application. Understanding this effect, and developing advanced interface features to accommodate different levels of user expertise, should be a high priority for those engaged in human-computer interaction research and development. Novice users should find the features of a GUI intuitively obvious and easy to learn. Expert users should also find these features intuitive, consistent, and efficient to manipulate. A key goal for HCI designers must be to not place either experts or novices at a disadvantage by incorporating complex or inefficient features into a GUI. Moreover, interface features that might significantly influence the operation of the system or application should first be examined in prototype form and then be evaluated in realistic work settings with groups of potential users who vary considerably in their prior computer experience.

One example of GUI features that created difficulty for novice users was apparent in this study. Several subtle marking techniques--often small or marginally visible graphical symbols--were used in this GUI that indicated changes in status for icons, windows, and other controls. These subtle visual cues were difficult for novice users to recognize. Novices appeared to learn to recognize and identify these markings less readily than experts. Also, icons that appeared nearly identical (e.g., OS/2 icons for folders and folder templates) were often misidentified by novice users. More experienced users required less practice to correctly identify and manipulate such similarly appearing objects. HCI designers should carefully evaluate instances of minimally cued interface changes, and the use of similarly appearing visual symbols, to determine whether novices can correctly identify and manipulate them.

<u>Sensitivity to cognitive style</u>. Design problems similar to those discussed above also relate to designing graphical interfaces that are as usable for field-dependent users as they are for field-independent users. There was no data from this study to suggest that expert users were also highly field-independent. Designers therefore cannot assume that features of a graphical interface that are more usable for novice users will automatically be usable by those with low field independence. Different design issues arise. For example, would field-dependent users find a tree-structured file management interface more usable than a flowed-icon interface? Would field-dependent users find a series of graphical function buttons more efficient to manipulate than pull-down menus? How would the performance of field-independent users be influenced by these different interface structures? These design questions can best be resolved if further research into these phenomena is conducted.

<u>Appropriate use of dynamic pictorials in help</u>. One important instructional design issue prompting this research was the appropriate use of pictorial message

content in online help. The objective of this study was to examine what relationships exist between the presentation of dynamic pictorials in help, the users' computer experience and cognitive style, and their performance on application tasks in a GUI. One inference that may be drawn from these results is that motion video images did not appeal to or did not benefit the most experienced computer users. Using text-only help, expert users were somewhat better able to understand and control the application interface. When motion video images were added to online help, expert users' performance did not increase as much as with text-only help. Also, the addition of motion video images appeared to increase the performance of the most field-independent users while there was no such benefit for field-dependent users.

If these results can be replicated, designers of online help systems might utilize these findings in designing online help and other computer-based tutorial environments. The design of online help for novice users may make greater use of dynamic pictorial content than would be used in help designed for expert users. In addition, alternative visualization techniques might be incorporated to support field-dependent users who would not benefit from the type of dynamic pictorials used in this study. Interface designers, whether focused on computer application or operating system interface features, should systematically evaluate the range of cognitive and affective responses elicited by the online information in their products.

Shneiderman (1986) identified sensitivity to individual differences as one of the most important issues in HCI design. He urged researchers to develop "design guidelines to support individuals with differing gender, age, education, ethnic background, cultural heritage, linguistic background, cognitive styles, [and] learning styles" (p. 346). Advances in HCI design must rely more heavily on the result of rigorous, theoretically motivated studies of user behavior. Studies that concentrate on the effects of individual differences, and the ATI effects between these differences and features of the user interface, will help improve the quality of human-computer interaction.

Intelligent Tutoring Systems

The effects of computer experience and field independence on the use of online help identified in this study can be applied to improving the design of adaptive instructional systems. Online help is one class of computer-based instructional system that typically has very limited ability to adapt to users' individual characteristics. An intelligent tutoring system (ITS) is a computer-based instructional environment that incorporates heuristic decision-making capabilities which allow it to appropriately adapt instructional presentations to best fit certain characteristics of each individual learner.

The results of this and similar studies may be incorporated into the design of an ITS through the development of heuristics relating aptitude variables to parameters of instructional design. This would allow an ITS, for example, to appropriately adapt aspects of an instructional presentation to users with differing levels of computer experience or field independence. This can be done through interactive determination of individual aptitudes, tracking user interface actions, and providing for user control and customization of information presentation parameters.

Interactive assessment of individual differences. In this study, group-administered assessment instruments were used to determine the level of students' computer experience and field independence. Ideally, an ITS could measure these traits using an interactive, online assessment. Methods for performing a variety of interactive assessments are being developed by researchers (Perez & Seidel, 1990). Interactive techniques to measure field independence, such as an online form of the Group Embedded Figures Test, might be developed. Alternative techniques to assess field independence could be incorporated into the interface such that the individual would not become aware that an aptitude test, per se, was being administered. One advantage of this approach would be that aptitudes could be measured without requiring separate testing and data entry procedures. The major benefit, however, would be the capacity to individualize presentation of information by matching presentation attributes to learner characteristics.

Tracking interface activity. In this study, the use of online help was tracked automatically by software that logged all help topic display activity without the student's awareness. The data collected included time in help, the number of help topics opened, the names of the help topics opened, time in help per topic, and the frequency of playing motion video sequences. Further collection and analysis of such data might provide information useful to both the online help designers and the application developers. In an ITS, tracking user actions in this way would provide a continuously updated source of information containing patterns of user response to instructional messages. Decisions regarding instructional presentation may then be made on the basis of that information. In addition, similar tracking logic could be incorporated into any graphical application to construct a profile of a user's manipulation of interface objects. This profile could be examined periodically, and if the pattern of manipulation fell outside certain parameters, the interface might automatically present an explanation of that object, or change the object so it would be easier for that user to understand.

User customization of interface features. Most graphical user interfaces developed for wide use, such as the workplace model incorporated into IBM Operating System/2, provide features that support interface customization by individual users. This customization includes how icons are arranged, how different mouse buttons affect objects in the interface, the colors used to highlight various interface controls, the type and degree of confirmations required for actions on certain objects, and many other features. Online help systems should provide for similar customization capabilities. One user may wish to have information presented with audio-only or audio-visual content, while other users may prefer a text-only display. Once users have determined what information formats and features best suit their needs, they would be able to customize the help environment accordingly.

Related Research Issues

Beyond the scope of this study are many related issues that future research should address. Core issues raised by this study concerned the design of online help messages, appropriate use of dynamic pictorials in online help, and the relationship between cognitive style, computer experience, and performance on tasks in graphical user interfaces. Related issues raised by this study that require further investigation are: (a) the effects of alternative visualization techniques; (b) measuring performance sensitive to field independence; (c) potentially negative effects of using dynamic pictorial elements; and (d) the relationship between computer experience and field independence.

Alternative visualization techniques. The trend toward an interaction between cognitive style and use of dynamic pictorials in online help that influenced application task performance indicated that for some users, in certain applications, such visuals may have a desirable effect. Would a similar effect have occurred if the dynamic pictorial content had been presented using a different technique? For example, would animated bitmaps that precisely matched the application interface have been more effective in improving performance across all levels of field dependence-independence? Would a similar ATI effect be observed with an animated bitmap sequence, or would the effect be modified? Did the visual blur effect in the digital motion video images in this study have a negative influence on students' task performance? Future studies comparing motion video with other visualization techniques may answer these and other related questions.

Performance measurements sensitive to field independence. Another research question is related to the small effect size detected for field independence in this study. Did the manner in which performance was measured, whereby each subtask was scored as either success or failure, overlook subtle ability differences related to field independence? Would a more fine-grained measure of task completion have been more sensitive to the effects of cognitive style? Would a different approach to performance testing yield a more sensitive measure for this type of study? The disembedding skill attributed to highly field-independent users might not be measured in certain user interface tasks. Further research in this area should focus on identifying the categories of interface actions or objects that field-dependent users find most difficult to master. Such studies would lead toward a more complete understanding of the nature of cognitive style influences on human-computer interaction.

Negative effects of dynamic pictorial elements. Although not significant, the trend toward an interaction effect between computer experience and use of motion video on application task performance is indicative of an effect that should be investigated further. Specifically, does the use of dynamic pictorials (i.e., motion video) in online help contribute to a decline in performance as an individual's computer experience increases? If such an effect can be clearly demonstrated, online help systems may be designed to track user activity so that as a user's experience in the application interface increases, the use of dynamic pictorials in online help would be decreased. More conclusive evidence of such an ATI effect is required before such implementations would be justified.

The relationship between field independence and computer experience. In this study, no relationship was found between these aptitude measures. The data appear to indicate that as computer users become more experienced, their level of field independence is not affected. Would field independence remain constant through all types of computer experience? Could prolonged, intensive experience with GUI applications increase individuals' field independence? If such an effect could be demonstrated, future online help systems could be designed to accommodate change in a user's cognitive style, as well as change in the user's level of computer experience.

Future studies addressing these and other related questions would help develop valuable HCI design guidelines, and contribute further to understanding the effects of field dependence-independence and computer experience on learning to operate computers with direct-manipulation graphical user interfaces. This promising research direction provides an opportunity to develop and evaluate instructional message design theory while simultaneously advancing the art of human-computer interface design.

Each of the findings reported here require further investigation. This study demonstrated trends toward ATI effects anticipated by instructional design theory. It also provided new evidence of a significant positive relationship between field independence and task performance in a graphical user interface. For HCI design to benefit from these findings, however, further studies are needed to isolate specific features of graphical user interfaces that contribute to poor performance in field-dependent users. Such interface features might then be eliminated for those users by incorporating user-customization capabilities or by adding adaptive interface features. Greater sensitivity to individual differences will help make computers more human-literate, rather than requiring all users to become computer-literate.

Summary

This study was conducted to examine the effects of field independence and computer experience on learning application functions in a graphical user interface where online help was the primary instructional resource. The experimental instructional treatment consisted of online help incorporating dynamic pictorial message elements. From a university student population in an undergraduate business management course, 38 subjects volunteered for computer-based training. The subjects were randomly assigned to one of two treatment groups that varied only the online help format: text-only help and text-with-motion-video help. In both treatment groups, the display of help information was controlled by the subjects.

For this study, a fully randomized design with pretest-treatment-posttest sequence was used. Data were analyzed using a multiple covariance analysis. Field dependence-independence, computer experience, and time in help were applied as the covariates. The grouping factor was the online help treatment. The dependent measure was performance on application tasks in the training posttest. Significant effects on task performance were found for both field independence and computer experience. No significant interaction effects were found between field independence and treatment or between computer experience and treatment.

From an analysis of these results, several tentative conclusions were drawn. First, the performance of computer users on GUI application tasks increased as either field independence or computer experience increased. Second, there was a trend toward a Field Independence x Treatment interaction. The positive influence of field independence on task performance was greater when online help incorporated dynamic pictorial message elements. There was also evidence of a trend towards a Computer Experience x Treatment interaction. As computer experience increased, the presence of dynamic pictorial elements in online help had an increasingly negative effect on application task performance.

The results of this study should be independently confirmed before these tentative conclusions are applied in HCI design and development. Also, these results may not apply to other populations or for other types of computer applications or user interfaces. Further research into the effects of field independence and computer experience on human-computer interaction are warranted. The results of future studies in this area can lead to the development of intelligent, adaptive user interfaces that are sensitive to individual differences. As working with computers becomes an increasingly pervasive aspect of life, the application of instructional design principles to improving human-computer interaction must become an interdisciplinary priority.

APPENDIX A
OS/2 TRAINING SIGN-UP FORM

The sign-up form on the following page was distributed to students in university classrooms when the students were first contacted about participating in this study. The students were requested to volunteer for training on a new computer system (OS/2) and spreadsheet application. The sign-up form included a brief series of demographic and computer experience questions to obtain cursory descriptive data on the population being solicited.

Volunteer Sign-up Form for OS/2 Training

IBM personnel will be conducting an OS/2 training session as part of a study of learning using computers.

By completing this form, I am indicating my interest in participating in an Introduction to OS/2 Version 2.0 training session to be held this semester at IBM Corporation facilities in Boca Raton. Participation is voluntary.

Please fill in the following information, and sign below the statement at the bottom. You may be contacted later to confirm your interest in volunteering for the study.

______________________________ Student's Name

__________________ Local Phone

Indicate best time to call:

Day(s): ____________________________

Time(s): ____________________________

Academic status at FAU (check one):

Freshman ____ Sophomore ____ Junior ____ Senior ____

Graduate ____ Other ____

Prior computer experience (check all that apply):

	"A little"	"Moderate"	"A lot"
Have used a computer	____	____	____
IBM or compatible	____	____	____
Apple Macintosh	____	____	____
Other	____	____	____
Have used spreadsheet	____	____	____
Have used word processor	____	____	____
Have used Windows (TM)	____	____	____
Have used OS/2	____	____	____

By signing below I indicate my interest in participating in an Introduction to OS/2 Version 2.0 training session. I understand that I am under no obligation to participate, and may withdraw from participation at any time.

______________________________ Student's signature

____________ Date

APPENDIX B
COMPUTER EXPERIENCE AND COMPETENCE SURVEY

Survey items excerpted from the Computer Experience and Competence Survey (CECS) are included in this appendix. The CECS, as described in Chapter 3, was composed of three scales. For this study, only 23 of the 39 items in the Computer Experience scale were used to assess the students' prior computer experience. These 23 items are presented in the following pages, preceded by the instructions for taking the survey. Some items included in the CECS were derived from the Computer Competence test of the 1986 National Assessment of Educational Progress[1]. These items have been edited for use here.

[1] Items from the 1986 NAEP Computer Competence test are used by permission from Educational Testing Service and the Office of Educational Research, United States Department of Education.

COMPUTER EXPERIENCE AND COMPETENCE SURVEY

Taking the Computer Competency Test

The test is comprised of 94 items divided into four sections:

1. Background Survey
2. Computer Experience
3. General Computer Knowledge
4. Computer Applications Knowledge

The first two sections are composed of survey questions. For these two sections, please respond to all questions with the one answer that describes you best.

For the last two sections, each question has a single correct answer. You should try to respond to all questions. All questions are multiple choice. Only one response is allowed for any question.

You will have 30 minutes to complete all questions.

STOP HERE. Wait for the test proctor to tell you to begin.

Computer Experience

For all questions in this section, select only the one best response describing your experience with computers.

9. Have you ever used a computer?
   a. Yes
   b. No

(Note: Items 10, 11, and 12 were removed from the scored assessment.)

13. How long have you used a computer at work or school?
    a. I have never used a computer at work.
    b. Less than 6 months
    c. 6 months to 1 year
    d. 2 years to 5 years
    e. More than 5 years

14. About how many hours per day do you use a computer at work or school?
    a. I never use a computer at work.
    b. Less than 1 hour
    c. 1 to 2 hours
    d. 2 to 4 hours
    e. More than 4 hours

15. How long have you used a computer at home?
    a. I have never used a computer at home.
    b. Less than 6 months
    c. 6 months to 1 year
    d. 2 years to 5 years
    e. More than 5 years

16. About how many hours per day do you use a computer at home?
    a. I never use a computer at home.
    b. Less than 1 hour
    c. 1 to 2 hours
    d. 2 to 4 hours
    e. More than 4 hours

(Note: Items 17 and 18 were removed from the scored assessment.)

19. How often do you use a windowing computer system or product?
    a. Never
    b. Less than once a week
    c. About once a week
    d. Several times a week
    e. Almost every day

20. How often do you use a computer to write reports, letters, stories, or other compositions?
    a. Never
    b. Less than once a week
    c. About once a week
    d. Several times a week
    e. Almost every day

21. How often do you use a computer to solve mathematical or statistical problems?
    a. Never
    b. Less than once a week
    c. About once a week
    d. Several times a week
    e. Almost every day

22. How often do you use a computer to perform scientific measurements, solve science problems?
    a. Never
    b. Less than once a week
    c. About once a week
    d. Several times a week
    e. Almost every day

23. How often do you use a computer to play games?
    a. Never
    b. Less than once a week
    c. About once a week
    d. Several times a week
    e. Almost every day

24. How often do you use a computer to manipulate a database?
    a. Never
    b. Less than once a week
    c. About once a week
    d. Several times a week
    e. Almost every day

25. How often do you use a computer to send and receive electronic messages (e-mail)?
    a. Never
    b. Less than once a week
    c. About once a week
    d. Several times a week
    e. Almost every day

26. How often do you use a computer to draw?
    a. Never
    b. Less than once a week
    c. About once a week
    d. Several times a week
    e. Almost every day

27. How often do you use a computer to make spreadsheets?
    a. Never
    b. Less than once a week
    c. About once a week
    d. Several times a week
    e. Almost every day

28. How often do you use a computer to create charts or graphs?
    a. Never
    b. Less than once a week
    c. About once a week
    d. Several times a week
    e. Almost every day

29. How often do you write computer programs?
    a. Never
    b. Less than once a week
    c. About once a week
    d. Several times a week
    e. Almost every day

(Note: Items 30 and 31 were removed from the scored assessment.)

When using a computer, have you ever:

32. Named or renamed a file?
    a. Yes
    b. No

33. Made a copy of a file?
    a. Yes
    b. No

34. Deleted a file from a disk?
    a. Yes
    b. No

35. Loaded a program into memory?
    a. Yes
    b. No

36. Saved a program on a disk?
    a. Yes
    b. No

37. Run a program?
    a. Yes
    b. No

(Note: Items 38, 39, and 40 were removed from the scored assessment.)

41. Used a printer?
    a. Yes
    b. No

(Note: Items 42 to 47 were removed from the scored assessment.)

This is the end of the computer experience section. YOU MAY PROCEED directly to the next section.

APPENDIX C
COMPUTER TRAINING LESSONS, PRETEST AND POSTTEST

Representative portions of the training materials used in this study are included in this appendix. The training materials consisted of general instructions, pretest tasks, training lessons, and posttest tasks. The general instructions are presented first, followed by the three pretest tasks. Of the 12 training lessons, Lessons 1, 5, and 9 are then presented, followed by the three posttest tasks. Lessons 1, 5, and 9 are representative of the gradually increasing difficulty encountered in the training lessons. Lesson 1 was drawn from the low difficulty group, Lesson 5 was selected from the medium difficulty group, and Lesson 9 was taken from the high difficulty group.

INTRODUCTION TO OS/2

"Working with OS/2" Video

You are about to begin learning how to use OS/2 version 2.0 and an advanced OS/2 spreadsheet and graphing program. Relax and have fun learning to use OS/2!

The first step of the training is to watch a video tape about OS/2 called Working With OS/2 Version 2. This video runs about 40 minutes. In it you will learn about many features of OS/2 and how to use them.

Don't worry about having to remember everything in the video. After you have finished watching the OS/2 video, you will begin working with the OS/2 system and practice the tasks you see in the video tape.

When you are finished watching the video tape, let the instructor know that you are ready to start the OS/2 Tutorial.

OS/2 Online Tutorial

The next step in the training is to complete the OS/2 Tutorial. This is an interactive, online tutorial for OS/2 version 2.0. You'll find that the system has been set up for you with the tutorial already open and ready for you to begin.

If you are already familiar with the information presented by the OS/2 Tutorial, feel free to skip any part. However, if you have little experience using OS/2 version 2.0, you should learn as much as you can from the Tutorial. Remember that OS/2 version 2.0 is different from previous versions of OS/2. The tutorial has been prepared to help you understand these differences and make using OS/2 easier.

When you are finished with the OS/2 Tutorial, please let the instructor know you are ready to begin the pretest.

OS/2 Training Pretest -- Instructions

Before starting the training, we want to measure what you have learned from the OS/2 video tape and Tutorial, along with what you may already know about using OS/2 or computer systems like OS/2.

There are three tasks in this pretest, and each task is timed. You will have up to 10 minutes to complete each task. Online Help is available if you need it. If you cannot figure out how to do the task, even after using online help, you may stop work on the task and ask the instructor for the next one.

If you are still working on a task when the 10 minute time expires, the instructor will ask you to stop, and will prepare your computer for the next task. You may then continue to the next pretest task.

Note: Because OS/2 and PM Chart may be totally new to you, it may be hard to complete any of these pretest tasks. Don't worry though. After the pretest, you'll begin the training tasks. There you will learn, step by step, to master the OS/2 2.0 Workplace Shell and the PM Chart application.

Before starting the pretest, please make sure that you have closed the OS/2 Tutorial. Remember, you may click on the "Exit" push-button in the Tutorial, or double-click on its title bar icon.

Now your OS/2 Desktop should have several icons, but no open windows.

When you are ready to begin the first pretest task, let the instructor know and he will give you the task description.

Pretest Task A -- Opening an Existing Graphic File

The objective of this task is to create a new folder object, open an OS/2 application, and load a graphic file. There are five steps in this task. Be sure to complete each step, one at a time.

1. Locate the "Templates" folder icon and open the Templates folder.
2. Using the "Folder" template, create a new folder on the OS/2 desktop.
3. Open the "APPS" folder and locate PMCHART.EXE. Using the PMCHART.EXE program icon, open the PM Chart application.
4. Using the PM Chart "File" menu, Open the file called INVEST.GRF. The drawing that appears includes a pie chart with labels, a colored background, a title, and a "dollar bill" figure.
5. Using the "Change" menu, open the "Color/Style" dialog box. Then select the "Text" button in the Color/Style dialog. The "Text" button should now be highlighted.

You have completed the first pretest task. The PM Chart application is open, the INVEST.GRF file is displayed, and the Color/Style dialog is open.

Please let the instructor know you are ready to start Task B.

Pretest Task B -- Modifying a Pie Chart

The objective of this task is to add a rectangle around the pie chart, and change the worksheet data for the pie chart. There are five steps to this task. Be sure to complete each step, one at a time.

1. First, close the Color/Style dialog. Then select the Draw tool button on the tool bar and select the Rectangle button in the pop out menu. The mouse pointer will change to look like a pencil with a small rectangle when the pointer is over the drawing area.

2. Using the "draw rectangle" pointer, create a rectangle around the pie chart. This rectangle will serve as a frame for the pie chart. Next, select the rectangle you just drew and then open the Color/Style dialog.

3. Select the Line button in the Color/Style dialog, and using the "Style" pulldown menu, select "Width" and set the rectangle's line width to .104" (.104 inch).

4. Now, using the Worksheet tool on the tool bar, open the PM Chart worksheet window and then move the worksheet window down toward the bottom of the PM Chart window.

5. In the worksheet, change the data value for "Bonds" from 17.0 to 7.0, and change the value for "Savings" from 8.45 to 18.45. Then close the worksheet window to update the pie chart.

You have completed the second pretest task. The graphic now has a frame surrounding the pie chart, and the pie chart looks different because the data in the worksheet was changed.

Please let the instructor know you are ready to start Task C.

<u>Pretest Task C</u> -- Changing Title Text and Color

The objective of this task is to modify the title, change text font settings, and save the graphic to a new file. There are five steps to this task. Be sure to complete each step, one at a time.

1. Using the mouse pointer, select the chart title text: "Investment Returns". Then using the Edit menu, clear the title from the drawing page.

2. To create a new title, select the Text tool, then select the Create Text button. The mouse pointer will change shape to an arrow with the letter "T". Now, select a point near the top of the drawing page and type: "Portfolio Earnings".

3. Now, select the title text object you created. Again using the Text tool button, open the "Fonts - Options" dialog. Then set the font size to 36 points, "Times New Roman Outline", with "bold" style.

4. Now move the title text to center it over the pie chart. Then resize (stretch) the title text so that it appears about as long as the pie chart is wide.

5. With the title text still selected, open the Color/Style dialog and use the "Palette" menu to select a new color range. Then set the text color for the title. Finally, save the graphic to a new file named INVEST2.GRF.

Good! You have completed the third and final pretest task. The modified pie chart graphic has a new, colorful title and has been saved to a new file.

Please let the instructor know you have completed the pretest. You are now ready to begin the <u>PM Chart</u> training lessons.

OS/2 Training -- Instructions

The next training activity is comprised of twelve lessons that will help you learn how to use OS/2 Version 2, and a spreadsheet and graphics program that comes with OS/2: PM Chart.

PM Chart lets you prepare informative graphical data presentations based on data from spreadsheet programs such as Lotus 1-2-3[2] or Microsoft Excel[3]. Spreadsheet data is first loaded into PM Chart, then by using the PM Chart "Toolbar" functions, you can create and edit graphs and charts that display the data in various formats and colors.

In this training, for example, you will turn an "Expenses" spreadsheet into a presentation graphic containing a bar chart. You will then add a title, colors, drawings, and other graphical contents to help depict the data.

To successfully complete these lessons, you may need help! OS/2 provides online help in the Desktop and in applications. In addition, keep in mind the following suggestions as you work on the lessons:

1. Use the online Help functions whenever needed to learn how to manipulate the OS/2 Desktop or the PM Chart program. Although you have no printed manuals describing how to use the system, all the information you need is ready at your fingertips using the online help functions. Remember these four ways to access Help:
    a. Press the "F1" key
    b. Pull down the "Help" menu from the action bar
    c. Click on the "Help" push-button when one appears
    d. Look up the topic in the Master Help Index

2. Although time is limited, feel free to explore the information available to you in Help. Likewise, feel free to explore using the application menus if you think it may help you to more fully understand the application and how to finish the lesson.

3. This training is organized in the form of a project, arranged as a series of lessons. Each lesson, when completed, prepares you to start the next lesson. For each lesson, five steps are given. Follow these steps in order to complete the lesson.

4. For each step in any lesson, directions are given that describe what to do. This is followed by a "HINT" section that gives you more information on how the step may be completed. These "HINTS" may be ignored if you feel comfortable with how to complete that step. When in doubt, read the "HINT" closely.

---

[2] Lotus and 1-2-3 are registered trademarks of Lotus Development Corporation.

[3] Microsoft is a registered trademark and Excel is a trademark of Microsoft Corporation.

5. If you are not sure what is meant by the printed instructions you receive for a lesson, and you cannot proceed, you may ask the instructor for assistance.

6. You can take a break between lessons. Just let the instructor know before starting the next lesson that you'd like to take a break.

When you are ready, you may begin with Lesson #1. Please ask the instructor for the lesson description.

Lesson #1 -- Create a New Folder

This lesson is the first step in your project to create a graphical data presentation chart using OS/2 and PM Chart. First you'll practice making changes to an existing graphic, then you'll create your own graphic presentation file.

Before starting this lesson, please make sure that you have closed the OS/2 Tutorial. Remember, you may click on the "Exit" push-button in the Tutorial, or double-click on its title bar icon. Now your OS/2 Desktop should have several icons, but no open windows.

The objective of this lesson is to create a new Folder object that you will call "My Project". There are five steps to this lesson. Be sure to complete each step, one at a time. Use the HINTS if you need help.

Remember -- when in doubt: Use OS/2 HELP!

1. Locate the "Templates" folder icon on the OS/2 desktop and double-click on the icon to open the Templates folder.

   HINT: To get help for Templates, move the pointer over the Templates icon and press the right mouse button. This opens the object's menu. In the menu, select Help, and "Help for Templates" will be displayed.

2. Create a new folder by dragging a Folder template from the Templates window onto a blank area of the OS/2 desktop.

   HINT: If you need help for creating a new folder, open the Folder template's popup menu and select "Help". In the help window, double-click on "Creating an object (using a template)" to view that information. Double-click on any other highlighted (green) phrase for further information.

3. Change the name of the new folder from "Folder" to "My Project".

   HINT: If you need help for renaming objects, open the Master Help Index, select the "Search topics..." button, enter "changing names" in the search string, and select the "Search" button. Look for "folder object -- changing names". Double click on this item to display the "Changing names of Objects" information.

4. Now, close the Templates folder window.

   HINT: You can close a window by double-clicking its title bar icon.

5. Finally, open the "My Project" folder window, and then move the window to the lower left corner of the OS/2 desktop.

You have created a "My Project" folder, and now have its window open on your OS/2 desktop. Currently, the folder is empty.

You have completed Lesson #1. Please let the instructor know when you are ready to proceed with Lesson #2.

Lesson #5 -- Add New Graphic Objects to Figure

In this lesson, you will create a rectangular frame around the pie chart, set the rectangle's line width and color, and save the graphic file. Each step may require more than one operation. Be sure to complete each step before proceeding to the next step.

Remember: When in doubt use PM Chart Help!

1. Select the Draw tool button on the tool bar (near the left edge of the PM Chart window), and then select the Rectangle button in the pop out menu. The mouse pointer will change to look like a pencil with a small rectangle when the pointer is over the drawing area.

   HINT: Moving the pointer over the tool bar displays information about each tool in the status area at the bottom of the PM Chart window. To get help using the Draw tool, open the PM Chart Help Index and find "drawing". Double-click on that topic to open the "Help for Draw" information.

2. Using the "draw rectangle" pointer, create a rectangle around the pie chart. This rectangle will serve as a frame for the pie chart.

   HINT: To get help for drawing rectangles, with the "draw rectangle" pointer visible, press the "F1" key to display "Help for Rectangle". Select the phrase "Creating closed symbols" for more information.

3. Now, return to the default pointer by clicking on the Select Arrow tool on the tool bar. Next, select the rectangle you just drew (click on any corner). Then, open the Color/Style dialog (using the "Change" pulldown menu).

   HINT: Help for Select Arrow can be displayed by opening PM Chart Help, selecting the "Search..." button, and typing "select arrow". Select the "All sections" button, then press "Search". From the search results window, select "Help for Select Arrow".

4. Select the "Line" button in the Color/style dialog, and using the "Style" pulldown menu, select "Width" and change the rectangle's line width to .062" (.062 inch).

   HINT: Help for setting line width can be displayed by pressing "F1" when the "Width" option is selected from the "Style" pulldown menu.

5. With the Line button still selected in the "Color/Style" dialog, change the color of the rectangle outline. Then, save the graphic file once again.

In this lesson, you created a rectangle around the pie chart, changed the line width for that rectangle, and set its color, making a frame for the pie chart. You also saved the graphic file.

You have completed Lesson #5. Please let the instructor know when you are ready to proceed with Lesson #6.

Lesson #9 -- Change Chart Position, Size and Colors

The next step in your project is to move and resize the bar chart for the new page format, and then change the chart colors and label colors. There are five steps in this lesson.

Remember: When in doubt use PM Chart Help!

1. Move and resize the bar chart to center and fit in the new drawing page orientation. Using the page rulers, leave approximately 1-inch margins around the edges as you size and position the chart. Then, open the Colors/Style dialog.

   HINT: Help for moving and sizing objects can be displayed by selecting "Moving a symbol" in the Help index.

2. To change the colors of the bar chart, be sure the chart is selected. Using the Color/Style dialog, select "Chart colors" from the pulldown menu. Then select a color (double-click on any color square).

   HINT: To get help for "Color/Style", press the "Help" push-button in the dialog.

3. Next, to change the text colors, select one of the text labels in the chart (such as "Phone"). Then open the "Colors/Style" dialog. Select the "All text" button (at the bottom of the dialog), then select a color.

4. Now, color the chart "axis frame". In the three-dimensional chart, each axis is a rectangle viewed in perspective. Select the axis frame (small white "handles" will appear at each corner). Open the Color/Style dialog. Select the same color you used for the chart labels.

5. Finally, deselect the chart object (click in any blank area of the drawing page away from the chart), then save the drawing in a file called "EXPENSE3.GRF".

In this lesson, you resized and moved the chart to fit the page orientation, and changed the chart and label colors. Finally you saved the graphic to a new file.

You have completed Lesson #9. Please let the instructor know when you are ready to proceed with Lesson #10.

OS/2 Training Posttest -- Instructions

Now that you have completed the training, we want to measure what you have learned. We can then compare the result of this posttest to the score you received on the pretest.

There are three tasks in the posttest, and each task is timed. You will have up to 10 minutes to complete each task. Online Help is available if you need it. If you cannot figure out how to do the task, even after using online help, you may stop work on the task and ask the instructor for the next one.

If you are still working on a task when the 10 minute time expires, the instructor will ask you to stop, and will prepare your computer for the next task. You may then continue to the next task. If you become stuck on a task and cannot proceed,

Before starting the posttest, please make sure that you have closed the PM Chart application. Also close any other open folders.

Now your OS/2 Desktop should have several icons, but no open windows.

When you are ready to begin the first posttest task, let the instructor know and he will give you the task description.

Posttest Task A -- Opening an Existing Graphic File

The objective of this task is to create a new folder object, open an OS/2 application, and load a graphic file. There are five steps in this task. Be sure to complete each step, one at a time.

1. Locate the "Templates" folder icon and open the Templates folder.

2. Using the "Folder" template, create a new folder on the OS/2 desktop.

3. Open the "APPS" folder and locate PMCHART.EXE. Using the PMCHART.EXE program icon, open the PM Chart application.

4. Using the PM Chart "File" menu, open the file called GREEN.GRF. The drawing that appears includes a column chart with labels, a colored background, a title, and a green "leafy" figure.

5. Using the "Change" menu, open the "Color/Style" dialog box. Then select the "Set" button in the Color/Style dialog. The "Set" button should no longer be highlighted.

You have completed the first posttest task. The PM Chart application is open, the GREEN.GRF file is displayed, and the Color/Style dialog is open.

Please let the instructor know you are ready to start Task B.

Posttest Task B -- Modifying a Column Chart

The objective of this task is to add a rectangle around the column chart, and change the worksheet data for the column chart. There are five steps to this task. Be sure to complete each step, one at a time.

1. First, close the "Color/Style" dialog. Then select the Draw tool button on the tool bar and select the Rounded Rectangle button in the pop out menu. The mouse pointer will change to look like a pencil with a small rounded rectangle when the pointer is over the drawing area.

2. Using the "draw rounded rectangle" pointer, create a rounded rectangle around the column chart. This rounded rectangle will serve as a frame for the column chart. Next, select the rounded rectangle you just drew and then open the Color/Style dialog.

3. Select the Line button in the Color/style dialog, and using the "Style" pulldown menu, select "Width" and set the rectangle's line width to .083" (.083 inch).

4. Now, using the Worksheet tool on the tool bar, open the PM Chart worksheet window and then move the worksheet window down toward the bottom of the PM Chart window.

5. In the worksheet, change the data value for year '88 from 15 to 25, and change the value for year '90 from 42 to 52. Then close the worksheet window to update the column chart.

You have completed the second posttest task. The graphic now has a rounded rectangle frame, and the column chart looks different because the data in the worksheet was changed.

Please let the instructor know you are ready to start Task C.

Posttest Task C -- Changing Title Text and Color

The objective of this task is to modify the title, change text font settings, and save the graphic to a new file. There are five steps to this task. Be sure to complete each step, one at a time.

1. Using the mouse pointer, select the chart title text: "Environmental Awareness". Then using the Edit menu, clear the title from the drawing page.

2. To create a new title, select the Text tool, then select the Create Text button. The mouse pointer will change shape to an arrow with the letter "T". Now, select a point near the top of the drawing page and type: "Earth Consciousness".

3. Now, select the title text object you created. Again using the Text tool button, open the "Fonts - Options" dialog. Then set the font size to 38 points, "Tms Rmn Outline", with "bold" style.

4. Now move the title text to center it over the column chart. Then resize (stretch) the title text so that it appears about as long as the column chart is wide.

5. With the title text still selected, open the Color/Style dialog and use the "Palette" menu to select a new color range. Then set the text color for the title. Finally, save the graphic to a new file named GREEN2.GRF.

Congratulations! You have completed the third and final posttest task. The modified column chart graphic has a new, colorful title and has been saved to a new file.

Please let the instructor know you have finished the posttest.

APPENDIX D
HELP TRACKING LOG FILE EXAMPLE

This appendix contains an example of an online help tracking log file for one subject. These log files were created automatically by a computer program that monitored the subject's use of help messages. The log file contains subject identification data, help usage summary data, followed by a list of individual records of online help message use. Note that each record in this list was recorded with the time of day. Time in help data was obtained from these log files.

Test Name:	MMPM Help for PM Chart
Test Date:	Monday, Dec 28, 1992
Test Time:	06:23:09 pm
Test Location:	Usability Test Lab Cell 7
Test Hardware:	MOD 56SLC, Action Media II
Test Subject Name:	D---- C-------
Test Moderator Name:	John Tyler
Test Data File Name:	D:\MMPMDATA
Test Log File Name:	D:\MMPM0618.LOG
Test INI File Name:	D:\MMPMHELP.INI

Total Help Time is:	00:04:55
Extended Time in Help is:	00:06:36
Number of times help was referenced is:	10
Number of help topics viewed is:	8
Number of videos viewed is:	5

Topics viewed :

Help for Draw
Help for Colors/Style
Help for View
Help for Pages
Resizing and Moving Symbols
Help for Rotate
Help for Move To
Help for Print

Videos viewed :

Help for Rectangle Video
Help for Colors/Style Video
Help for View Page Video
Help for Pages Video
Help for Rotate Video

LINE	HELPTYPE	TITLE	START	END	TOTAL
0001	HELP	PM Chart Help	00:00:00		
0002	TOPIC	Help for PM Chart	00:00:00	00:00:15	00:00:15
0003	TOPIC	Help for Draw	00:00:15	00:00:23	00:00:07
0004	VIDEO	Help for Rectangle Video	00:00:41		
0005	TOPIC	Help for Draw	00:00:23	00:01:25	00:01:01
0006	HELP	PM Chart Help	00:00:00	00:01:25	
0007	HELP	PM Chart Help	00:04:44		
0008	TOPIC	Help for PM Chart	00:04:44	00:04:53	00:00:08
0009	VIDEO	Help for Colors/Style Video	00:04:55		
0010	TOPIC	Help for Colors/Style	00:04:53	00:05:29	00:00:36
0011	HELP	PM Chart Help	00:04:44	00:05:29	
0012	HELP	PM Chart Help	00:17:59		
0013	TOPIC	Help for PM Chart	00:17:59	00:18:09	00:00:10
0014	TOPIC	Help for View	00:18:09	00:18:14	00:00:05
0015	VIDEO	Help for View Page Video	00:18:21		
0016	TOPIC	Help for View	00:18:14	00:18:48	00:00:33
0017	HELP	PM Chart Help	00:17:59	00:18:48	
0018	HELP	PM Chart Help	00:21:32		
0019	TOPIC	Help for PM Chart	00:21:33	00:21:45	00:00:11
0020	VIDEO	Help for Pages Video	00:21:51		
0021	TOPIC	Help for Pages	00:21:45	00:22:06	00:00:21
0022	HELP	PM Chart Help	00:21:32	00:22:06	
0023	HELP	PM Chart Help	00:26:17		
0024	TOPIC	Help for PM Chart	00:26:18	00:26:35	00:00:16
0025	TOPIC	Resizing and Moving Symbols	00:26:35	00:27:02	00:00:27
0026	HELP	PM Chart Help	00:26:17	00:27:02	
0027	HELP	PM Chart Help	00:29:31		
0028	TOPIC	Help for PM Chart	00:29:32	00:29:43	00:00:11
0029	TOPIC	Help for Colors/Style	00:29:43	00:29:51	00:00:08
0030	HELP	PM Chart Help	00:29:31	00:29:51	
0031	HELP	PM Chart Help	00:45:04		
0032	VIDEO	Help for Rotate Video	00:45:21		
0033	TOPIC	Help for Rotate	00:45:04	00:45:45	00:00:40
0034	HELP	PM Chart Help	00:45:04	00:45:45	
0035	HELP	PM Chart Help	00:46:31		
0036	TOPIC	Help for Rotate	00:46:31	00:46:52	00:00:20
0037	HELP	PM Chart Help	00:46:31	00:46:52	
0038	HELP	PM Chart Help	00:50:54		
0039	TOPIC	Help for PM Chart	00:50:55	00:51:07	00:00:12
0040	TOPIC	Help for Move To	00:51:07	00:51:25	00:00:17
0041	HELP	PM Chart Help	00:50:54	00:51:25	
0042	HELP	PM Chart Help	00:56:49		
0043	TOPIC	Help for PM Chart	00:56:50	00:57:04	00:00:13
0044	TOPIC	Help for Print	00:57:04	00:57:20	00:00:16
0045	HELP	PM Chart Help	00:56:49	00:57:20	

APPENDIX E
OBSERVER LOG FILE EXAMPLE

The following pages in this appendix present an observer's log file for one subject. This log file was created by the observer using a computer program specifically designed for this purpose. Note that each entry in the log consists of the entry number, the entry code, a description of the observation, and the time of day. Log entry codes represented the type of log entry being made.

Entry	CODE	DESCRIPTION	TIME STAMP
0	Note	Date: 12/28/92 Subject: D---- C------	16:10:30
1	Note	Video Help Treatment	16:10:34
2	Note	Test Cell 7	16:10:39
3	Note	John Tyler Monitoring	16:10:49
4	Note	GEFT Only test required	16:10:54
5	Note	Scheduled for 4:00pm	16:11:01
6	**********	**************************************	16:11:02
7	Note	Starting GEFT	16:12:48
8	Note	Start Section 1	16:16:25
9	Note	End Sec1	16:18:08
10	Note	Start Sec 2	16:18:12
11	Note	end Sec 2	16:23:05
12	Note	Start Section 3	16:23:14
13	Note	End Sec 3	16:28:14
14	**********	**************************************	16:29:04
15	Note	Starting OS/2 Video Tape	16:29:14
16	Note	End of OS/2 Video	17:11:53
17	**********	**************************************	17:25:16
18	Start Test		17:25:18
19	Task Start	The OS/2 Tutorial	17:25:27
20	Note	USING THE MOUSE lesson	17:26:26
21	Note	USING OBJECTS lesson	17:30:59
22	Note	USING WINDOWS lesson	17:34:34
23	Note	GETTING HELP lesson	17:38:28
24	Note	OS/2 SYSTEM OVERVIEW Lesson	17:40:51
25	FinishTask	End of OS/2 Tutorial	17:44:28
26	**ANAL**	Task 0:19:01 P = 0 H = 0 U = 0	17:44:28
27	**********	**************************************	17:44:30
28	Task Start	PRETEST A	17:46:12
29	Step Compl	1	17:46:26
30	Note	opened template object	17:47:02
31	Note	copied template object to desktop	17:47:12
32	Note	closed templates folder	17:47:25
33	Step Compl	2	17:47:39
34	Note	created folder from template on desktop	17:47:50
35	Note	deleted template from desktop	17:48:10
36	Step Compl	3	17:48:38
37	Note	mouse not tracking well	17:48:43
38	Step Compl	4	17:49:18
39	Step Compl	5	17:49:36
40	FinishTask		17:49:37
41	**ANAL**	Task 0:03:25 P = 0 H = 0 U = 0	17:49:37
42	Task Start	PRETEST B	17:50:01
43	Note	selecting objects in drawing page	17:50:47
44	Note	dragging background color object	17:51:06
45	Action	stick to the instructions	17:51:47
46	Comment	trying to select rect	17:51:55
47	Action	what does this step say?	17:52:00
48	Comment	select rect	17:52:04
49	Action	before that?	17:52:07
50	Comment	select draw tool	17:52:12

Entry	CODE	DESCRIPTION	TIME STAMP
51	Action	OK do that	17:52:16
52	Step Compl	1	17:52:17
53	Step Compl	2	17:52:20
54	Step Compl	3	17:52:58
55	Action	what does this step saY?	17:55:11
56	USABILITY	step4, required ASSIST to open wrksheet	17:56:15
57	Step Compl	5	17:57:16
58	FinishTask		17:57:16
59	**ANAL**	Task 0:07:15 P = 0 H = 0 U = 1	17:57:16
60	Task Start	PRETEST C	17:57:34
61	Step Compl	1	17:57:53
62	Note	edit window opens, entering title text	17:58:36
63	Note	hit enter to complete	17:59:00
64	Step Compl	2 (doubleclick on drawing page to exit)	17:59:20
65	Step Compl	3	18:00:00
66	Step Compl	4 (ok, but not exactly to instruct)	18:01:15
67	Step Compl	5	18:01:55
68			18:01:55
69	FinishTask		18:01:56
70	**ANAL**	Task 0:04:22 P = 0 H = 0 U = 0	18:01:56
71	**********	************************************	18:01:58
72	Note	took a bio break	18:14:06
73	Task Start	LESSON #1	18:14:12
74	Step Compl	1	18:15:21
75	Step Compl	2	18:15:24
76	Step Compl	3	18:15:58
77	Step Compl	4	18:16:00
78	Step Compl	5	18:16:02
79	FinishTask		18:16:02
80	**ANAL**	Task 0:01:50 P = 0 H = 0 U = 0	18:16:02
81	Task Start	Lesson #2	18:17:12
82	Step Compl	1	18:17:13
83	Step Compl	2	18:17:15
84	Step Compl	3	18:17:21
85	Note	open popup menu, Drive A:	18:17:53
86	HELP	Master Index	18:18:08
87		search : Copying	18:18:53
88	HELP	Copying from a Diskette	18:19:03
89	Step Compl	4	18:19:46
90	Step Compl	5	18:19:50
91	FinishTask		18:19:51
92	**ANAL**	Task 0:02:39 P = 0 H = 2 U = 0	18:19:51
93	Task Start	LESSON #3	18:20:09
94	Step Compl	1	18:23:07
95	Step Compl	2	18:23:08
96	Step Compl	3	18:23:09
97	Step Compl	4	18:23:09
98	Step Compl	5	18:23:12
99	FinishTask		18:23:13
100	**ANAL**	Task 0:03:04 P = 0 H = 0 U = 0	18:23:13

Entry	CODE	DESCRIPTION	TIME STAMP
101	Task Start	USING PMCHART HELP	18:23:25
102	Step Compl	1	18:24:50
103	Step Compl	2	18:24:51
104	Step Compl	3	18:24:53
105	Step Compl	4	18:24:54
106	Step Compl	5	18:25:24
107	FinishTask		18:25:28
108	**ANAL**	Task 0:02:03 P = 0 H = 0 U = 0	18:25:28
109	Task Start	LESSON #4	18:26:21
110	Step Compl	1	18:26:57
111	Step Compl	2	18:28:02
112	Step Compl	3	18:28:05
113	Step Compl	4	18:28:11
114	VIDEO Time	Tape is synchronized >>>>>>>>>>>>>>	18:28:47
115	HELP	Help for Color/Style	18:28:56
116	HELP	Help for Color Style Video	18:29:03
117	Note	close help	18:29:29
118	Step Compl	5	18:29:36
119	FinishTask		18:29:37
120	**ANAL**	Task 0:03:16 P = 0 H = 2 U = 0	18:29:37
121	Comment	I really like that video, it's neat	18:30:12
122	Task Start	LESSON #5	18:30:17
123	Step Compl	1	18:30:54
124	Note	created small rect	18:31:06
125	Note	used EDIT/Undo to delete	18:31:13
126	Step Compl	2	18:31:18
127	Step Compl	3	18:31:50
128	Step Compl	4	18:32:16
129	Step Compl	5	18:33:08
130	FinishTask		18:33:08
131	**ANAL**	Task 0:02:51 P = 0 H = 0 U = 0	18:33:08
132	Task Start	LESSON #6	18:33:20
133	Step Compl	1	18:34:24
134	Step Compl	2	18:34:38
135	Step Compl	3	18:35:53
136	Step Compl	4	18:36:36
137	Step Compl	5	18:37:15
138	FinishTask		18:37:17
139	**ANAL**	Task 0:03:57 P = 0 H = 0 U = 0	18:37:17
140	Task Start	LESSON #7	18:37:32
141	Step Compl	1	18:38:58
142	Step Compl	2	18:39:04
143	Step Compl	3	18:41:07
144			18:41:14
145	HELP	Help Index	18:41:57
146	HELP	Help for View	18:42:17
147	HELP	Help for View Page	18:42:23
148	HELP	Help for View Page Video	18:42:29
149	Step Compl	4	18:42:54
150	Step Compl	5	18:43:38

Entry	CODE	DESCRIPTION	TIME STAMP
151	FinishTask		18:43:38
152	**ANAL**	Task 0:06:06 P = 0 H = 4 U = 0	18:43:38
153	Action	did the video window help there	18:44:35
154	Comment	I liked that, it showed the buttons well	18:44:49
155	Task Start	LESSON #8	18:45:02
156	HELP	Help Index	18:45:36
157	HELP	Help for Pages	18:45:48
158	HELP	Help for Pages VIdeo	18:45:54
159	Note	close help	18:46:07
160	Step Compl	1	18:46:21
161	Note	set page view	18:46:47
162	Step Compl	2	18:47:10
163	Step Compl	3	18:47:44
164	Step Compl	4	18:48:34
165	Step Compl	5	18:49:00
166	FinishTask		18:49:01
167	**ANAL**	Task 0:03:59 P = 0 H = 3 U = 0	18:49:01
168	Task Start	LESSON #9	18:49:18
169	HELP	Reszing and Moving Symbols	18:50:47
170	Step Compl	1	18:52:23
171	Note	closed & reopened Color/Style	18:53:06
172	Note	closed Color/Style, select EDIT menu	18:53:25
173	HELP	Help Indxex	18:53:35
174	HELP	Help for Color Style	18:53:51
175	Action	what step are you on?	18:55:11
176	Comment	step2, set chart colors	18:55:19
177	Action	did you try the menus in Color/Style	18:55:31
178	Comment	OH, OK	18:55:37
179	Step Compl	2	18:56:16
180	Step Compl	3	18:56:46
181	Step Compl	4	18:57:32
182	Step Compl	5	18:58:00
183	FinishTask		18:58:01
184	**ANAL**	Task 0:08:43 P = 0 H = 3 U = 0	18:58:01
185	Task Start	LESSON #10	18:59:19
186	Step Compl	1	18:59:24
187	Step Compl	2	19:00:19
188	Note	used text font options to set font size	19:00:58
189	Note	text title split to 2 lines	19:01:07
190	Note	resizing & moving title	19:01:13
191	Step Compl	3	19:01:18
192	Note	reset font size using options to 18	19:01:45
193	Note	resizing & moving title again	19:01:55
194	Note	reset font size using font-options	19:02:42
195	Note	resizing & moving title a third time	19:02:53
196	Step Compl	4	19:03:29
197	Note	set text color 3 times	19:04:21
198	Note	using different palettes, colors	19:04:38
199			19:04:49
200	Note	setting different bkg colors	19:05:08

Entry	CODE	DESCRIPTION	TIME STAMP
201	Step Compl	5	19:05:25
202	FinishTask		19:05:26
203	**ANAL**	Task 0:06:07 P = 0 H = 0 U = 0	19:05:26
204	Task Start	LESSON #11	19:06:03
205	Step Compl	1	19:06:48
206	Step Compl	2	19:07:24
207	Step Compl	3	19:08:54
208	HELP	Help for Rotate (F1)	19:09:09
209	HELP	Help for Rotate Video	19:09:26
210	Note	TS nodding head	19:09:40
211	Note	close help	19:09:45
212	Note	open worksheet, button 2 click	19:10:08
213	Note	reselect rotate	19:10:30
214	Note	open worksheet again	19:10:40
215	HELP	Help for Rotate	19:10:48
216	Step Compl	4	19:11:33
217	Step Compl	5	19:12:20
218	FinishTask		19:12:21
219	**ANAL**	Task 0:06:18 P = 0 H = 3 U = 0	19:12:21
220	Task Start	LESSON #12	19:12:34
221	Step Compl	1	19:13:30
222	Note	adjusting size of rect	19:13:54
223	Note	scrolling page to check rect is selected	19:14:25
224	HELP	Help Index	19:14:57
225	HELP	Help for Move To	19:15:11
226	Note	closing help	19:15:24
227	Step Compl	2	19:15:29
228	Note	gradient is solid green	19:18:25
229	USABILITY	step3, did not select style for gradient	19:18:59
230	Step Compl	4	19:20:09
231	Action	did you do step 5?	19:20:19
232	Comment	I've not gotten there yet	19:20:26
233	Step Compl	5 (did not select white from Primary	19:22:02
234		palette colors, chose green instead)	19:22:11
235	FinishTask		19:22:16
236	**ANAL**	Task 0:09:42 P = 0 H = 2 U = 1	19:22:16
237	Note	chart printing	19:22:22
238	End Test		19:22:25
239	**TEST**	Test 1:34:38 P = 0 H = 19 U = 2	19:22:25
240	Start Test		19:22:29
241	**********	**************************************	19:22:31
242	Task Start	POSTTEST A	19:26:22
243	Step Compl	1	19:26:30
244	Step Compl	2	19:26:37
245	Step Compl	3	19:27:06
246	Step Compl	4	19:27:33
247	Step Compl	5	19:27:59
248	FinishTask		19:27:59
249	**ANAL**	Task 0:01:37 P = 0 H = 0 U = 0	19:27:59

Entry	CODE	DESCRIPTION	TIME STAMP
250	Task Start	POSTTEST B	19:28:33
251	Step Compl	1	19:28:45
252	Step Compl	2	19:29:26
253	Step Compl	3	19:29:45
254	Step Compl	4 (w/ button2	19:30:06
255	Step Compl	5	19:30:35
256	FinishTask		19:30:36
257	**ANAL**	Task 0:02:03 P = 0 H = 0 U = 0	19:30:36
258	Task Start	POSTTEST C	19:30:54
259	Step Compl	1	19:31:10
260	Note	Typing title	19:31:41
261	Step Compl	2	19:31:49
262	Step Compl	3	19:32:39
263	Step Compl	4	19:33:29
264	Step Compl	5	19:34:20
265	FinishTask		19:34:20
266	**ANAL**	Task 0:03:26 P = 0 H = 0 U = 0	19:34:20
267	End Test		19:34:24
268	**TEST**	Test 0:07:06 P = 0 H = 19 U = 2	19:34:24
269	END LOG	************************************	19:34:24

REFERENCES

Agresti, A., & Agresti, B. F. (1979). Statistical methods for the social sciences. San Francisco: Dellen.

Aster, D. J., & Clark, R. E. (1985). Instructional software for users who differ in prior knowledge. Performance and Instruction Journal, 24(5), 13-15.

Bernard, R. M. (1990). Effects of processing instructions on the usefulness of a graphic organizer and structural cueing in text. Instructional Science, 19, 207-217.

Burwell, L. B. (1991). The interaction of learning styles with learner control treatments in an interactive videodisc lesson. Educational Technology, 31(3), 37-43.

Canelos, J., Taylor, W., Dwyer, F., & Belland, J. (1988, January). Programmed instructional materials: A preliminary analysis. Proceedings of selected research papers presented at the annual meeting of the Association for Educational Communications and Technology. New Orleans. (ERIC Document Reproduction Service No. ED 295 630).

Canino, C., & Cicchelli, T. (1988). Cognitive styles, computerized treatments on mathematics achievement and reaction to treatments. Journal of Educational Computing Research, 4, 253-264.

Cathcart, W. G. (1990). Effects of LOGO instruction on cognitive style. Journal of Educational Computing Research, 6, 231-242.

Cavaiani, T. P. (1989). Cognitive style and diagnostic skills of student programmers. Journal of Research on Computing in Education, 21, 411-420.

Chiesi, H. L., Spilich, G. L., & Voss, J. F. (1979). Acquisition of domain-related information in relation to high and low domain knowledge. Journal of Verbal Learning and Verbal Behavior, 18, 257-273.

Clark, R. E., & Salomon, G. (1986). Media in teaching. In M. C. Wittrock (Ed.), Handbook of research on teaching (pp. 464-478). New York: Macmillan.

Cliff, N. (1987). Analyzing multivariate data. San Diego, CA: Harcourt Brace Jovanovich.

Cronbach, L. J., & Snow, R. E. (1977). Aptitudes and instructional methods: A handbook for research on interactions. New York: Irvington.

DeHaemer, M. J., & Wallace, W. A. (1992). The effects on decision task performance of computer synthetic voice output. International Journal of Man-Machine Studies, 36, 65-80.

Dwyer, F. M., & Moore, D. M. (1992). Effect of color coding on visually and verbally oriented tests with students of different field dependence levels. Journal of Educational Technology Systems, 20, 311-320.

Educational Testing Service (1988). A framework for assessing computer competence: Defining objectives. Princeton, NJ: National Assessment of Educational Progress.

Federico, P. A. (1983). Changes in the cognitive components of achievement as students proceed through computer-managed instruction. Journal of Computer-Based Instruction, 9(4), 156-168.

Fleming, M. L. (1987). Displays and communication. In R. M. Gagne (Ed.), Instructional technology: Foundations (pp. 233-260). Hillsdale, NJ: Lawrence Erlbaum.

Fleming, M. L., & Levie, W. H. (1978). Instructional message design: Principles from the behavioral sciences. Englewood Cliffs, NJ: Educational Technology.

Germann, P. J. (1989). Directed-inquiry approach to learning science process skills: treatment effects and aptitude-treatment interactions. Journal of Research in Science Teaching, 26, 237-250.

Gregorc, A. F. (1984). Style as symptom: A phenomenological perspective. Theory Into Practice, 23(1), 51-55.

Horton, W. K. (1990). Designing and writing online documentation: Helpfiles to hypertext. New York: J. Wiley.

Huitema, B. (1980). The analysis of covariance and alternatives. New York: J. Wiley.

LaLomia, M. J., & Sidowski, J. B. (1990). Measurements of computer satisfaction, literacy, and aptitudes: A review. International Journal of Human-Computer Interaction, 2, 231-253.

Lesgold, A. M., Gabrys, G., & Magone, M. (1990). Cognitive and instructional theories of impasses in learning (Final report). Pittsburg University, PA. Learning Research and Development Center. (ERIC Document Reproduction Service No. ED 317 578)

MacGregor, S. K., Shapiro, J. Z., & Niemiec, R. (1988). Effects of a computer-augmented learning environment on math achievement for students with differing cognitive style. Journal of Educational Computing Research, 4, 453-456.

McNeil, B. J., & Nelson, K. R. (1991). Meta-analysis of interactive video instruction: A 10 year review of achievement effects. Journal of Computer-Based Instruction, 4, 1-6.

Martin, M. A. (1983). Cognitive styles and their implications for computer-based instruction. Proceedings of the 24th International ADCIS Conference [Special issue]. Journal of Computer-Based Instruction, 9, 241-244.

Martinez, M. E., & Mead, N. A. (1988). Computer competence: The first national assessment. Princeton, NJ: Educational Testing Service.

Mestre, J., & Touger, J. (1989). Cognitive research: What's in it for physics teachers? The Physics Teacher, 27, 447-456.

Miller, A. (1987). Cognitive styles: An integrated model. Educational Psychology, 7(4), 251-268.

Moran, T. P. (1981). An applied psychology of the user. The psychology of human-computer interaction [Special issue]. ACM Computing Surveys, 13(1), 1-11.

Mykytyn, P. P. (1989). Group embedded figures test (GEFT): Individual differences, performance, and learning effects. Educational and Psychological Measurement, 49, 951-959.

Perez, R. S., & Seidel, R. J. (1990). Using artificial intelligence in education: Computer-based tools for instructional development. Educational Technology, 30(3), 51-58.

Pocius, K. E. (1991). Personality factors in human-computer interaction: A review of the literature. Computers in Human Behavior, 7, 103-135.

Post, P. E. (1987). The effect of field-independence/field-dependence on computer-assisted instruction achievement. Journal of Industrial Teacher Education, 25(1), 60-67.

Rieber, L. P. (1990). Animation in computer-based instruction. Educational Technology, Research and Development, 38(1), 77-86.

Rieber, L. P., & Kini, A. S. (1991). Theoretical foundations of instructional applications of computer-generated animated visuals. Journal of Computer-Based Instruction, 18(3), 83-88.

Salomon, G. (1978). On the future of media research: No more full acceleration in neutral gear. Educational Communications and Technology, 26, 37-46.

SAS Institute, Inc. (1985). SAS user's guide: Statistics, version 5 edition. Cary, NC: SAS Institute.

Selker, E. J. (1992). A framework for proactive interactive adaptive computer help. Dissertation Abstracts International, 53, 1473B. (University Microfilms No. DA-9218270)

Shneiderman, B. (1983). Direct manipulation: A step beyond programming languages. Computer, 16(8), 57-69.

Shneiderman, B. (1986). Seven plus or minus two: Central issues in human-computer interaction. In M. Mantei & P. Orbeton (Eds.), Proceedings of CHI'86: Human Factors in Computing Systems (pp. 343-349). New York: Association for Computing Machinery.

Sinatra, R. (1986). Visual literacy connections to thinking, reading, and writing. Springfield, IL: Charles C. Thomas.

Snow, R. E., & Lohman, D. F. (1984). Toward a theory of cognitive aptitude for learning from instruction. Journal of Educational Psychology, 76(3), 347-376.

Stevens, J. P. (1990). Intermediate statistics: A modern approach. Hillsdale, NJ: Lawrence Erlbaum.

Tall, D., & Thomas, M. (1989). Versatile learning and the computer. Focus on Learning Problems in Mathematics, 11(2), 117-125.

Tyler, J. G. (1993). Effects of field dependence/independence and computer expertise on learning application functions in a graphical user interface. In H. Maurer (Ed.), Proceedings of ED-MEDIA '93: World Conference on Educational Multimedia and Hypermedia (pp. 533-540). Charlottesville, VA: Association for the Advancement of Computing in Education.

van der Veer, G. C. (1990). Operating systems in education: Mental models in relation to user interfaces. In A. Finkelstein, M. J. Tauber, & R. Traunmuller (Eds.), Human factors in information systems analysis and design, (pp. 223-241). Amsterdam: Elsevier.

van der Veer, G. C., & Wijk, R. (1990). Teaching a spreadsheet application: Visual-spatial metaphors in relation to spatial ability, and the effects of mental models. In P. Gorny & M. J. Tauber (Eds.), Visualization in human-computer interaction, (pp. 194-208). Berlin: Springer-Verlag.

van Merrienboer, J. J. G. (1988). Relationship between cognitive learning style and achievement in an introductory computer programming course. Journal of Research on Computing in Education, 21, 181-186.

van Merrienboer, J. J. G. (1990). Instructional strategies for teaching computer programming: interactions with the cognitive style reflection-impulsivity. Journal of Research on Computing in Education, 23, 45-53.

Whiteside, J., Jones, S., Levy, P. S., & Wixon, D. (1985). User performance with command, menu, and iconic interfaces. In L. Borman & B. Curtis (Eds.), Proceedings of CHI '85: Human Factors in Computing Systems (pp. 185-191). New York: Association for Computing Machinery.

Wiggs, C. L., & Perez, R. S. (1988). The use of knowledge acquisition in instructional design. Computers in Human Behavior, 4, 257-274.

Witkin, H. A., Moore, C. A., Goodenough, D. R., & Cox, P. W. (1977). Field-dependent and field-independent cognitive styles and their educational implications. Review of Educational Research, 47(1), 1-64.

Witkin, H. A., Oltman, P. K., Raskin, E., & Karp, S. A. (1971). A manual for the Embedded Figures Tests. Palo Alto, CA: Consulting Psychologists Press.

BIOGRAPHICAL SKETCH

John Gordon Tyler was born August 28, 1953 in Saginaw, Michigan. He received the Bachelor of Science with honor in psychology at Michigan State University in June, 1976. From 1977 to 1978, he served as a volunteer with the United States Peace Corps in the Republic of Korea.

John received his Master of Arts in the College of Education at Michigan State University in August, 1979, specializing in instructional development and technology. In December, 1983 he was awarded the Specialist in Education degree at the University of Florida, with a specialization in educational media and instructional design. In December, 1984 he received the Master of Science degree at the University of Florida, majoring in computer and information sciences. In December, 1993 he received the Doctor of Philosophy degree from the University of Florida.

John is an Advisory Programmer with International Business Machines Corporation in Boca Raton, Florida. He joined IBM in 1984 and has specialized in the design and development of advanced personal computer systems software.

John married Umavadee Phaovibul of Bangkok, Thailand, on October 23, 1980. They live in Boynton Beach, Florida, with their daughter, Jessica Ann.

I certify that I have read this study and that in my opinion it conforms to acceptable standards of scholarly presentation and is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

Lee J. Mullally, Chair
Associate Professor of Instruction and Curriculum

I certify that I have read this study and that in my opinion it conforms to acceptable standards of scholarly presentation and is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

Elroy J. Bolduc, Jr.
Professor of Instruction and Curriculum

I certify that I have read this study and that in my opinion it conforms to acceptable standards of scholarly presentation and is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

Douglas D. Dankel, II
Assistant Professor of Computer and Information Sciences

I certify that I have read this study and that in my opinion it conforms to acceptable standards of scholarly presentation and is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

Jeff A. Hurt
Associate Professor of Instruction and Curriculum

This dissertation was submitted to the Graduate Faculty of the College of Education and to the Graduate School and was accepted as partial fulfillment of the requirements for the degree of Doctor of Philosophy.

December, 1993

David E. Smith
Dean, College of Education

Dean, Graduate School