

250  
78

**EFFECTS OF GRAPHICAL USER INTERFACE INCONSISTENCIES  
ON SUBJECTIVE AND OBJECTIVE MEASURES OF USABILITY**

by

Richard H. Miller

Thesis submitted to the Faculty of the  
Virginia Polytechnic Institute and State University  
in partial fulfillment of the requirements for the degree of


MASTER OF SCIENCE


in

Industrial and Systems Engineering

APPROVED:

  
\_\_\_\_\_  
Robert J. Beaton, Chair

  
\_\_\_\_\_  
Paul T. Kemmerling, Jr.

  
\_\_\_\_\_  
James Wilson

April, 1991

Blacksburg, Virginia

C.2

LD  
5655  
V855  
1991  
R567  
C.2

# **EFFECTS OF GRAPHICAL USER INTERFACE INCONSISTENCIES ON SUBJECTIVE AND OBJECTIVE MEASURES OF USABILITY**

by  
Richard H. Miller

Committee Chair: Robert J. Beaton  
Human Factors Engineering Laboratory  
Industrial and Systems Engineering

## **(ABSTRACT)**

This research assessed the effects of inconsistencies in graphic direct manipulation interfaces. Objective and subjective measurement techniques were employed to determine how inconsistencies affected performance in an Apple Macintosh-based computer application called "The Personal Organizer." Three groups of 11 participants, all familiar with the Macintosh computer, were given a set of similar tasks on different versions of the application in a pretest (control version), treatment (control or one of two inconsistent versions), post-test (control version) experimental design.

Performance was measured using two objective measures: task completion time and the number of input control actions. Analysis of variance and correlational procedures were used to interpret these measures. A set of 29 bipolar semantic differentials were used to form a subjective measure of consistency. The linear sum of the scores on a subset of these items was used to create a composite measure of consistency. An analysis of variance procedure was performed on the composite measure, called the Preference Index.

Results show that time and subjective measures are not identical in their ability to discriminate between inconsistent versions of the interface. It is concluded that the inconsistency of the interface has different effects on subject's ability to complete tasks as compared to subject's ability to rate interface consistency. Interface designers should be aware that subjective ratings of interface consistency need to be collected in concert with objective performance measures to assess the effects of graphical user interface inconsistencies upon human performance.

---

# ACKNOWLEDGMENTS

The author would like to thank the committee members for their assistance during this endeavor. My committee chair Dr. Robert Beaton, was instrumental in creating a manageable document and helping me keep my sanity during the revision process. Dr. James Wilson, who helped me originate the topic of consistency and helped me with the many revisions, and Prof. Paul Kemmerling, Jr. whose honesty and open door were always a valuable asset, were both large contributors.

Dr. John Casali, Dr. Robert Williges, Dr. Jeff Woldstad, Mrs. Beverly Williges, and Dr. Terrell were helpful sounding boards for my statistical analyses. Special thanks go to Dr. Walter W. Wierwille who not only was willing to take time out to listen to my problems, but would give me frank opinions and insight during our extended discussions.

Fellow students Sung Han, Kay C. Tan, Min Young Park, and Gerard Jorna all were helpful during the project. Special thanks go to Bob Miller and his trusty red pen. Additionally, I would like to thank Bill Cushman, Wendy Kellogg, Peter Polson, and Andy Cohen for their advice. Martha Sprague, Kathy Menard, and especially Charles “Butch” Nunnally and David Sobotta were helpful in acquiring the necessary equipment for the experiment to flow smoothly. Stan Caplan at the Design Resource Center of the Eastman Kodak Company was helpful in providing a place to develop the software, conduct a literature review, and run pretest subjects.

Finally, I want to thank my mother and father, Betty Carr and Robert Miller for all their guidance and support. It is difficult to convey the love and support that they have given me during my long tenure in college. Hopefully they will gain some satisfaction in knowing that my education will not be wasted.

---

# TABLE OF CONTENTS

	List of Figures	vi
	List of Tables	ix
Chapter 1	Introduction	1
Chapter 2	Background and Literature Review	5
2.1	Characteristics of Consistent Interfaces	8
2.2	Methods for Detecting Inconsistencies	9
2.3	Objective	13
Chapter 3	Method	14
3.1	Participants	15
3.2	Materials and Apparatus	15
3.3	Experimental Design	16
3.4	Experimental Procedure	21
3.4.1	Pre-Experiment Session	21
3.4.2	Experiment Session	21
3.4.3	Data Collection	22
Chapter 4	Results	25
4.1	Time and Step Data Analysis	25
4.2	Time to Complete Tasks Analysis	25
4.3	Subjective Questionnaire (Preference Index)	42
Chapter 5	Discussion	49
5.1	Anomalies in the Objective Measure	49
5.2	Objective Performance Measures	53
5.3	Subjective Performance Measures (Preference Index)	54
Chapter 6	Conclusions	57

Chapter 7	Future Research	60
	References	63
Appendix A:	Macintosh Screening Test for Users	69
Appendix B:	Informed Consent Form	71
Appendix C:	Seeded Consistencies and Inconsistencies	74
Appendix D:	Post-test Questionnaire	89
Appendix E:	Post-Experiment Questionnaire	94
Appendix F:	Tasks Performed By Subjects	98
Appendix G:	General Questionnaire	104
Appendix H:	Instructions for Users	106
Appendix I:	Correlation Matrix: Preference Index	109
Appendix J:	Sample Screen Displays from the Personal Organizer	115
Appendix K:	Technical Information	140
	Vita	141

---

# LIST OF FIGURES

Figure 1.	Different ranges of consistency in a computer environment (Adapted from Robert, 1988).	2
Figure 2.	Moran's three components of a user interface.	3
Figure 3.	The experimental design.	17
Figure 4.	A sample of steps required to complete Task H of Block 6.	23
Figure 5.	Relationship between time and step data (n=198).	26
Figure 6.	Main effect of Set, averaged over Subjects, Groups, and Blocks on time-to-complete tasks. Error bars show $\pm 1$ standard error of the mean (n=66).	29
Figure 7.	Main effect of Block, averaged over Subjects, Groups, and Sets on time-to-complete tasks. Error bars show $\pm 1$ standard error of the mean (n=66).	30
Figure 8.	Block x Sets interaction, averaged over Subjects and Groups, on time-to-complete tasks. Error bars show +1 standard error of the mean (n=33). Plotted as a function of Set.	30
Figure 9.	Group x Set interaction, averaged over Blocks and Subjects, on time-to-complete tasks. Error bars show +1 standard error of the mean (n=22). Plotted as a function of Set.	31
Figure 10.	Group x Set interaction, averaged over Blocks and Subjects on time-to-complete tasks. Error bars show +1 standard error of the mean (n=22). Plotted as a function of Group.	34
Figure 11.	Block x Group interaction, averaged over Set and Subjects on time-to-complete tasks. Error bars show +1 standard error of the mean (n=33). Plotted as a function of Group.	38

Figure 12.	Group x Block interaction , averaged over Set and Subjects on time-to-complete tasks. Error bars show +1 standard error of the mean (n=33). Plotted as a function of Block.	39
Figure 13.	Block x Set interaction, averaged over Group and Subjects, on time-to-complete tasks. Error bars show +1 standard error of the mean (n=33). Plotted as a function of Block.	40
Figure 14.	Main effect Set, averaged over Group and Subjects on PI. Error bars show +1 standard error of the mean (n=33).	44
Figure 15.	Group x Set interaction, averaged over Block and Subjects on PI. Error bars show +1 standard error of the mean (n=11). Plotted as a function of Set.	45
Figure 16.	Group x Set interaction, averaged over Block and Subjects on PI. Error bars show +1 standard error of the mean (n=11). Plotted as a function of Group.	47
Figure 17.	Set x Blocks x Group interaction, averaged over subjects. Error bars show +1 standard error of the mean (n=11).	50
Figure 18.	Time to complete Block 1 of Set 1 and 2 for each subjects (n=1). Note different scales for each graph.	52
Figure J-1.	Consistent version- Main Screen function.	116
Figure J-2.	Consistent version- Address Roledex function.	117
Figure J-3.	Consistent version- Phone Setup function.	118
Figure J-4.	Consistent version- Area Code function.	119
Figure J-5.	Consistent version- Six-month (Yearly) Calendar.	120
Figure J-6.	Consistent version- Weekly Planner function.	121
Figure J-7.	Consistent version- To Do list function (with delete box open).	122

Figure J-8.	Inconsistent version A- Main Screen function.	123
Figure J-9.	Inconsistent version A- Address Roledex function.	124
Figure J-10.	Inconsistent version A- Phone Setup function.	125
Figure J-11.	Inconsistent version A- Area Code function.	126
Figure J-12.	Inconsistent version A- Six-month (yearly) Calendar.	127
Figure J-13.	Inconsistent version A- Weekly Planner.	128
Figure J-14.	Inconsistent version A- To Do list function (with delete box open).	129
Figure J-15.	Inconsistent version B- Main Screen function.	130
Figure J-16.	Inconsistent version B- Address Roledex function.	131
Figure J-17.	Inconsistent version B- Phone Setup function.	132
Figure J-18.	Inconsistent version B- Area Code function.	133
Figure J-19.	Inconsistent version B- Six-month (yearly) Calendar.	134
Figure J-20.	Inconsistent version B- Weekly Planner.	135
Figure J-21.	Inconsistent version B- To Do list function (with delete box open).	136
Figure J-22.	The consistent (top), inconsistent (middle) Phone Dial windows, and the Phone List window (bottom).	137
Figure J-23.	A demonstration of the help system for the menu bar.	138
Figure J-24.	The menus used in all versions of the application (except the style menu).	139

---

# LIST OF TABLES

TABLE 1.	Commands for Cut, Copy, Paste, and Undo in Various Applications	7
TABLE 2.	A Framework for Detecting Inconsistency (Adapted from Robert, 1988)	11
TABLE 3.	Sample Attributes of a Bibliographic System (Adapted from Robert, 1988)	11
TABLE 4.	Description of Tasks Found in the Blocks of Tasks	18
TABLE 5.	Analysis of Variance Summary Table for Performance (Time)	27
TABLE 6.	Results of the Newman-Keuls Test on the Main Effect of Set	29
TABLE 7.	Results of the Simple-Effect F-tests on Block at the Different Levels of Set	31
TABLE 8.	Results of the Simple-Effect F-tests on Group at the Different Levels of Set	32
TABLE 9.	Results of the Newman-Keuls Test on the Effect of Group Within Set 1	32
TABLE 10.	Results of the Newman-Keuls Test on the Effect of Group Within Set 2	32
TABLE 11.	Results of the Simple-Effect F-tests on Set at the Different Levels of Group	35

TABLE 12.	Results of the Newman-Keuls Test on the Effect of Set Within Group 1	35
TABLE 13.	Results of the Newman-Keuls Test for the Effect of Set Within Group 2	36
TABLE 14.	Results of the Newman-Keuls Test for the Effect of Set Within Group 3	36
TABLE 15.	Results of the Simple-Effect F-tests on Block at the Different Levels of Group	38
TABLE 16.	Results of the Simple-Effects F-tests on Group at the Different Levels of Block	39
TABLE 17.	Results of the Newman-Keuls Test on the Effect of Group Within Block 1	40
TABLE 18.	Results of the Simple-Effect F-tests on Set at the Different Levels of Block	41
TABLE 19.	Results of the Newman-Keuls Test on the Effect of Set Within Block 1	41
TABLE 20.	Results of the Newman-Keuls Test on the Effect of Set Within Block 2	41
TABLE 21.	Spearman Correlation Coefficient ( $r^s$ ) for Bipolar Adjective Pairs with the Consistent-Inconsistent Item in the PI	43
TABLE 22.	Analysis of Variance Summary Table for Preference Index (PI)	44
TABLE 23.	Results of the Newman-Keuls test on the Main Effect Set	45
TABLE 24.	Results of the Simple-Effect F-tests on Group at the Different Levels of Set	46

TABLE 25.	Results of the Newman-Keuls Test on the Effect of Group Within Set 2	46
TABLE 26.	Results of the Simple-Effect F-tests on Set at the Different Levels of Group	48
TABLE 27.	Results of the Newman-Keuls Test on the Effect of Set Within Group 2	48
TABLE 28.	Results of the Newman-Keuls Test on the Effect of Set Within Group 3	48
TABLE C-1.	Seeded Inconsistencies	75
TABLE C-2.	Defining the Classes of Inconsistencies	87
TABLE C-3.	Summary of Inconsistencies by Class	88
TABLE C-4.	Summary of Inconsistencies by Topic	88
TABLE I.	Spearman Correlations and $p$ -values for the Preference Index Items	109

---

# CHAPTER 1

---

## INTRODUCTION

---

*To design the user interface of a system  
is to design the user's model*

Thomas P. Moran, 1981

---

The design of graphical computer interfaces involves many complex issues. The issue of consistency is a primary criterion for usable interface designs. Consistency across applications allows users to transfer existing computer skills to new applications. Consistency can reduce training time and user errors, while increasing the user's opinion of software quality (Kellogg, 1987; Polson, 1988). In addition, the wide availability of various computer platforms increases the need for consistency across computer hardware.

Figure 1 shows ranges of consistency for a computer environment (adapted from Robert, 1988). An interface design that is consistent within one program, defined in Range 1, might not be consistent with other programs in Range 2. It is not clear when specific consistency issues that appear at one range take priority over inconsistencies at other ranges. Consistency created at one level may create inconsistencies at another level. The ability to determine where and how to make consistent software is an important issue to interface designers. With better understandings of consistency, designers will build graphical interfaces that perform better and be accepted by the end users.

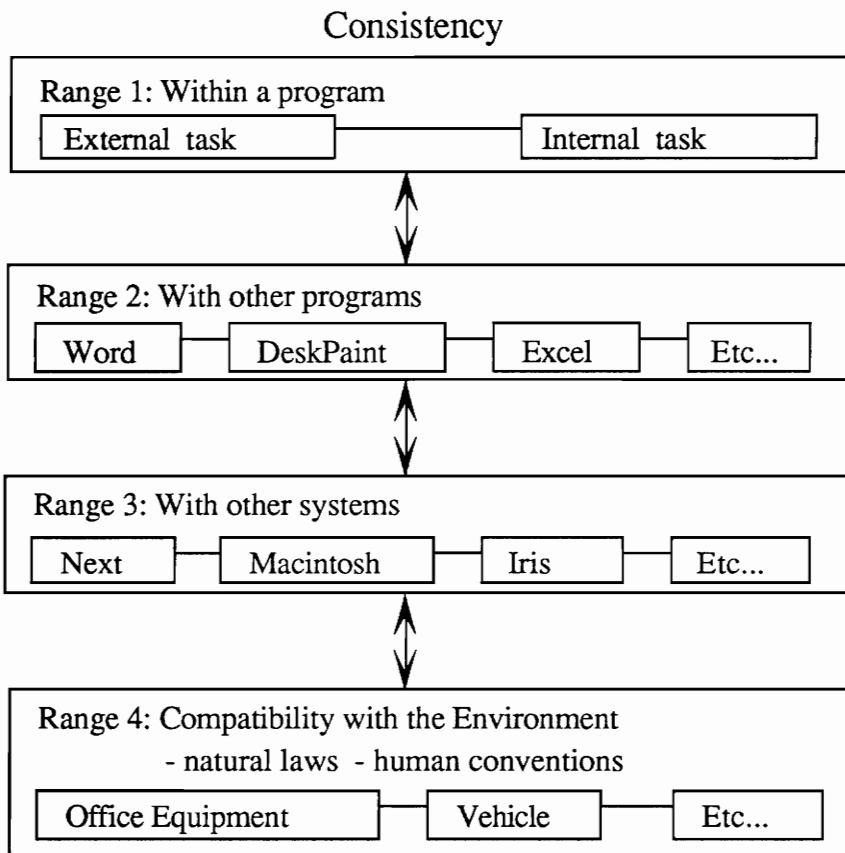


Figure 1. Different ranges of consistency in a computer environment (Adapted from Robert, 1988).

Consistency is defined as “system usage that meets the user’s expectations.” As shown in Figure 1, external tasks refer to actions that allow links to system software or other applications. Internal task consistency defines those tasks specific to the application. Consistency with other systems relies on document conversion capabilities and does not rely on functional or interface similarity. Today the trend is to make applications and documents transportable, thus expanding the scope of Range 3. Compatibility with the environment relies on consistency with external conventions. This is achieved with metaphors in the software design.

Moran's Command Language Grammar (CLG) (1981) provides a framework for researchers to stratify consistency issues (Kellogg, 1987; Robert, 1988; Bouchard and Robert, 1989). CLG defines a series of rules that state the operations required to complete tasks in a system. Moran divides a user interface into three component parts, each with two levels (Figure 2). The overlap of rings in Figure 2 reflects the interactions associated with the CLG. Polson stated that strict decomposition of an interface into the six independent levels probably is futile. Items that fit in one component can also affect the other components – this is represented by the overlapping areas. Polson (1989) pointed out that design decisions ripple through the CLG. A decision that applies to one operation or task in the interface can also affect other parts of the interface.

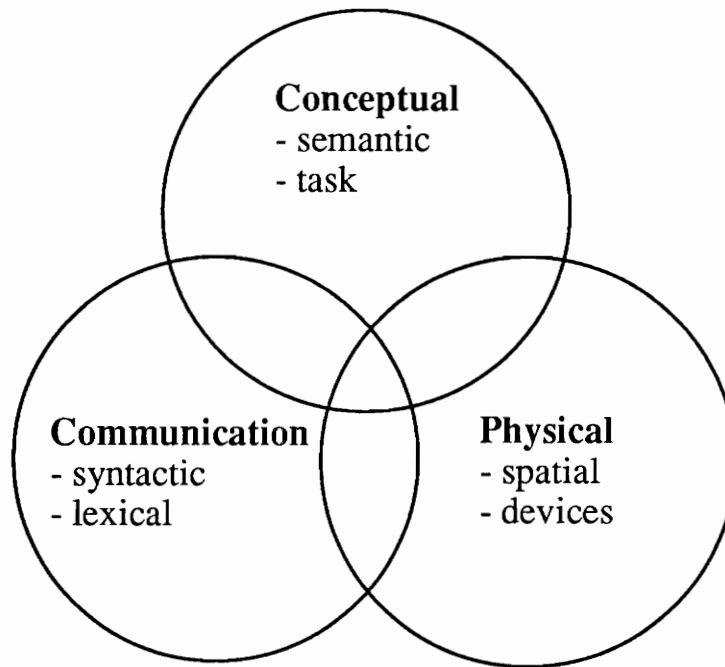


Figure 2. Moran's three components of a user interface.

Icon placement, naming conventions, and command sequences are a few issues that need to be classified in the CLG. The placement of these issues into CLG levels is based on where the primary effect of the inconsistency occurs in the design. This statement does not imply that the placement of an inconsistency into a specific level affects only that level. The description of an inconsistency as belonging to a particular level is based on how the inconsistency is identified. No formal methods are available at present to describe and classify inconsistencies.

The CLG shows the relationship among the physical, conceptual, and communication components. The physical component was not addressed in Moran's work, while Robert (1988) did use it to find inconsistencies in a user interface. The conceptual component of the CLG includes the task and semantic levels, which are both based on abstractions of system functions. The task level organizes the user's needs into a structure that defines what events occur in the system. The semantic level defines the system's attributes and operation without defining the implementation. The communication component consists of the syntactic and lexical level. The syntactic level recodes the semantic level concepts into actual commands, arguments, and state variables. The lexical level defines the actions and system rules needed to invoke the defined syntax.

Clarification of how inconsistencies at these levels impact objective and subjective measures will help user interface designers. Three versions of an application, called "The Personal Organizer," were used in a pretest, treatment, post-test experimental design. The design used three groups of 11 subjects each, all familiar with the Apple Macintosh computer. Each group was given a set of similar tasks on a control version, followed by another set of tasks on the consistent (control) version, or one of two inconsistent versions. The post-test followed with similar tasks using the control version for all three groups. Subjective ratings and objective measures of time and input control actions were collected to identify the effect of inconsistencies in the graphical direct-manipulation interface. The application versions varied in terms of the communication, conceptual, and physical levels of consistency.

---

# CHAPTER 2

---

## BACKGROUND AND LITERATURE REVIEW

Many technical areas contribute to the design of computer applications. General handbooks and guidebooks for user interface design can help create properly designed interfaces (Brown, 1988; Shneiderman, 1987; Smith and Mosier, 1984). Shneiderman's (1987) first golden rule of dialog design is to "strive for consistency." Shneiderman stated that "consistent sequences of actions should be required in similar situations, identical terminology should be used in prompts, menus, and help screens, and consistent commands should be employed throughout" (1987, p. 61). This rule is good in theory, but difficult to apply.

Some user interface design guidelines were designed for specific computer environments. The *Human Interface Guidelines: The Apple Desktop Interface* (Apple, 1987) draws on some research for defining the implementation of the Macintosh environment. Digital Equipment Corporation published the *XUI Style Guide* (1988) to assist programmers in developing an X User Interface for DecWindows. IBM published *Common User Access Panel Design and User Interaction* (1987) for its System Application Architecture (SAA). Every computer company uses unique rules and conventions for specifying their graphic user interfaces. Consistency is stressed in these works because it is an essential feature of effective user interface designs.

In a broad sense, rules are requirements, style guides define specific implementations, and guidelines state reasonable and practical implementation practices. Each of these documents have a common focus: the consistency of the user interface.

Carroll, Mack, and Kellogg (1988) discussed the use of interface metaphors in user interface design. Computer applications, designed to handle work previously done without computers, use metaphors to draw on previous knowledge and methodologies that might lead to a positive training effect and a sense of familiarity for the user. Since many problems can be solved with computers, the link between the external environment and the computer is useful in helping users adapt. The ability to learn and use a product is usually based on previous experience. Whether the person has ever seen or heard of the product, the user will use his or her experiences to form a knowledge base for new experiences. This knowledge base can stem from “typewriters, calculators, or video games, as well as from computer systems. A good user interface relies on these experiences” (IBM, 1987, p. 9). The use of metaphors creates links from this knowledge base to computer applications. Similarly, consistency within a specific application can be used to help a user map the knowledge of one function to untried tasks within the application. This mapping might involve a common metaphor, a specific interaction style, similar command syntax, or a common procedure.

Kieras and Bovair (1986) showed that learning time for a new system is reduced if the user is familiar with another system that employs similar rules. This also can be applied to applications found within the same system. They determined that learning a new set of command keys for every application is less efficient than transferring a single set of rules to new environments. Reisner (1981) working on similar research concluded that “structural” consistency (i.e., the number of rules needed to define a system procedures) helped user performance. She observed that users made “expectation errors” in the treatment version, where they applied new, learned procedures to new objects. No statistical analysis was performed by Reisner.

In a command line environment, some programs use different methods for creating command syntax. Table 1 shows four common commands; Cut, Copy, Paste, and Undo. The command syntax shown in Table 1 contrasts with guidelines set forth for the Macintosh (Apple, 1987). Macintosh programs with these edit functions use the same set of keys (Command Z, X, C, V for Cut, Copy, Paste, and Undo, respectively). This means that despite the program used in the Macintosh environment, there usually will be a

knowledge base that can be applied to the new application. This is an example of applying consistency in Range 2 of Robert’s levels of consistency.

TABLE 1. Commands for Cut, Copy, Paste, and Undo in Various Applications

IBM PC Program	Cut	Copy	Paste	Undo
Word Perfect 5.0	Ctrl F4 then #1	Ctrl F4 then #2	Ctrl F4 then #3	Ctrl F4 then #4
MS Write	Shift+Del	Ctrl+Ins	Shift+Ins	Alt+Bksp
Volkswriter 3.0	Alt F8	none	Alt F6	None
PFS Write	F4	F3	F6	Ctrl F4

Designers can incorporate consistency in applications using a variety of methods. Lotus 1-2-3 (a popular IBM PC spreadsheet program) is a good example. Cohen and Schirmer’s (1989) review of Lotus 1-2-3 pointed out five major features that contribute to its dominance in the industry. One feature is the “implementation of the <Escape> key [that gives] the user a consistent, non-destructive method of retracing his steps or correcting an error” (Cohen and Schirmer, 1989, p. 9).

Koritzinsky (1989) concluded that it was more important to be consistent with user’s perspective of an interface than to constrain the shape of a new icon. The author was referring to a circular clock icon. A square clock icon would be consistent with the other system icons, while a circular clock icon would be consistent with the shape of watches and wall clocks. No experimental evidence was cited to indicate whether the shape of the object affects subjective or objective performance. Making the clock consistent with “square” digital clocks was not addressed.

Whiteside, Jones, Levy, and Wixon (1985) examined user performance with command, menu, and iconic interfaces. They tested a system as a whole because it “represents a

design team's best overall solution to optimizing many variables, [and] we claim that a test of the system as a whole is more fair and valid than a (perhaps impossible) decomposition into component variables" (1985, p. 185). They suggest that, due to the complexity of a user interface, attempts at decomposition probably are futile.

Ham (1987) stated that "no matter how much trouble it is, take pains to be consistent in every way possible: punctuation, significance of colors, mode of input, location of messages..." (1987, p. 114). On the other hand Grudin (1989) pointed out many instances where consistency is not the most important issue to be resolved. Other researchers (Wolf, 1989; Briggs; 1989; Grudin, Ehrlich, and Shriner, 1987) agree that it is better to strive for consistency. Ham clearly addresses the need for an interface to be consistent with the users model. Unfortunately, Ham's theory proposed in that statement is problematic, since it might not be realistic to be consistent in every possible way. Something consistent in one context probably will be inconsistent from a different point of view. In addition, there are costs associated with being consistent: user testing, the system timeline, and designer effort. A possible solution is to allocate resources toward creating consistency in aspects of the interface where performance would be affected by its absence, than creating consistency in every aspect of the interface. Grudin (1989) suggests that one should center effort on understanding the tasks of the worker, rather on just being consistent. Grudin pointed out that consistency is not the goal, but only a method for achieving usability in a design. An appropriate design should allow for consistency within the context of the workers' tasks.

## **2.1 Characteristics of Consistent Interfaces**

Even a simple definition of consistency is difficult to apply. A dictionary defines consistency as an "agreement with what has already been done or expressed; conformity with previous practice" (Mish, 1983, p. 303). Often, a designer may be trying to match an interactive style to a user's cognitive representation of the system's workings. The user's representation might not include any previous computer practice and would be based only on experiences from non-computer activities (like reading a restaurant menu or using a typewriter). This type of agreement can play an important role in the learning process. Bouchard and Robert defined consistency as "the quality of the system which respects the user's expectations" (p. 2, 1989). Some research has gone further in this vein by

attempting to build methods for detecting and understanding how consistency effects the usability of an interface.

Hammer, Kunin, and Schoichet (1983) concluded that a consistent interface is one “in which different things are handled in similar ways with similar syntax” (p. 129). The word “things” in this statement refers to the object in a verb-object syntax, where actions were performed on system objects. They cite four criteria for designing a user interface: the interface should be natural, easy to learn, easy to use, and consistent. Although the authors do not cite empirical data to support their criteria, they claim that consistency is the cornerstone of a usable system.

Blake (1986) defined a consistent interface as having five characteristics:

- Predictable - Users anticipate what the system will do.
- Dependable - The system fulfills the user’s expectations.
- Habit-forming - The system encourages development of behavior patterns.
- Transferable - Habits developed in one context apply in new situations.
- Natural - The interface is consistent with the user’s understanding.

These attributes can be part of the communication, conceptual, and physical components of an interface. All could be used as synonyms for consistency.

## **2.2 Methods for Detecting Inconsistencies**

In 1988, a group of experts in Human Computer Interaction (HCI) gathered to discuss consistency but could not create a good definition (Grudin, 1989). Researchers have created numerous methods for explaining and quantifying HCI. Some researchers have found generic methodologies for comparing interface designs (Bachman, 1989; Roberts and Moran, 1982). Tullis (1983) developed a metric for determining layout complexity, while Card, Moran and Newell (1983) developed the well known Keystroke Level Model (KLM). Reisner (1981) used formal grammatical descriptions to predict ease of use and Robert and Wang (1989) continue to build statistical metrics for evaluating the importance of inconsistencies. Others have tried to quantify and classify consistency (Barnard,

Hammond, Morton, Long, and Clarke, 1981; Berry, 1988; Bouchard and Robert, 1989; Robert 1988). Although these efforts are laudable, true taxonomies for defining consistency are still only works in progress.

Briggs (1989) investigated the use of a command wheel diagram, and proposed that it can “provide a suitable meta-language with which to illustrate the consistent properties of a command set” (p. 160). The study looked at the consistency of WordPerfect and Word, both commercial word processors. The command wheel is used to map out the word processor’s commands for cursor control and deletion tasks. Missing spokes in the wheel identified inconsistencies in the command syntax. Briggs does not define the command wheel, but explains it by example. The command wheel method is difficult to adapt for commands beyond the cursor control paradigm. Briggs questioned if consistency was dependent on a person’s awareness of the underlying rule structure, or if a user would benefit from consistency even if he or she were not aware of the inconsistency in the GUI.

Robert (1988) presented a method for detecting inconsistencies in user interfaces. The first step entailed defining the system. Computer-based bibliographic information retrieval system using “IF (user action) THEN (system response)” type descriptions were used in Robert’s work. Robert found that this method, by itself, was not effective for detecting inconsistencies, and expanded on the six description levels from the Command Line Grammar (CLG) model proposed by Moran(1981) to evaluate the interface. Moran’s model has three main components: Conceptual, Communication, and Physical, as discussed in the Introduction. Each main component has two underlying levels. Robert views these levels as “six different angles for examining... precise issues where consistency comparisons can be made” (1988, p.360). Table 2 shows the three components, six levels, and some interface components analyzed in the bibliographic database system.

The attributes in Table 3 were generated by comparing differences in interface components found within a specific level. For example, Robert defined command names that were verbs, nouns, adjectives, expressions, etcetera as the “type of identification” into the lexical level. When command names were abbreviated differently, the “type of abbreviation” was created as a lexical attribute.

TABLE 2. A Framework for Detecting Inconsistency (Adapted from Robert, 1988)

COMPONENTS		INTERFACE COMPONENTS							
		command	menu	prompt	help facility	error message	screen title	feedback	et cetera
<b>Conceptual</b>	task								
	semantic								
<b>Communication</b>	syntactic								
	lexical								
<b>Physical</b>	spatial layout								
	devices								

TABLE 3. Sample Attributes of a Bibliographic System (Adapted from Robert, 1988)

<b>Syntactic</b>	number of arguments position of arguments default option (to skip steps) shortcuts (to go faster) command nesting (grouping commands under one name) recursion (to use same command w/o selecting it again) scope of recursion (over all or some arguments) time at which commands are shown, vanished, dimmed, or highlighted
<b>Lexical</b>	language (English, French) type of delimiters and type styles blinking and blinking speeds identification (noun, verb, expression) representation (iconic, alphabetic, numeric) number of words, word length and abbreviations abbreviations, capitalization, font size and style
<b>Spatial</b>	position on a support (visual display, keyboard, mouse) menu item position (grouping vs. separation) alignment, indentation

Robert did not systematically determine the number or type of attributes. He states, “to a large extent, the number of attributes that is considered may determine the power of the method for detecting inconsistencies” (1988, p. 360). No previously created list of possible attributes was referenced, and identification of inconsistencies depended on the evaluators.

Once these attributes were determined, differences within the same category and across each attribute at specific levels were compared. Differences between the components were labeled as possible inconsistencies. Differences then were classified as inconsistencies “as long as the classification made sense in the context of the task. Indeed, the task was used as the ultimate guide because it is responsible, to a large extent, for the structure and operation of a system” (1988, p. 360). Although not systematically repeatable, this heuristic method is one of the few examples in the literature that tests inconsistencies in an interface.

On a subjective level, all but one participant reported that the system was easy to learn and use, with a few negative feelings. So, while there were many inconsistencies, as defined by Robert’s method, it was not reflected in the subjective evaluations of the users. Decrements to objective performance (i.e., time to complete tasks, number of errors) from the inconsistencies was not measured. It is unclear if inconsistencies not found by the subjects were (1) masked by other interface features, (2) not salient enough to be noticed, or (3) not within the scope of the users’ objectives.

Kellogg’s (1987) experiment addressed consistency in the user interface and its effect on user performance using Moran’s (1981) CLG framework. In Robert’s (1988) work the levels were crossed with specific components of the interface (Table 2), while Kellogg divided the interface components into two distinct groups at the conceptual level: internal and external consistency. Kellogg looked at conceptual rather than procedural consistency. Procedural consistency is generally found in design guidelines (e.g., the assignment of function keys or the placement of a menu item). Conceptual consistency was defined in terms of the “internal coherence of a system’s structure and the nature of the mapping from user-level goals to system procedures” (Kellogg, 1987, p. 389).

In Kellogg’s (1987) experiment, three groups of users were assigned two sets of tasks, each group having a different version of the interface with analogous tasks. Versions were consistent, inconsistent, and consistent plus salient. The versions all performed the same functions, but varied in the semantic procedure. In the first trial a significant difference was found between the inconsistent and the two consistent interfaces, but in the second trial the effects were not significant. Subjects also recalled more correct procedures from the consistent interfaces and recalled fewer inappropriate procedures. An important result from this experiment was that the subjects rated the consistent interfaces more positively on 15 of 17 bipolar adjective scales taken from Coleman, Williges and Wixon (1985). Kellogg

summarized that users of the consistent system performed tasks more quickly, recalled system procedures almost perfectly, and correctly inferred untried procedures. Users in the consistent group also had greater confidence in their abilities to use the system.

Kellogg's conclusions were similar to Payne and Green's (1986) findings that consistency in system design allowed users to generalize rules and actions from known procedures to novel tasks. Payne and Green concluded that consistency can impact the performance of users without users knowledge. In an experiment on syntactic consistency, Payne and Green (1986) found that a high-level consistent syntax for three types of tasks was the easiest to learn of three syntax structures tested. A second design incorporating two rule structures used in the same tasks was harder to learn than a design with a distinct command structure for each category. The ability to use a consistent set of syntax in a variety of tasks proved to be the most effective method for inter-application consistency.

In conclusion, consistency and its effect on the creation of usable interface designs is defined poorly and not well understood. The general viewpoint is that consistency is good, but it is not known how to determine where and in what context something should be consistent. Also, the use of information from users in a software design process is warranted, but knowing when to ask for their input and what questions should be addressed has not been adequately researched. The present research looked at how subjective and objective performance differ in graphical user interfaces.

### **2.3 Objective**

The present research addressed the following question concerning consistency in graphic user interface design.

How do time, keystroke and subjective measures differ with variations in the consistency of an interface?

---

# CHAPTER

# 3

---

# METHOD

## 3.1 Participants

A screening test was administered to 112 “experienced” users to determine their level of proficiency on a Macintosh computer. *A priori* assumptions were not made about the participants’ level of expertise, although experience with the Macintosh and HyperCard was required to pass the screening test. The screening test was designed by human factors engineers and pilot participants during the iterative design of the control version of the application. Passing of the screening test was defined as the top scoring 33 participants that were administered the test. This screening test is outlined in Appendix A. The questions were chosen to represent an experienced Macintosh user’s knowledge.

Thirty-three participants (29 males and 4 females) volunteered for the actual experimental study. Each participant spoke English as their native language, and read and signed an informed consent form (Appendix B). Sixteen undergraduate students, 12 graduate students, 3 faculty members, and 2 staff members from Virginia Polytechnic Institute and State University (VPI) participated. Participants possessed an average of approximately four years of experience with Macintosh computers. The age of participants ranged from 18 to 39, with a mean age of 25 years.

Each subject participated for approximately 2-1/2 hours in one session, not including the screening test. Participants received compensation of software (15 disks of shareware), or six disks and six dollars.

## 3.2 Materials and Apparatus

Three versions of an application were created using SuperCard 1.5. Hardware consisted of a Macintosh IIfx with a Radius high resolution color monitor (1152 x 882 pixel addressability), an 8-bit color board, 8 megabytes of RAM, and an Apple mouse. The System (v. 6.0.5) operated in the MultiFinder mode. The application, called “The Personal Organizer,” included a datebook, notepad, calendar, facsimile transmitter, address book, area-code finder and telephone dialer.

Functionally, all versions of the application performed the same operations. The consistent version had “no” inconsistencies. It was refined and iterated by human factors engineers and graphic designers from the Eastman Kodak Company in Rochester, NY, human factors graduate students from VPI, and the author. This was done by designing the control version to follow existing guidelines (Apple, 1987; Apple, 1989). Twelve pilot participants from the Eastman Kodak Company and VPI, each having 5 to 15 hours of experience with the application, were used to refine the consistent version.

The two inconsistent versions embodied unique sets of inconsistencies. Appendix C outlines the differences between the three versions of the interface. Some inconsistencies in Appendix C were similar in both inconsistent versions of the interface – in other cases it varied between all three versions. Sometimes a consistent “state” could only have one inconsistent state. For example, the *Find* function remembered the previously found word in the consistent version. In the inconsistent versions, the *Find* function did not remember the previously listed text. Both inconsistent versions varied similarly in this “inconsistency.”

A modified Questionnaire for User Interface Satisfaction (QUIS) was used to rate the interfaces after participants completed each of the three sets of tasks (Chin, Diehl, and Norman, 1988). Bowers (1990) cited others that reported users of the QUIS were not reading every question when the scale ranged from left to right (negative implication to positive). The questions were bipolar semantic differentials on a 1 to 7 rating scale. Bowers modified the structure of the scale without changing the questions. Bowers concluded that users were more thoughtful in their answers. Therefore, the order of the questions and the

position of the positive and negative anchors to the scale were randomly assigned for each questionnaire used in the experiment. All participants received the three questionnaires in the same order (See Appendix D for a sample of the questionnaire). The questions were used to determine if consistency affected the opinions of the participants.

Upon completion of the three sets of tasks and questionnaires, an additional set of subjective measures consisting of bipolar scales and interview-style questions were administered (Appendix E). These measures were for research in progress and were not analyzed in this experiment. However, they are included for completeness. Bipolar scales were used to collect preference data for the application and the interface. An unstructured set of 22 questions at the end of the experiment were used to elicit information about specific attributes of the treatment version that may have been missed by the bipolar scales. The answers were used to assess the type and number of inconsistencies identified by the participants. Additional questions were asked to obtain the subject's age, type of computer use, and computer experience.

### **3.3 Experimental Design**

Participants were assigned randomly to a group (Figure 3). The first set of tasks (a set has two blocks of eight tasks and a questionnaire) lasted approximately 60 minutes. The next two sets lasted approximately 30 minutes each, with rest periods after every block. The blocks of tasks were similar across sets. For example, one type of task was for placing a name, address, and phone number in the Address Rolodex, but each occurrence used a different name, address, and phone number. The blocks were presented in a random order for each participant, as were tasks within Blocks.

Although the sets of tasks were identical, the two blocks of tasks within each set were intentionally different. The types of tasks found in each block are defined in Table 4. A complete listing of the 48 tasks used in the experiment is presented in Appendix F. There were three reasons for choosing a variety of tasks. First, participants would become more familiar with the extent of consistency in the interface when exposed to a variety of tasks. Second, by having tasks repeated infrequently, participants would become less bored. Third, participants also could learn about more of the interface without being able to anticipate tasks.

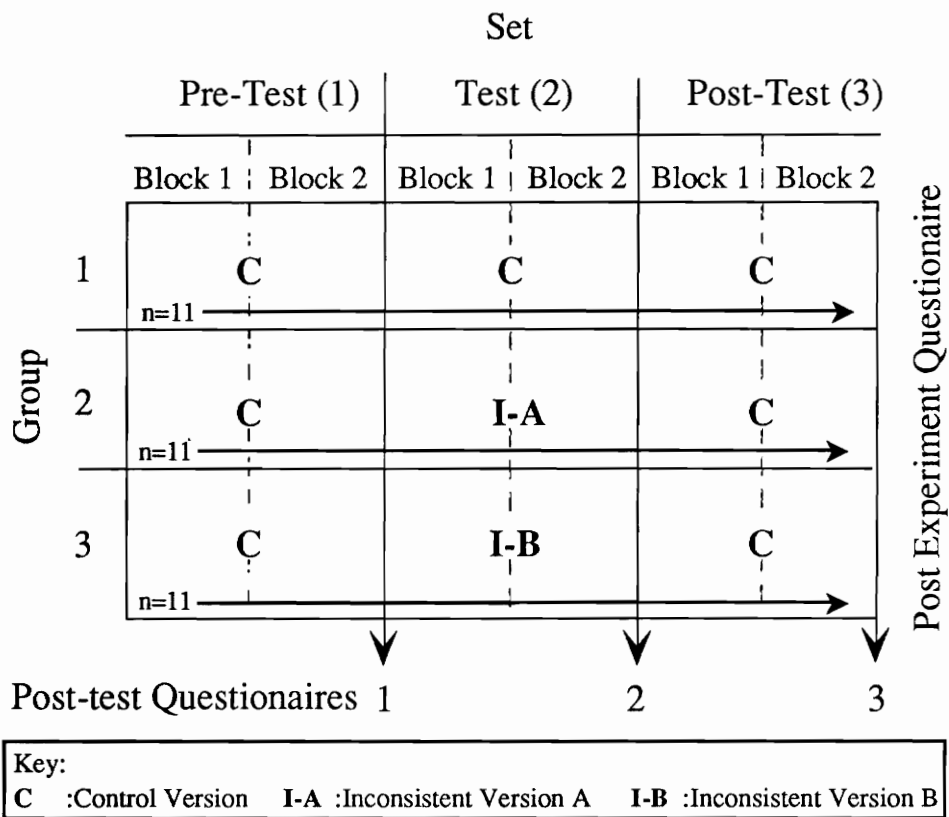


Figure 3. The experimental design.

TABLE 4. Description of Tasks Found in the Blocks of Tasks

Tasks for Blocks 1, 3 and 5	Tasks for Blocks 2, 4 and 6
<b>A-</b> Move a meeting and place a conference in the Weekly Planner.	<b>B-</b> Place a meeting in the weekly planner twice (two weeks apart).
<b>C-</b> Find a phone number and other information and enter it into two places in the Weekly Planner, without going to the Address Rolodex.	<b>D-</b> Find the correct area code for a specific city and place it in front of both phone number found on two Address Rolodex cards that show that city as the address.
<b>E-</b> Add or change two settings in the Phone Setup function.	<b>F-</b> Sort the Address Rolodex by last name.
<b>G-</b> Delete two specified pages of the To Do function and add some comments in a specific size and font style to a specific page.	<b>H-</b> Add two pages to the To Do function at a specific place in the function and add some wording in a specific size and font style to a specific page.
<b>I-</b> Add a set of cities to a specific region in the Area Code function.	<b>J-</b> Add an address card to the Rolodex with a person's name, company, address, and office phone number.
<b>K-</b> Delete the Address cards for three individuals.	<b>L-</b> Find a person's telephone number, without going to the Address Rolodex, and dial the phone number with a specific phone setting from a specified page in the Weekly Planner or To Do function.
<b>M-</b> Correct the name of a company as it appears in the weekly planner, with the correct information found in the Address Rolodex.	<b>N-</b> Find the correct area code for a phone number, specified in the Weekly Planner, by comparing it to what was in the Area Code function.
<b>O-</b> Get help for a specific icon in a specific function.	<b>P-</b> Get help for a specific icon in a specific function.

Tasks were matched across rows for time and steps, even if the length of the instruction is short the time and steps to completion are equal.

Two performance-based dependent measures (time and number of steps to complete each task) were recorded. These were computed from the sum of the individual task completion time and step data. Participants were not allowed to go onto the next task until they had completed the previous task. Less than 1% of the tasks were not completed correctly (less than 16 out of 1584). No correction factor was employed to account for the small number of uncompleted tasks, since participants would miss one small part of the task (e.g., not making a word Bold, etc.). Steps were defined into one of 3 groups; completed with quick keys (command keys), completed with menu selections (using the mouse), or completed by clicking on icons (using the mouse). Inappropriate steps were not classified as errors due to the difficulty in establishing error types (i.e., how to define steps taken to complete the tasks “the long way”, or how to define when a participant starts to explore other alternatives to completing a task).

An analysis of variance was performed on a 3x3x2 mixed factor design. A control group was used to account for learning effects associated with using similar software applications. Groups (3 levels) and Subjects within Groups (11 participants per group) served as between-subject variables. Set (3 levels) and Block (2 levels) served as within-subject variables. Set and Block were both completely crossed with the between-subject variables. A block consisted of a set of eight tasks (Table 4). Groups 1, 2 and 3 received the control versions, inconsistent version A, inconsistent version B, respectively in Set 2.

Blocks in Set 1 were training sessions to familiarize Groups with the interface. This helped control for differences between groups before being administered the treatment version. It also gave the participants a basis for making judgements on inconsistencies found in the treatment version. Set 2 exposed Groups 2 and 3 to the inconsistent versions of the interface. Any changes in performance between groups could then be associated with the onset of the inconsistent versions. Set 3 would identify the effect of transfer of training from the inconsistent versions in Set 2.

A short subjective evaluation (Appendix D) was performed after each trial. The participants rated the interface on a set of 29 bipolar semantic differential scales. The data were analyzed using an analysis of variance (ANOVA) on a subset of the questionnaire. The 29 questions from the first post-test questionnaire were used to determine the bipolar adjective pairs that related to the consistency of the interface.

The Consistent-Inconsistent (C-I) item in the questionnaire was the item of interest in this experiment. The C-I item was correlated with the other 28 items to find items that would reflect a degree of similarity. Since, the anchors used to describe the scales were different, we can consider them as testing a different, but similar, set of preferences. Items that had a correlation of  $p < 0.05$  were included in the scale. The scores from the questionnaire (from 1 to 7) for the correlated items were summed to define a composite measure of the subjects preference of the interface. The resulting Preference Index (PI) would be a “general” measure for determining the consistency of an interface. Other researchers have used similar methodologies to determine a composite subjective measure (Casali, Lam, and Epps, 1987; Epps, and Casali, 1985; Park, 1989).

A 3x3 mixed-factor ANOVA was performed on the PI values. Set (3 levels), the within-subject variable, was crossed with Groups (3 levels), the between-subject variable. The PI was the dependent measure. Comparisons of the changes in PI ratings due to the version presented in Set 2 were made.

Participants were allowed to analyze the version used during the second set to help them in completing the questionnaire in Appendix E. These were treated to a separate analyses in the future report and is explained here for the sake of completeness. Free response questions addressed the consistency of the interface. Participants were given 22 topics to rate as either consistent or inconsistent. A score of 1 was assigned to a rating of consistent, and a score of 2 for inconsistent. When the participant did not rate an item, or chose both consistent and inconsistent, a score of 1.5 was assigned. This method for dealing with missing or ambiguous answers decreases the variability of the sums making the analysis more conservative. A Mann-Whitney U test performed on the comparison of the sum of participants' ratings.

Participants were given space next to each choice to provide an explanation for their answer. The answers resulted in the creation of a new categorical scheme based on the CLG. Comparisons between the number and type of inconsistencies found by each group were made based on this scheme. Finally, the participant completed a questionnaire, parts of which were used in this research, about his/her background and computer experience (Appendix G).

## 3.4 Experimental Procedure

### 3.4.1 Pre-Experiment Session

Participants were informed that the purpose of the experiment was to evaluate the user interface to a software application. They then were asked to sign an informed consent form. Upon completion of the informed consent form and after performing the monitor and keyboard setup procedure outlined by the instructions in Appendix H, the actual experiment began. The mouse, keyboard, screen, and chair were positioned to accommodate the participant. The mouse and keyboard response rates were not changed. The screen highlight color was a light blue, and the keyboard repeat rate was set at the third fastest setting. Participants were allowed to position the mouse on the left or right side of the keyboard.

### 3.4.2 Experiment Session

Participants were given eight tasks, in a random order, from one of six sets of tasks. The experimenter intervened only when absolutely necessary and requested by the participant, or when the participant incorrectly assumed that they had completed the exercise. Participants were prompted with either, “Use the Help system if you are having problems,” or “Please reread the question and make sure you are doing everything the question has asked for.” If this prompting did not help the participant, then the experimenter would single out the part of the question that the participant did not understand by reading the question aloud and emphasizing the misunderstood passage. For example, if the participant did not “**bold**” a piece of the text that the task requested, the experimenter read the question with emphasis on the word “bold.” Two instances required the experimenter to add additional comments for the participant to complete the task (out of a total of 1584 tasks).

After the first two blocks, a post-test questionnaire was administered followed by a rest break. During the next two sets, some participants forgot how to complete one or two of the previous tasks and were prompted with, “You have not completed the task as specified” followed by “Please reread the question to verify that you have completed everything the question has asked for.” Post-test questionnaires and rest breaks followed these sets.

Once the participant completed the three sets, the final questionnaire was administered. The participant was instructed to use the interface in any manner to help them in completing the questionnaire. The participants were not allowed to use the application when completing the prior questionnaires, but were encouraged to explore this version while filling out this questionnaire. They were made aware that no specific tasks were to be performed and that no data were collected during the final questionnaire period.

### *3.4.3 Data Collection*

The application automatically logged the time and input from the user. The software recorded user inputs in increments of  $1/60$  of a second. Although not all keystrokes were recorded, keystrokes that changed the users mode of operation were recorded. For example, if the user typed in a field, the act of entering the field was counted as a step, but the actual text typed was not recorded. Figure 4 outlines a sample of the actual information recorded during a task.

**Instruction Presented:**

Go to pages 5 and 6 of the To Do function and add two more pages. Put “Movies to Rent” in 18 pt bold type at the top of the left visible page. Put “Die Hard, Mask, Batman, Star Wars, Animal House” under it in 18 pt. plain type.

**Steps Taken to Complete Instruction:** (time in ticks)

Step #	Time Started	How and Where Step was Completed
0	20069	••Done! with EndVersion Instruction 008 Card 49••
1	20680	w/Icon bkgnd button “To Do” of bkgnd “Weekly” of window “Date”
2	20820	w/Icon bkgnd button “Next” of bkgnd “To Do” of window “Date”
3	20936	w/Icon bkgnd button “Next” of bkgnd “To Do” of window “Date”
4	21456	w/Icon item “New Pages” menu “Edit”
5	21615	w/icon bkgnd field “todoL” of bkgnd “To Do” of window “Date”
6	21765	w/Menu Size item “18” menu “Size”
7	21905	w/Menu Style item “Bold” menu “Style”
8	22206	w/Key Style item “Plain” menu “Style”
9	23118	w/Icon bkgnd button “Return_Overview” of bkgnd “To Do” of wd “Date”
10	23154	Done! withEventH3 Instruction 09 Card 44

**Explanation of Steps Taken to Complete Instruction - Participant:**

- 1) clicked the “To Do” button in the Main window,
- 2-3) clicked the next button twice to go from page 1/2 to 5/6,
- 4) clicked the New Pages icon (if he had used the menu it would have said “w/menu”),
- 5) clicked in the field and typed the text into the field (steps requiring typing text were observed and not recorded by the program),
- 6) selected all the text and selected size “18” from the menu using the mouse,
- 7) selected the Bold menu choice using the mouse,
- 8) selected the necessary text and used the keyboard equivalent for “Plain” text (note the w/Key prefix),
- 9-10) returned to the main window and clicked the DONE! button.

Figure 4. A sample of steps required to complete Task H of Block 6.

**Statistics Recorded From this Task:**

Time to complete task (23154-20069) = 51 seconds

Number of steps to complete task =10

Number of steps using Icons/Quick Keys/ Menu Choices= 6/1/2

[does not include clicking the DONE! button]

**Statistics Known About this Task:**

Minimum number of steps needed to complete task=8

Minimum amount of time needed to complete task= 27 seconds

**Quickest and Minimal Step Method for Completing this Task;**

[Task starts with the completion of the last task]

- 1) Go to the To Do function by clicking on the icon or by using the function key,
- 2) "Find" page 6 {with the Icon or Quick Key, this moves the user to page 6},
- 3) New page {with icon or menu or quick key},
- 4) Enter field {by clicking mouse in area of the field},
- 5) Type all text- select the text- and select 18 point from menu  
{this is considered one step since once one enters a field the next action must be outside the current focus of the field},
- 6) Select the text to be bold and choose the Bold menu choice  
{the rest of the text should already be plain, but some participants verify this by selecting the text to be plain and choosing plain from the menu, this is considered an extra step},
- 7) Finally the participant clicks the return to main screen button,
- 8) Then clicks the DONE! button. [This completes the task and starts the next task.]

Figure 4 (continued). A sample of steps required to complete Task H of Block 6.

---

# CHAPTER

# 4

---

# RESULTS

## 4.1 Time and Step Data Analysis

A regression analysis was performed to determine the relationship between time and step data. The results indicate that 73.2 percent of the variance in time data is attributed to the step effect. Figure 5 shows a scatter plot of the time/step rates recorded for each participant (33 subjects  $\times$  6 blocks = 198 observations). The step measure comprises the linear sum of the number of keyboard clicks, clicks on icons with the mouse, and menu selections with the mouse. The multiple regression analysis indicates that icon usage accounts for 64.8 percent ( $p < 0.0001$ ) of the time measure, while keyboard and menu usage account for 8.0 ( $p < 0.0001$ ) and 0.8 ( $p = 0.0170$ ) percent of the variance, respectively.

Due to the high correlation between time and steps, and the small additional variance accounted for by the keyboard and menu usage, only the time data was analyzed. These preliminary results suggest that not enough emphasis was placed on the keyboard and menu usage during the experiment.

## 4.2 Time to Complete Tasks Analysis

The results of an ANOVA for task completion time are presented in Table 5. The main effects of Set ( $p < 0.0001$ ) and Block ( $p < 0.0001$ ) were significant, as were Set  $\times$  Group ( $p = 0.0431$ ), Block  $\times$  Set ( $p < 0.0001$ ) and Block  $\times$  Group ( $p = 0.0081$ ) interaction effects.

The degrees of freedom for all significant within-subjects effects were adjusted according to the Greenhouse and Geisser (1959) correction for violations of sphericity. The degrees of freedom for the Set and Set x Block effects were adjusted with the Greenhouse Geisser epsilon values.

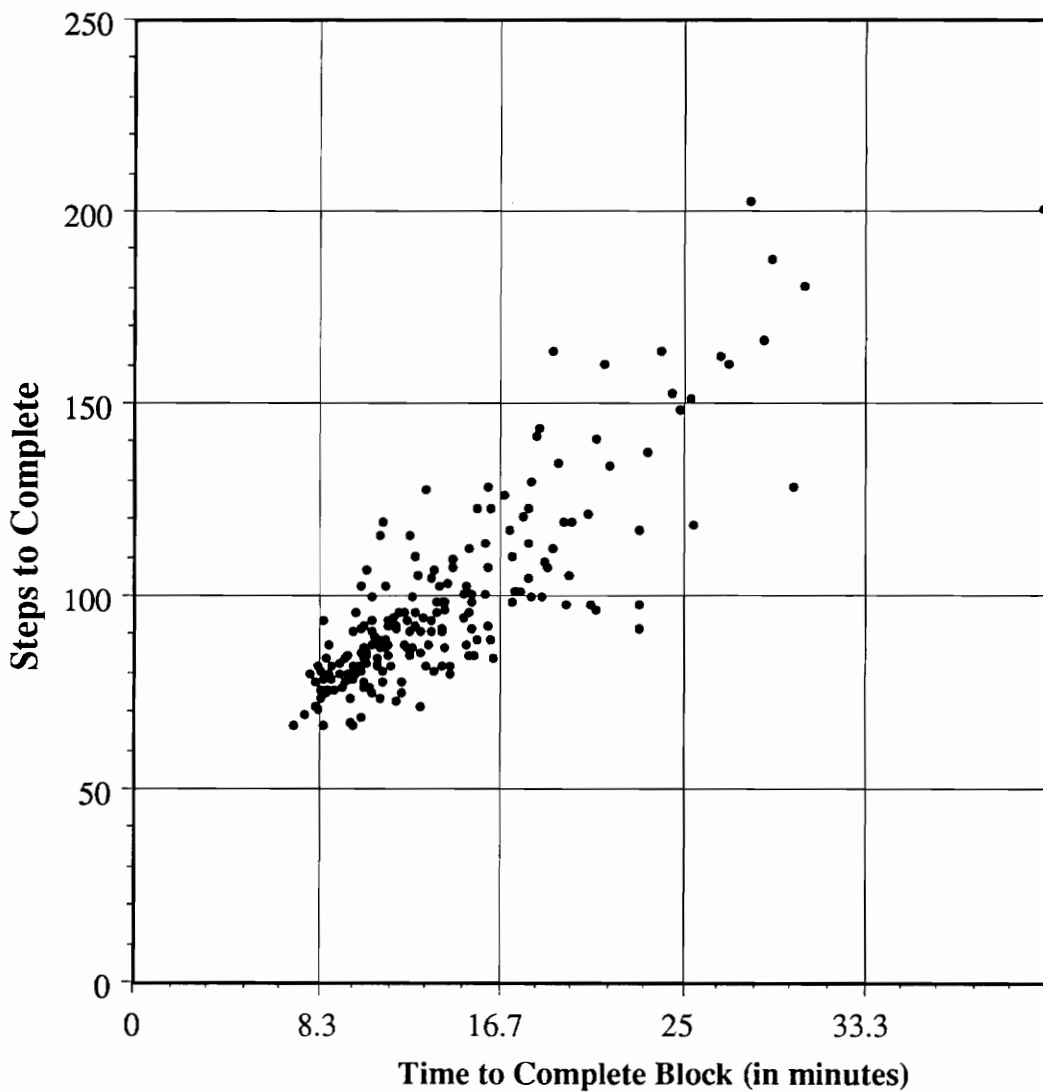


Figure 5. Relationship between time and step data (n=198).

TABLE 5. Analysis of Variance Summary Table for Performance (Time)

Source of Variance	df	MS	F	p
<u>Between Subjects</u>				
Group	2	231732.76	1.93	0.1627
Subjects/Group	30	120067.00		
<u>Within Subjects</u>				
Set	2	4145642.80	151.12	< 0.0001*†
Set*Group	4	72129.20	2.63	0.0431*
Set*Subjects/Group	60	27432.40		
Block	1	1589548.80	75.95	< 0.0001*
Block*Group	2	118679.50	5.67	0.0081*
Block*Subjects/Group	30	20929.70		
Block*Set	2	756412.30	34.25	< 0.0001*†
Block*Set*Group	4	43478.00	1.97	0.1108
Block*Set*Subjects/Group	60	22086.00		
Total degrees of freedom =		197		

\* Significant at  $p < 0.05$

† Greenhouse and Geisser (1959)  $p$ -value: Set ( $\hat{\epsilon} = 0.766$ ), and Set  $\times$  Block ( $\hat{\epsilon} = 0.7581$ ). Block (df=1) does not require a correction.

The main effect of Set on completion time is presented in Figure 6. The associated Newman-Keuls comparison across Sets is presented in Table 6. The comparison of mean times reveal that participants' performance improved across Sets.

The main effect of Block produce significantly longer completion times in Block 1 than Block 2 ( $p < 0.0001$ ). Figure 7 reveals that performance improved from Block 1 to 2. This trend reveals an overall learning effect.

Figure 8 presents the Block  $\times$  Set interaction, while Table 7 shows differences between blocks are significant only at Set 1 ( $p < 0.01$ ). This trend reveals that performance in the test and post-test sessions does not improve during these sets.

The Group  $\times$  Set interaction is presented in Figure 9. To find significant differences among the levels of Group within Sets, simple-effect F-tests were performed (Table 8). The results indicate that differences are significant ( $p < 0.005$ ) between groups in Set 1 and 2, but performance is unaffected by Group in Set 3. This indicates that it is possible to transfer from inconsistent (test) to consistent versions (post-test) of the interface without significantly affecting completion times.

Table 9 presents the results of Newman-Keuls test on the simple effect of Group within Set 1. The results show that Group 2 is significantly slower than the other two groups in Set 1. Differences are due to the variance in Block 1 and not Block 2. Thus, differences in the Newman-Keuls test of Group 2 can be attributed primarily to the results from Block 1.

Table 10 presents the Newman-Keuls test on the simple effect of Group 1 within Set 2. Group 1 is significantly faster than the other two groups for Set 2. Since group differences in block completion time were not significant in the previous block, the significant difference in Set 2 ( $p < 0.05$ ) establishes that performance by Group 2 and 3 was hindered by the inconsistent version presented in Set 2.

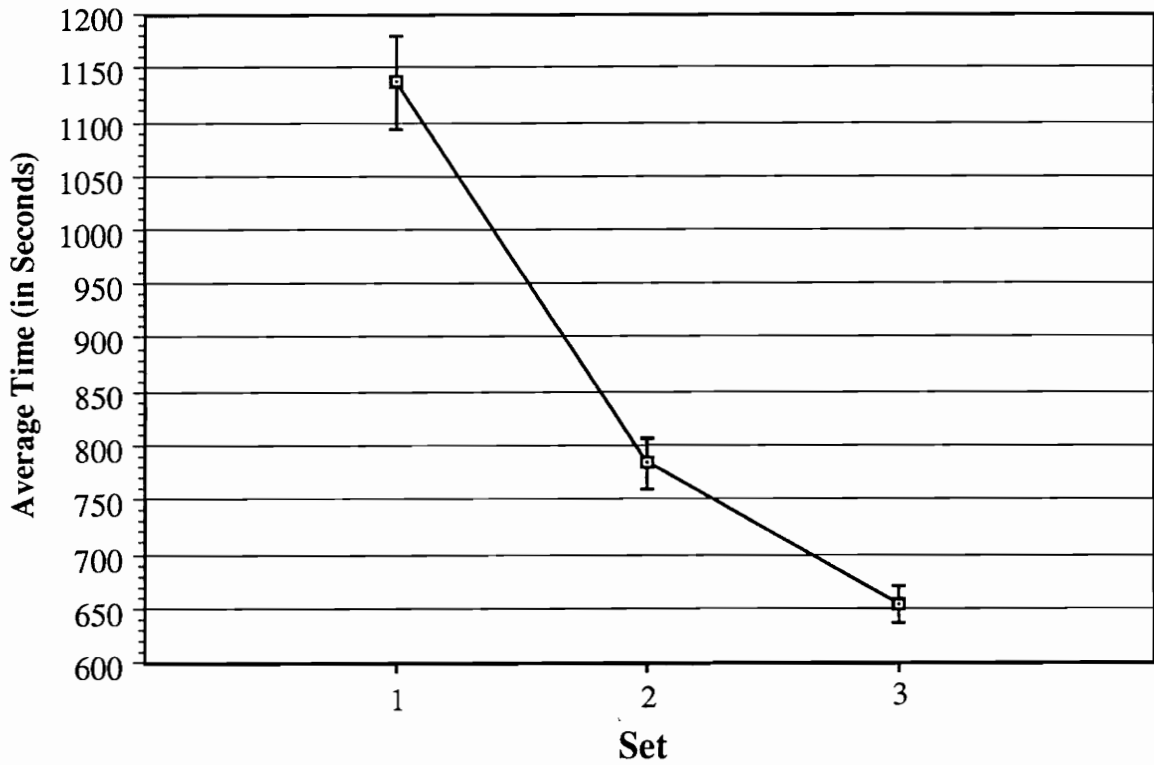


Figure 6. Main effect of Set, averaged over Subjects, Groups, and Blocks on time-to-complete tasks. Error bars show  $\pm 1$  standard error of the mean ( $n=66$ ).

TABLE 6. Results of the Newman-Keuls Test on the Main Effect of Set

<u>SET</u>	<u>Mean Time</u> (in seconds)	
1	1138	A
2	784	B
3	654	C

Means with the same letter are not significantly different at  $p < 0.05$ .

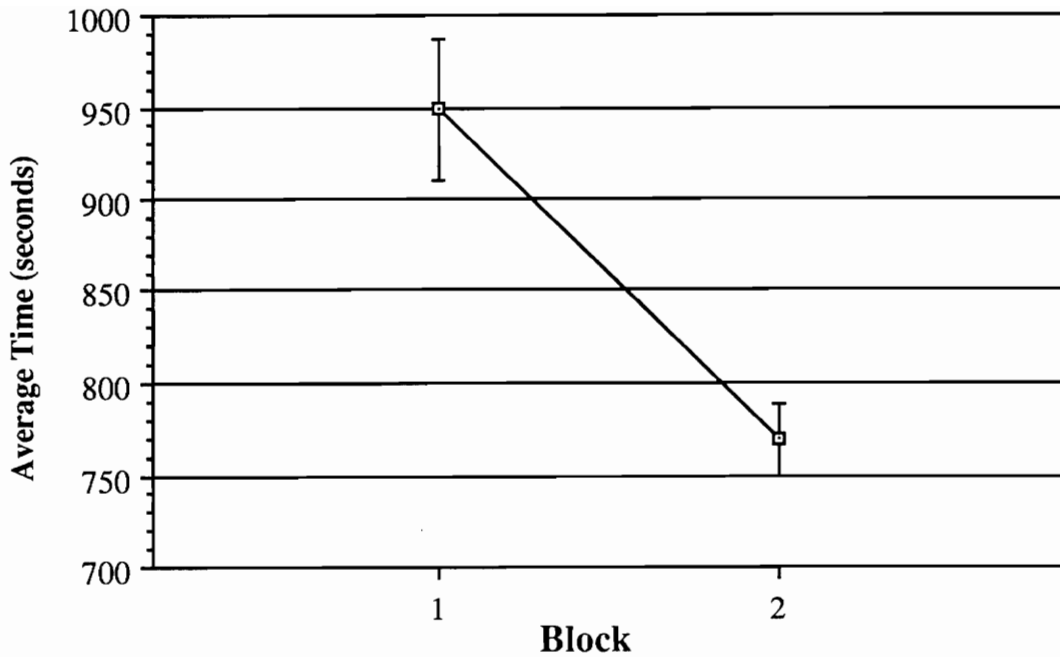


Figure 7. Main effect of Block, averaged over Subjects, Groups, and Sets on time-to-complete tasks. Error bars show  $\pm 1$  standard error of the mean (n=66).

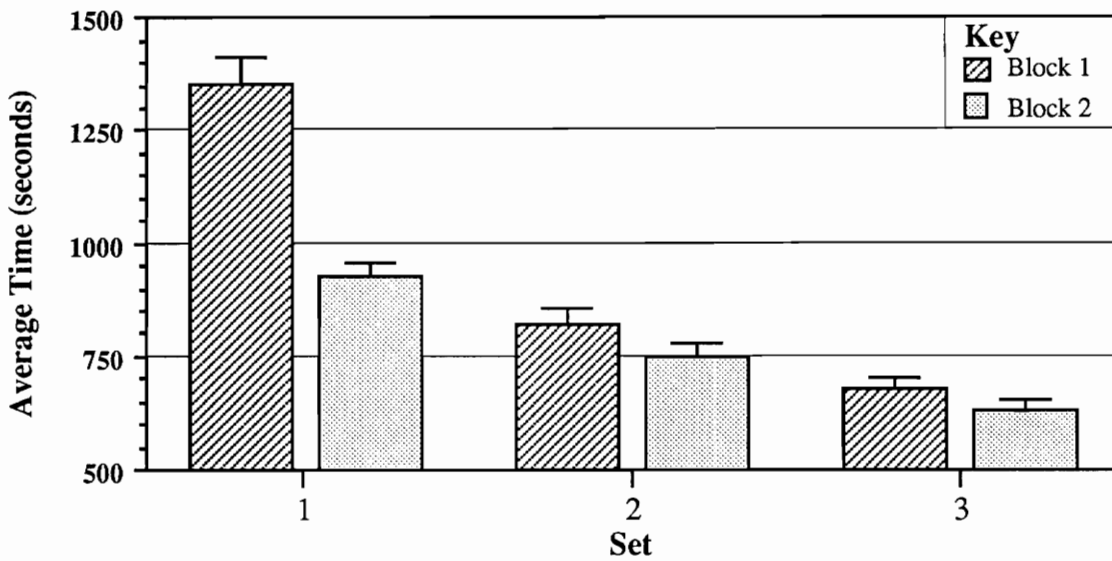


Figure 8. Block x Sets interaction, averaged over Subjects and Groups, on time-to-complete tasks. Error bars show +1 standard error of the mean (n=33). Plotted as a function of Set.

TABLE 7. Results of the Simple-Effect F-tests on Block at the Different Levels of Set

Source	df	MS	<i>F</i>	<i>p</i>
Block at Set 1	1	2994602.50	135.58	< 0.001
Block at Set 2	1	76686.35	3.47	0.078
Block at Set 3	1	31084.54	1.41	0.230
Block*Set*Subjects/Group	60	22086.00		

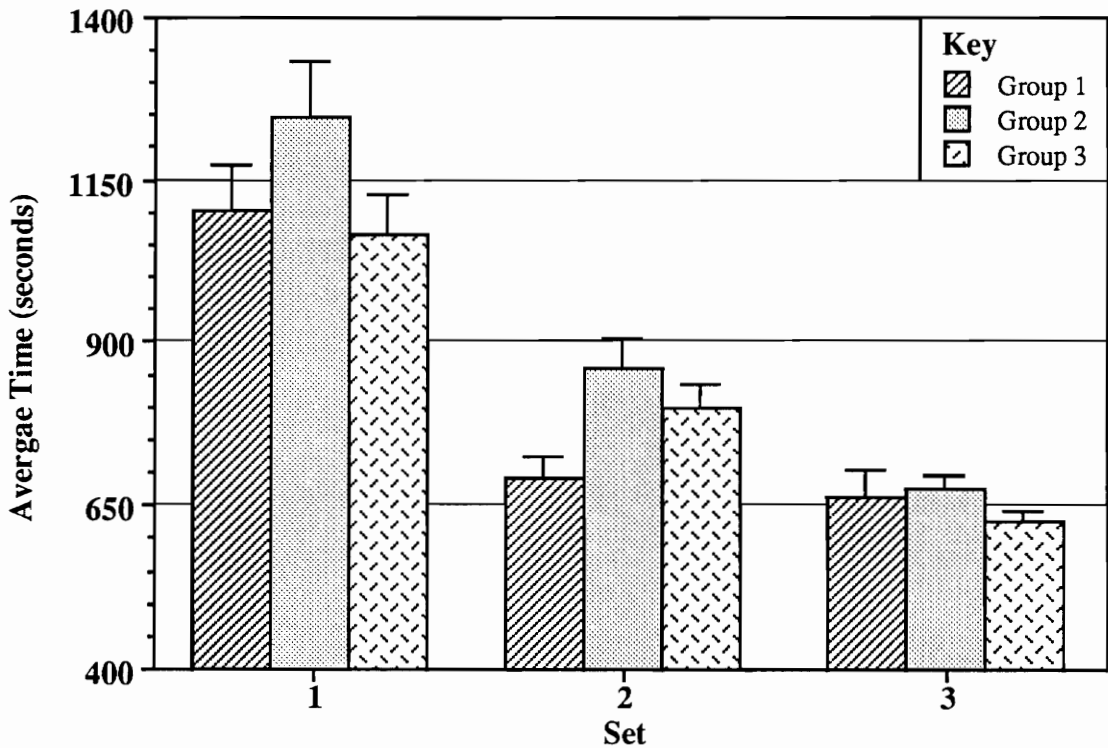


Figure 9. Group  $\times$  Set interaction, averaged over Blocks and Subjects, on time-to-complete tasks. Error bars show +1 standard error of the mean ( $n=22$ ). Plotted as a function of Set.

TABLE 8. Results of the Simple-Effect F-tests on Group at the Different Levels of Set

Source	df	MS	F	p
Group at Set 1	2	197020.57	7.18	< 0.002
Group at Set 2	2	163962.74	5.98	< 0.002
Group at Set 3	2	15007.94	.55	0.581
Set*Subjects/Group	60	27432.40		

TABLE 9. Results of the Newman-Keuls Test on the Effect of Group Within Set 1

Group	Mean Time (seconds)	
3	1065	A
1	1104	A
2	1245	B

Means with the same letter are not significantly different at  $p < 0.05$ .

TABLE 10. Results of the Newman-Keuls Test on the Effect of Group Within Set 2

Group	Mean Time (seconds)	
1	691	A
3	799	B
2	861	B

Means with the same letter are not significantly different at  $p < 0.05$ .

The Group  $\times$  Set interaction (graphed by Group) is presented in Figure 10, while Table 11 presents the simple-effects F-tests. Results indicate that differences are significant among Sets in each Group ( $p < 0.001$ ). Viewing this interaction by Group reveals how performance decreased for each Group between Sets.

Table 12 presents the Newman-Keuls test for the Group  $\times$  Set interaction (Group 1). Group 1 was significantly slower in Set 1 than in the other two Sets ( $p < 0.05$ ). Thus, performance did not improve between Sets 2 and 3, indicating some constant level of performance to measure against the performance achieved by Groups 2 and 3. Performance in Set 2 is similar to the performance achieved in Set 3 for the other two conditions.

Table 13 presents the Newman-Keuls test for the Group  $\times$  Set interaction (Group 2). Group 2 was significantly faster in each successive Set ( $p < 0.05$ ). This indicates that subjects were still learning how to manipulate the interface during the second set. Subjects still were learning in Set 3 as compared to Group 1 which did not improve between Sets.

Table 14 presents the Newman-Keuls test for the Group  $\times$  Set interaction (Group 3). The results are similar to that found in the Group 2 Newman-Keuls results. Performance of Group 3 showed significant improvements across Sets ( $p < 0.05$ ). Group 3 was still improving their performance while Group 1 was not improving between Sets.

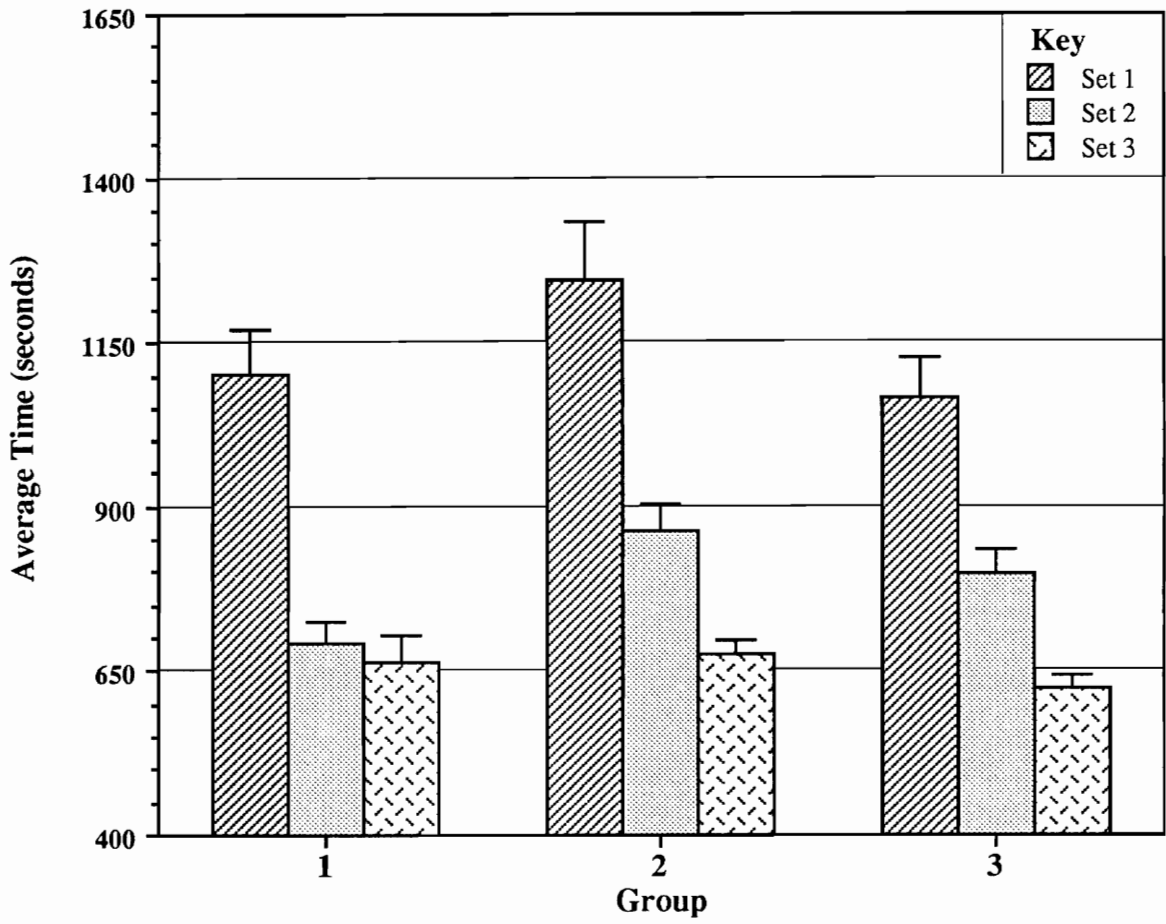


Figure 10. Group  $\times$  Set interaction, averaged over Blocks and Subjects on time-to-complete tasks. Error bars show +1 standard error of the mean (n=22). Plotted as a function of Group.

TABLE 11. Results of the Simple-Effect F-tests on Set at the Different Levels of Group

Source	df	MS	<i>F</i>	<i>p</i>
Set at Group 1	2	1338608.90	48.80	< 0.001
Set at Group 2	2	1866131.10	68.02	< 0.001
Set at Group 3	2	1085161.30	39.56	< 0.001
Set*Subjects/Group	60	27432.40		

TABLE 12. Results of the Newman-Keuls Test on the Effect of Set Within Group 1

<u>Set</u>	<u>Mean Time</u> (seconds)	
3	664	A
2	691	A
1	1104	B

Means with the same letter are not significantly different at  $p < 0.05$ .

TABLE 13. Results of the Newman-Keuls Test for the Effect of Set Within Group 2

<u>Set</u>	<u>Mean Time</u> (seconds)	
3	674	A
2	861	B
1	1245	C

Means with the same letter are not significantly different at  $p < 0.05$ .

TABLE 14. Results of the Newman-Keuls Test for the Effect of Set Within Group 3

<u>Set</u>	<u>Mean Time</u> (seconds)	
3	624	A
2	799	B
1	1065	C

Means with the same letter are not significantly different at  $p < 0.05$ .

The Block  $\times$  Group interaction (graphed by Block) is presented in Figure 11. The analysis revealed significant differences as shown by the results of the simple-effect F-tests in Table 15. These results indicate learning occurred during the Sets.

The Block  $\times$  Group interaction (graphed by Group) is presented in Figure 12. Group was significant at Block 1 in the simple-effect F-tests as shown by the results presented in Table 16 ( $p < 0.001$ ). This indicates that differences in performance can be attributed to differences in Block 1.

Table 17 presents the Newman-Keuls test for Groups in Block 1. The differences in the Newman-Keuls analysis reveal that Group 2 was significantly slower than the other two groups ( $p < 0.05$ ). Comparisons of Block 1 and Block 2 measures reflect a learning effect across that segment of the experiment.

The Block  $\times$  Set interaction (graphed by Set) is presented in Figure 13. The results of the simple-effect F-tests, presented in Table 18, indicate that Set has a significant effect at both levels of Block ( $p < 0.001$ ). The figure depicts a progressive learning curve in both Blocks.

The Newman-Keuls comparisons of mean completion times show significant performance gains in Block 1 (Table 19,  $p < 0.05$ ) and in Block 2 (Table 20,  $p < 0.05$ ) for each successive Set.

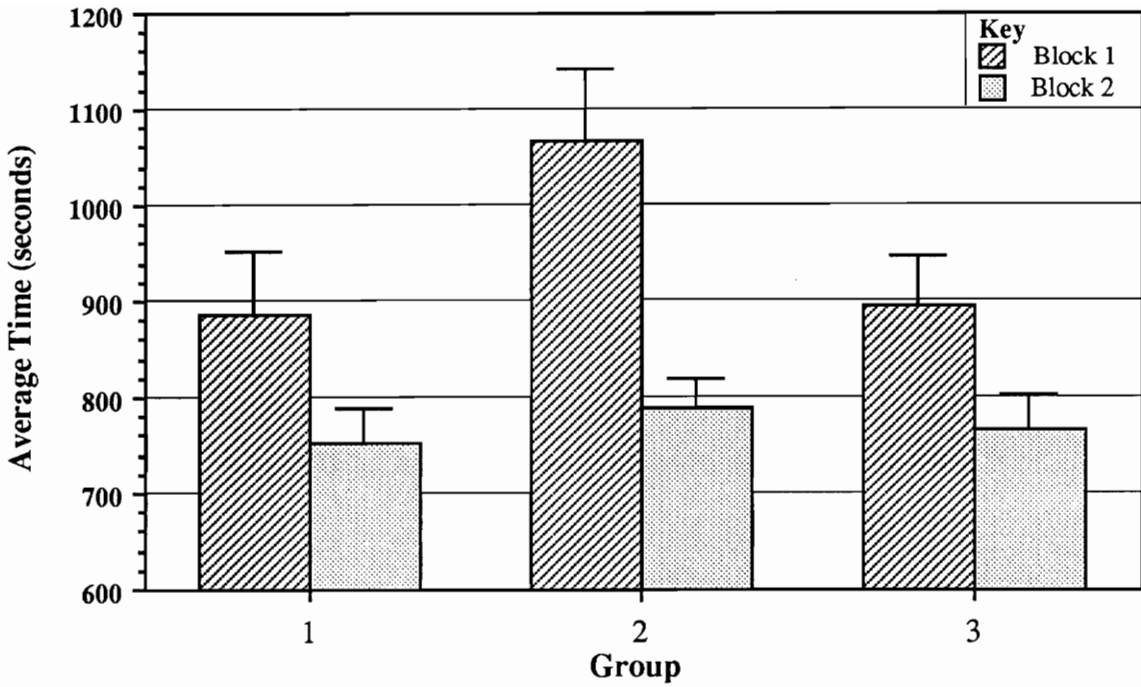


Figure 11. Block  $\times$  Group interaction, averaged over Set and Subjects on time-to-complete tasks. Error bars show +1 standard error of the mean ( $n=33$ ). Plotted as a function of Group.

TABLE 15. Results of the Simple-Effect F-tests on Block at the Different Levels of Group

Source	df	MS	<i>F</i>	<i>p</i>
Block at Group 1	1	295660.17	14.12	< 0.001
Block at Group 2	1	1266402.40	60.50	< 0.001
Block at Group 3	1	264845.26	12.65	< 0.005
Block*Subjects/Group	30	20929.70		

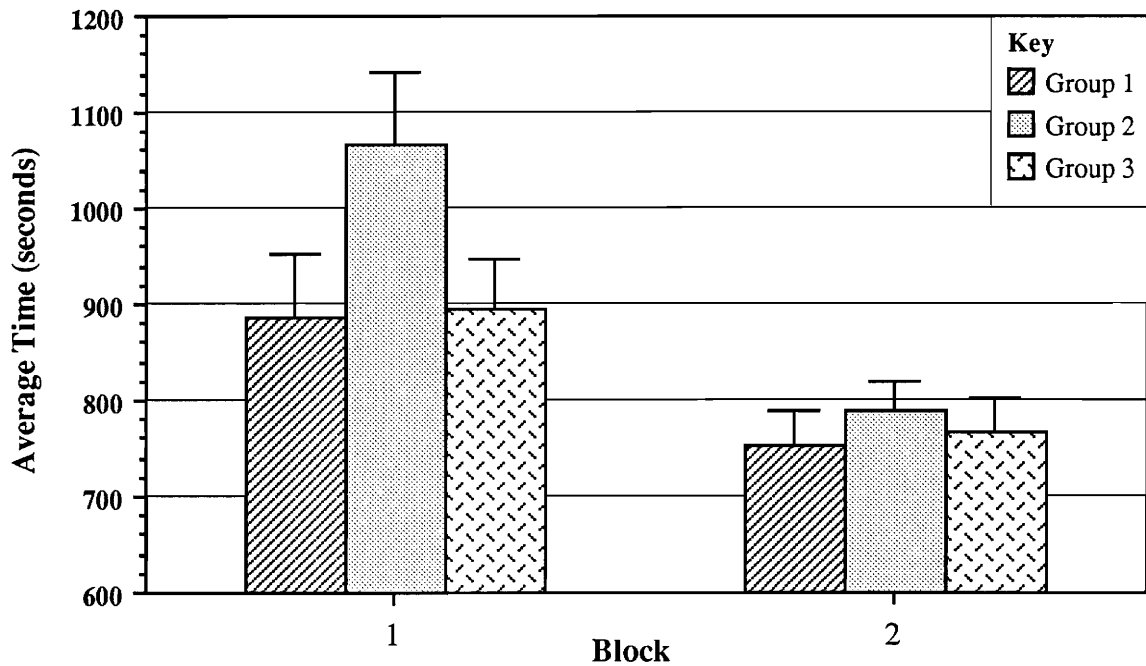


Figure 12. Group  $\times$  Block interaction, averaged over Set and Subjects on time-to-complete tasks. Error bars show +1 standard error of the mean ( $n=33$ ). Plotted as a function of Block.

TABLE 16. Results of the Simple-Effects F-tests on Group at the Different Levels of Block

Source	df	MS	<i>F</i>	<i>p</i>
Group at Block 1	2	339742.31	16.23	< 0.001
Group at Block 2	2	10669.96	0.51	0.605
Block*Subjects/Group	30	20929.70		

TABLE 17. Results of the Newman-Keuls Test on the Effect of Group Within Block 1

<u>Group</u>	<u>Mean Time</u> (seconds)	
1	886	A
3	893	A
2	1065	B

Means with the same letter are not significantly different at  $p < 0.05$ .

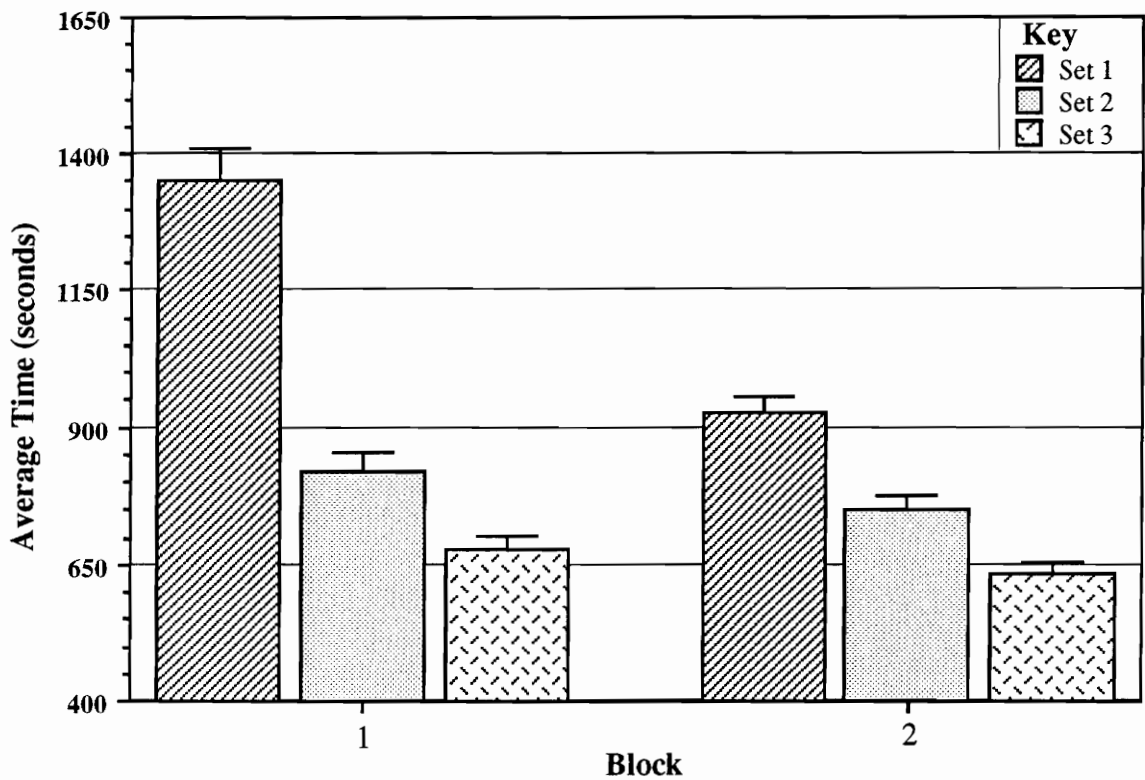


Figure 13. Block x Set interaction, averaged over Group and Subjects, on time-to-complete tasks. Error bars show +1 standard error of the mean (n=33). Plotted as a function of Block.

TABLE 18. Results of the Simple-Effect F-tests on Set at the Different Levels of Block

Source	df	MS	<i>F</i>	<i>p</i>
Set at Block 1	2	4185205.70	189.54	< 0.001
Set at Block 2	2	716849.40	32.46	< 0.001
Block*Set*Subjects/Group	60	22086.00		

TABLE 19. Results of the Newman-Keuls Test on the Effect of Set Within Block 1

<u>Set</u>	<u>Mean Time</u> (seconds)	
3	676	A
2	818	B
1	1351	C

Means with the same letter are not significantly different at  $p < 0.05$ .

TABLE 20. Results of the Newman-Keuls Test on the Effect of Set Within Block 2

<u>Set</u>	<u>Mean Time</u> (seconds)	
3	632	A
2	749	B
1	925	C

Means with the same letter are not significantly different at  $p < 0.05$ .

### 4.3 Subjective Questionnaire (Preference Index)

Table 21 lists the 29 bipolar adjective pairs used in creating the Preference Index (PI) along with the Spearman correlation coefficients for the Consistent-Inconsistent item and the other 28 items. The  $p$ -values for significance of the correlations also are presented. The complete correlation matrix for the 29 items is presented in Appendix I.

The summary of the analysis of variance on the PI data is presented in Table 22. The main effect of Set was significant ( $p < 0.0001$ ), as was the Set  $\times$  Group interaction ( $p < 0.0425$ ). The Greenhouse and Geisser (1959) correction for sphericity for the within-subject effect of Set did not significantly affect the  $p$ -value (0.0001 increased to 0.0007).

Figure 14 presents the main effect of Set, while Table 23 shows the results of the Newman-Keuls comparison between Sets. The results show that Groups rated the versions used in Set 2 as significantly lower than the control version used in Sets 1 and 3 ( $p < 0.05$ ).

Figure 15 presents the Set  $\times$  Group interaction, while the associated simple-effects  $F$ -tests were performed for the Group parameter (Table 24). Groups rated the interface significantly different in Set 2 ( $p < 0.0001$ ), but not in the Set 1 or 3. These results indicate that Groups rated the control version of the interface similarly before and after being administered the version in Set 2. Thus, subjects within Groups were consistent in their ability to use the PI.

It is useful to look at both the differences between Groups for each Set and the differences between Sets for each Group. The former reveals the uniformity of responses to the Set conditions stimuli, while the later reveals the uniformity of Group mean scores in the before and after treatment conditions.

The Newman-Keuls comparison of mean PI for Group at Set 2 is shown in Table 25. Results indicate the versions used by the Groups in Set 2 were rated significantly different ( $p < 0.05$ ). The control version was rated as better than the two versions used by Group 2 and 3. The Newman-Keuls comparison also reveals that the versions used by Group 3 was judged significantly worse than the version used by Group 2 on the PI scale.

TABLE 21. Spearman Correlation Coefficient ( $r_s$ ) for Bipolar Adjective Pairs with the Consistent-Inconsistent Item in the PI

Question	Bipolar Items		$r_s$	$p$
1)	Pleasing	Irritating	0.445	0.0118 *
2)	Friendly	Unfriendly	0.273	0.1230
3)	Complete	Incomplete	0.141	0.4262
4)	Cooperative	Uncooperative	0.463	0.0088 *
5)	Dependable	Undependable	0.618	0.0005 *
6)	Simple	Complicated	0.577	0.0011 *
7)	<i>Consistent</i>	<i>Inconsistent</i>	---	---
8)	Natural	Unnatural	0.324	0.0666
9)	Intelligent	Unintelligent	0.298	0.0914
10)	Interpretable	Uninterpretable	0.388	0.0283 *
11)	Fast	Slow	0.101	0.5670
12)	Adaptive	Unadaptive	0.001	0.9950
13)	Useful	Useless	0.301	0.0885
14)	Redundant	Concise	0.226	0.2003
15)	Uncluttered	Cluttered	0.166	0.3479
16)	Safe	Unsafe	0.009	0.9616
17)	Maintainable	Unmaintainable	0.353	0.0459 *
18)	Easy to learn	Difficult to learn	0.474	0.0074 *
19)	Not frustrated	Frustrated	0.463	0.0088 *
20)	Productive	Unproductive	0.482	0.0064 *
21)	Worth buying	Not worth buying	0.344	0.0517
22)	Relaxed	Tense	0.340	0.0547
23)	Optimistic	Pessimistic	0.525	0.0030 *
24)	Competent	Incompetent	0.429	0.0153 *
25)	Pleased	Disgusted	0.458	0.0096 *
26)	Engaged	Bored	0.296	0.0945
27)	Happy	Unhappy	0.054	0.7617
28)	Intelligent	Stupid	0.203	0.2517
29)	Pleasant	Annoyed	0.463	0.0088 *

\* Significant at  $p < 0.05$  and  $r_s > 0.35$

TABLE 22. Analysis of Variance Summary Table for Preference Index (PI)

Source	df	MS	<i>F</i>	<i>p</i>
<u>Between Subjects</u>				
Group	2	448.86	1.28	0.2929
Subjects/Group	30	350.81		
<u>Within Subjects</u>				
Set	2	525.53	10.50	< 0.0001*†
Set*Group	4	132.09	2.64	0.0425*
Set*Subjects/Group	60	50.07		
<hr/>				
Total degrees of freedom =	98			
<hr/>				
* Significant at $p < 0.05$				
† Greenhouse and Geisser (1959) $p$ -value: Set ( $\hat{\epsilon} = 0.7248$ ).				
<hr/>				

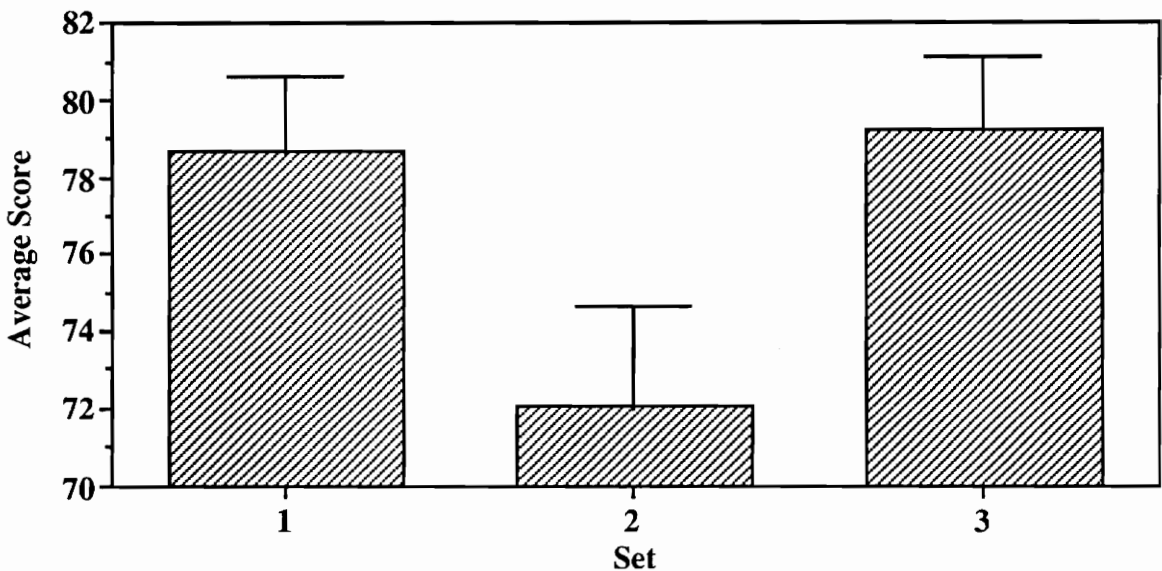


Figure 14. Main effect Set, averaged over Group and Subjects on PI. Error bars show +1 standard error of the mean (n=33).

TABLE 23. Results of the Newman-Keuls Test on the Main Effect Set

Set	Mean PI	
3	79.27	A
1	78.70	A
2	72.09	B

Means with the same letter are not significantly different at  $p < 0.05$ .

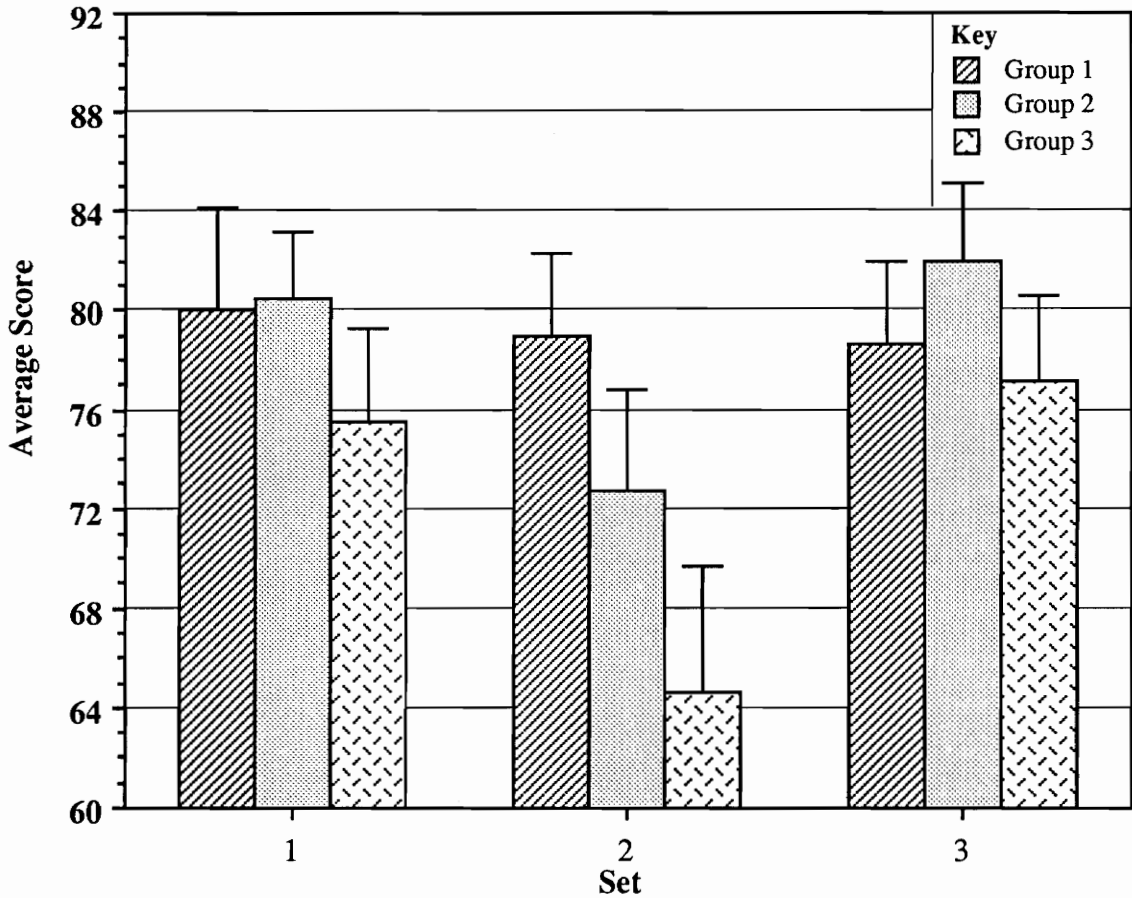


Figure 15. Group  $\times$  Set interaction, averaged over Block and Subjects on PI. Error bars show +1 standard error of the mean ( $n=11$ ). Plotted as a function of Set.

TABLE 24. Results of the Simple-Effect F-tests on Group at the Different Levels of Set

Source	df	MS	<i>F</i>	<i>p</i>
Group at Set 1	2	82.30	1.649	0.201
Group at Set 2	2	563.55	11.256	< 0.001
Group at Set 3	2	67.18	1.342	0.269
Set*Subjects/Group	60	50.07		

TABLE 25. Results of the Newman-Keuls Test on the Effect of Group Within Set 2

<u>Group</u>	<u>Mean PI</u>	
1	78.91	A
2	72.73	B
3	64.64	C

Means with the same letter are not significantly different at  $p < 0.05$ .

The Set x Group interaction (graphed by Set) is presented in Figure 16. Simple-effect F-tests were performed across Sets, at each level of Group (Table 26). Subjects in Group 1 were very consistent in their rating of the three identical versions of the interface and showed no significant difference in their ratings. Results show there are significant differences in Sets for Group 2 and 3.

Table 27 presents the results of the Newman-Keuls test on the effect of Set at the different levels of Group reveals that subjects in Group 2 rated the inconsistent version of the interface used in Set 2 as significantly worse than the control version used in Sets 1 and 3 ( $p < 0.05$ ).

Table 28 presents the results of the Newman-Keuls test on the effect of Set at the different levels of Group reveals that subjects in Group 3 rated the inconsistent version of the interface used in Set 2 as significantly worse than the control version used in Sets 1 and 3 ( $p < 0.05$ ).

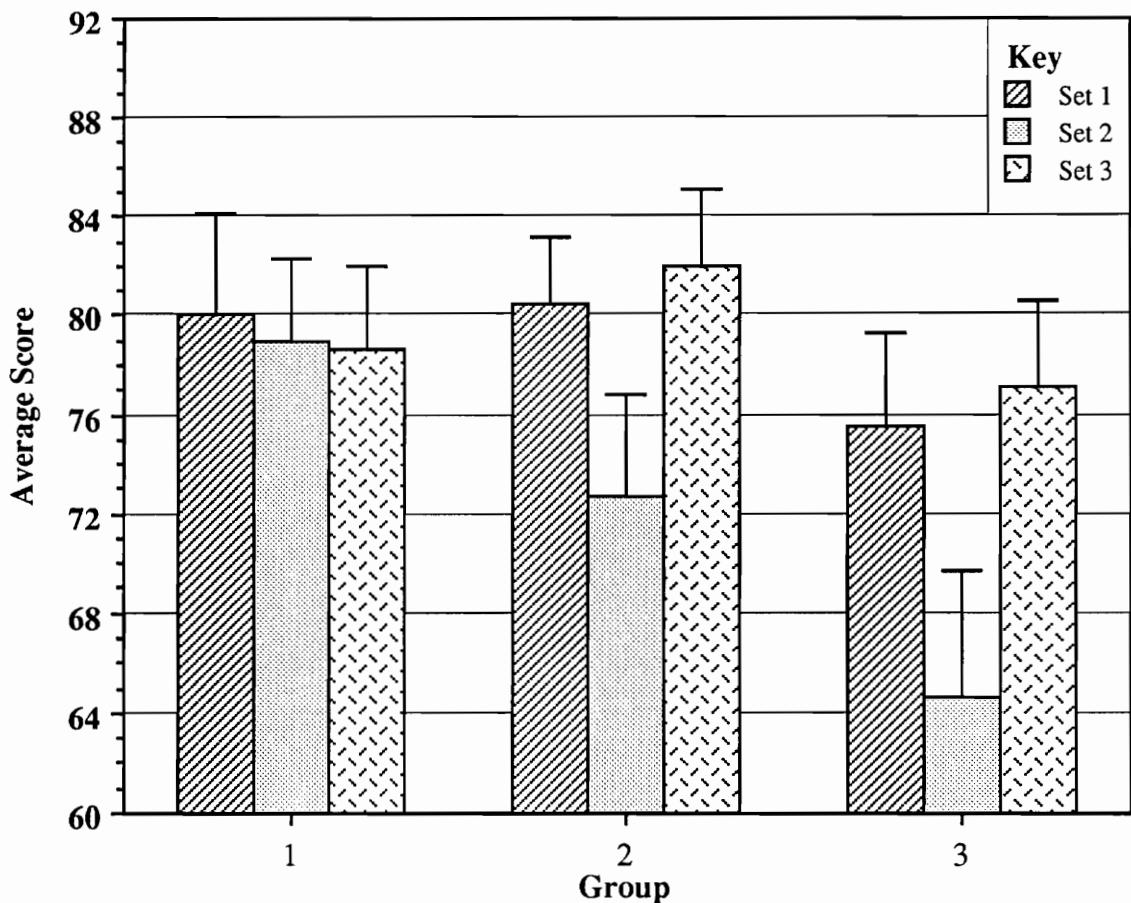


Figure 16. Group x Set interaction, averaged over Block and Subjects on PI. Error bars show +1 standard error of the mean (n=11). Plotted as a function of Group.

TABLE 26. Results of the Simple-Effect F-tests on Set at the Different Levels of Group

Source	df	MS	<i>F</i>	<i>p</i>
Set at Group 1	2	6.58	0.131	0.877
Set at Group 2	2	271.49	5.423	< 0.007
Set at Group 3	2	511.64	10.219	< 0.001
<hr/>				
Set*Subjects/Version	60	50.07		

TABLE 27. Results of the Newman-Keuls Test on the Effect of Set Within Group 2

<u>Set</u>	<u>Mean PI</u>	
3	82.00	A
1	80.46	A
2	72.73	B

Means with the same letter are not significantly different at  $p < 0.05$ .

TABLE 28. Results of the Newman-Keuls Test on the Effect of Set Within Group 3

<u>Set</u>	<u>Mean PI</u>	
3	77.18	A
1	75.55	A
2	64.64	B

Means with the same letter are not significantly different at  $p < 0.05$ .

---

# CHAPTER

# 5

---

# DISCUSSION

A comparison of the ANOVA results for the time and PI measures revealed an important finding; objective and subjective measures do not have the same degree of precision in assessing human performance characteristics. The time measure showed that both inconsistent versions impacted the time-to-complete tasks, while the results of the PI revealed significant differences between all three groups. The version used by Group 1 (the control) was the most preferred, followed by the version used by Group 2, and then the version used by Group 3. Thus, the PI was more sensitive in detecting inconsistencies among the three interfaces. Accordingly, both subjective and objective measures should be taken into account when assessing the consistency of a user interface design.

## 5.1 Anomalies in the Objective Measure

In order to draw conclusions about the differences in completion times found in Set 2, it is necessary to evaluate the cause of the difference between Groups in Set 1. Figure 17 presents the three-way Set  $\times$  Group  $\times$  Block interaction. The differences in mean completion times in Set 1 were due to the large differences between Groups for Block 1. Although the three-way interaction was not significant it is useful to view the plot to probe the cause of the significant Set  $\times$  Group interactions in Set 1 and 2 ( $p < 0.1108$ ).

Mean times for Groups differed only by 32 seconds, as compared to 328 seconds in Block 1. One might consider the propensity of subjects in Group 2 to do worse in future blocks based on their performance in Block 1 of Set 1. In fact, a review of the completion times for Block 1 revealed that one subject's performance in Group 2 was 4 standard deviations above the Block 1 mean times.

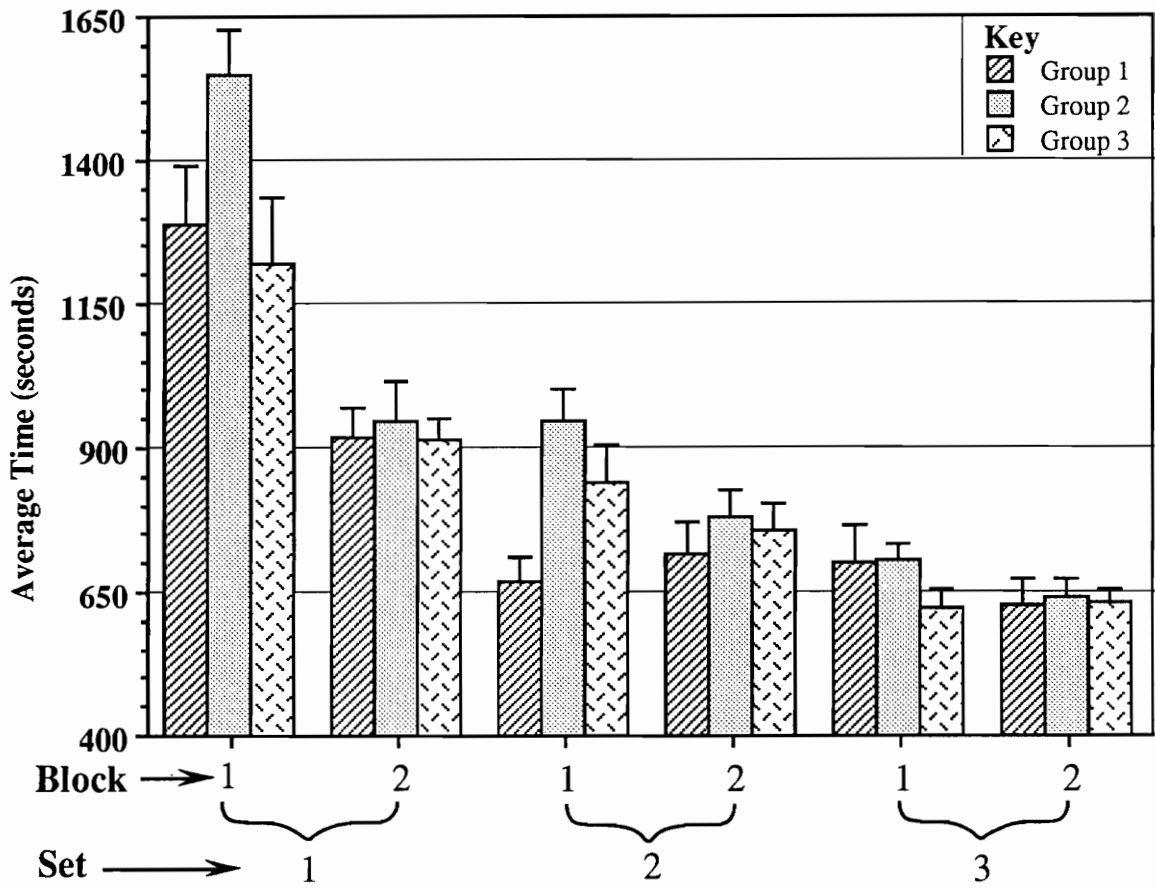


Figure 17. Set x Blocks x Group interaction, averaged over subjects. Error bars show +1 standard error of the mean (n=11).

The average time to complete Block 2 of Set 1 was approximately 30 seconds. Thus, performance in the control condition, immediately prior to the treatment condition was approximately equal across Groups. The performance times for Groups in the final block, Block 2 of Set 3, were about 13 seconds apart. These results give a basis to interpret the differences in Set 2 as significant real differences and not due to chance or pre-experimental group differences.

Another method for showing that Groups were similar enough to draw comparisons between differences in the treatment condition can be found in Figure 18. The figure presents scatter plots for the times to complete Block 1 for all subjects in Set 1 and 2. It is presented to show that the differences between Groups in Set 1 and 2 were not due to the same subjects in each Group. By calculating a Pearson correlation coefficient between Block 1 completion times for Set 1 and 2, it can be shown that subjects in Group 2 did not have a predisposition to perform worse than the subjects in the other Groups. A Pearson correlation of the task completion times for Set 1 and 2 was low for Group 2 ( $r^2 = 0.0081$ ,  $p = 0.8131$ ). The results indicate the comparisons of Set 2 data can be treated as due to differences between treatment conditions and not to a group bias, as indicated in the performance by Group 2 in Block 1 of Set 1.

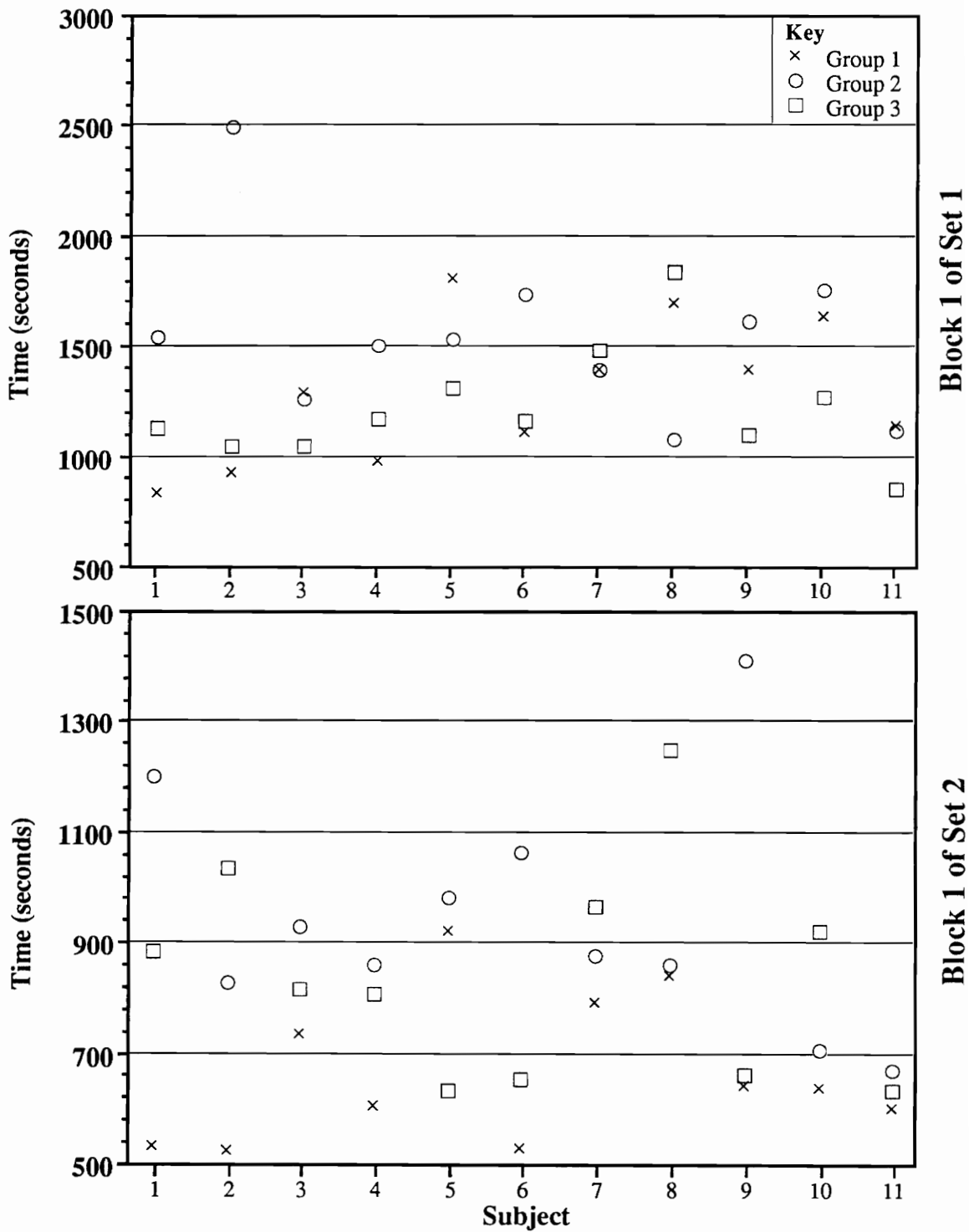


Figure 18. Time to complete Block 1 of Set 1 and 2 for each subjects (n=1). Note different scales for each graph.

## 5.2 Objective Performance Measures

An ANOVA of task completion times revealed that performance for subjects in Groups 2 and 3 was not as fast as for subjects in Group 1 when measured in the first block of the treatment condition. Subjects performed as well as the control group in the second block of tasks in Set 2. This finding was similar to that reported by Kellogg (1987) who found task completion times were not significantly different after only one set of tasks.

The version administered to Group 3 was the most visually distinct, as shown in the screen images of the three interfaces used in the experiment (Appendix J). The summary of inconsistencies presented in Table C-3 of Appendix C lists the known and purposely seeded inconsistencies for the three versions of the application. The physical layout of the inconsistent version used by Group 3 accounted for 18 of the 62 seeded inconsistencies in the interface. Briefly, this included rearrangement of icons in each function, placement of icons on a palette, use of color, and menus *in* the window. The version administered to Group 3 also used rules for the navigation features, the menus, pop-up icons, and other features that were different from both the consistent version and the inconsistent version used by subjects in Group 2. Thus, because the versions used by the subjects in Group 3 *looked* different one might anticipate performance would be worse than the other two versions. As shown in the results of the analysis of variance for the Set  $\times$  Group interaction (Figure 9), the performance of Group 3 was not significantly different than the performance of Group 2. Therefore, the poor performance was probably due to other inconsistencies or the interaction of other inconsistencies with the physical inconsistencies.

The results of the analysis of variance of time shows that subjects in all three groups learned how to use the application. Average performance times decreased by a factor of 2 from 20–25 minutes in Set 1 to 11–12 minutes in Set 3. Performance among groups was not statistically significant in Set 2. The non-significant differences in completion times for Set 3 indicates that no post-treatment effects occurred from using the inconsistent version in Set 2. Thus, experienced subjects could adapt to changes in the consistency of the interface.

### 5.3 Subjective Performance Measures (Preference Index)

The PI provided a measure of user opinion. Items in the PI were found to be present in the interface in a similar manner to the Consistent-Inconsistent item, as shown by the correlations presented in Table 21. With a set of items, like the ones in the PI, it is possible to document the changes in preference during the evolution of the interface. In essence one would try to extract what it means for an interface to be consistent. Users might be more comfortable using words like *pleasing*, *cooperative*, *dependable*, *simple*, *interpretable*, *maintainable*, *easy to learn*, *not frustrating*, and *productive* to help describe the interface. The PI was based on these bipolar adjectives and others such as; *optimistic*, *competent*, and *pleasing* to represent an overall measure of preference.

The correlated adjective pairs reveal what types of items relate to the consistency of a graphical user interface. Thus, by testing with a variety of items one can assess if deficiencies exist in the design. It might be difficult to determine specific deficiencies in the design, but it would be useful towards obtaining a general measure for guiding designers in creating a more consistent interface. The composite measure of PI was used as a benchmark to rate the effect of the treatment version on subjective assessments.

The Preference Index used only 14 of the 29 items from the questionnaires. The correlations between items can also be used to look at other sets of items. Since the C-I item was the focus of the PI, a factor analysis was not used as the results of a factor analysis would have been too general to be applicable to the analysis of consistency. The complete correlation matrix is presented in Appendix I for readers who might wish to review some of the other correlations.

An analysis of variance on PI scores showed that subjects in Group 2 and 3 rated the interface lower than did subjects in Group 1 who used only the consistent version (Figure 15). Subjects in Group 3 rated the treatment version in Set 2 lower than the version presented to Group 2. The PI allows for the two inconsistent versions of the interface to be distinguished. The results of the task completion time analysis reveal the performance of Group 2 was slower in the treatment condition than the Groups with the other two

versions, but the PI showed the version used by Group 3 to be the least preferred. This indicates that the subjective and objective measures were *not* identical and thus subject's opinions were not based on subject's ability to work quickly with the interface.

Subjects in Group 1 were very consistent in their ratings across Sets (Figure 16). The subjects used the same interface three times and the PI scores indicate no difference. If subjects became bored or uninterested in the application then one might have expected a drop in ratings. The consistent ratings indicate that the PI scores for subjects in Group 1 were not affected by the length of the experiment or by having to repeat tasks in the identical versions in each Set. Since subjects in Group 1 had to use the same version three times, one would expect boredom or fatigue to occur in Group 1 before it would occur in Groups 2 and 3, where subjects used different versions of the application in the Sets.

The PI scores for all three Groups were not significantly different when comparing Sets 1 and 3 as presented in Figure 16 ( $p < 0.05$ ). If end users had evaluated an interface at an early stage of development and found it inconsistent, they might overcompensate their estimate when rating the post-test version of the interface. Subjects finding the new version superior to the old version would rate it higher than users who had not used the interface. New users would not be able to draw direct comparisons between the versions since they would not know how poor the interface was previously. This is similar to the problems with allowing end users to preview a user interface during the development cycle. Exposure to the interface could bias a user to make judgements that lead to false expectations (Miller, 1989). These data *do not* support any indications of a bias due to moving from an inconsistent to a consistent interface. Since the post-test version was identical to the version used in the pre-test condition, the results show that the inconsistent versions did not bias participants into rating the consistent version in Set 3 differently than when rating the consistent version in Set 1.

Han, Jorna, Miller, and Tan (1990) evaluated four input devices for a graphic direct manipulation interface and concluded that first impressions of the usability of the input devices were not significantly different from subjective assessments made by experienced users of the devices. Measures of time did reveal differences in both expert and novice

performance, therefore both subjective and objective measures were required to assess the input devices. The PI is only a general subjective measure, while further ratings and free response questions can be used to identify specific inconsistencies in the interface.

A combination of subjective and objective measures helps to alleviate misinterpretation of a single measure. In this experiment, the use of matched blocks of tasks allowed participants to experience a variety of different situations. If PI scores for the Groups were higher than pre-test levels then one might find an inverse relationship between the decreasing block completion times and the increasing PI scores. Since Subjects did *not* rate the interface as better in Set 3 (Figure 15), even with significantly faster completion times (Figure 6), improvements in task completion times did not affect the subject's ability to give consistent PI ratings in Set 3.

The objective and subjective measures in this study gave two different indications of consistency. Furthermore, decomposition of the two measures can be used to identify specific problems in the usability of the application. The time measure can be used to identify specific tasks that cause poor performance. Once these tasks are identified, an analysis of the steps used to complete the task can help determine potential problems. Rewording of the tasks, redesigning the operations needed to complete the task, changing the look of an icon or the name of a menu item might be required to improve performance on the task. Once the redesign of the interface has been completed, a new set of subjects and subjects who participated previously can be used to retest the interface to measure improvements in time and subjective measures. With both measures it is possible to gain a more complete assessment of the effect of inconsistencies on a graphical user interface design.

---

# CHAPTER

# 6

---

# CONCLUSIONS

The analysis of objective and subjective data do not reveal the same differences between versions of the interface. Thus, subjective and objective measures were needed to get a more complete representation of a user's ability to use an (inconsistent) interface. The results support the following;

- Subjective and Objective measures differ in their ability to discriminate between interfaces with varying degrees of inconsistencies.
- Subjects are able to make consistent judgements of an interface without being biased by being exposed to an inconsistent version of the same application.

Results of the analysis of variance on time show that performance of Groups 2 and 3 was significantly slower than the control group, while the subjective measures show that the version used by Group 3 was the least preferred.

There are no significant differences between Groups during the third Set. Thus, the groups who were administered inconsistent versions during Set 2 were unaffected by having used an inconsistent version prior to using the consistent version in Set 3.

If subjective and objective measures responded in a same manner to inconsistencies there would be little need to use both measures. A more detailed analysis, such as one using free response questions, would be needed to identify differences between the inconsistent

versions of the interface. The results support that the effect of inconsistencies needs to be measured by both time or other performance measures, e.g., step or error data, and subjective measures to gain a better understanding of the effect of inconsistencies. The PI was able to differentiate between two inconsistent versions of an interface, when timed performance measures found them equal. In addition, the time required to give each questionnaire (about 3 minutes) can be viewed a good use of time. The total experiment took approximately two to two and one-half hours to complete per subject. The time measure takes at least two-thirds of the total time compared to 9 minutes for the 3 questionnaires.<sup>1</sup>

The results indicate that subjects are able to make comparisons between versions of the interface, as noted by their consistent ratings of the versions of the interface in the pre-test and post-test conditions, without being biased by using the treatment version of the interface. Thus, the same subject pool can be used to make comparisons between future versions of the interface given the PI ratings for the original version of the interface.

Since objective performance for Groups 2 and 3 improved to a point equal to the performance of Group 1, one must consider if the seeded inconsistencies were “inconsistent” enough to test. Since subjective and objective measures were affected by the treatment condition, one must conclude that the two treatment versions were “inconsistent.” Polson (1989) pointed out that a researcher needs to be clear to seed inconsistencies that would appear in a real environment and not simply a set of annoying inconsistencies that would only appear inside the laboratory. There are certain inconsistencies that would not be realistic or appropriate to create in an experiment. For example, switching the placement of the brake and accelerator pedals in an automobile simulator would not make any sense. At this time the decision, is left to the researcher to make the original determination of what is consistent or inconsistent in a specific situation, but user-testing should be used to determine if the design is appropriate.

---

<sup>1</sup> The remainder of the session was devoted to rest breaks and other questionnaires administered for additional research.

Others researchers have compared command and graphic interfaces and performance and the transfer of training between interface types (Tombaugh, Paynter, and Dillon, 1989; Antin, 1988; Whiteside, Jones, Levy, and Wixon, 1985). The results were contradictory and further research is needed to confirm how variations *within* the graphical interface will effect transfer of training. The rapid decrease in performance time for the two groups that used the inconsistent versions of the interface in Set 2 (Figure 17) upheld the present author's belief that the underlying consistency of a graphic direct manipulation interface with a set of standardized guidelines accounts for a large part of a user's good performance.

By making standards and guidelines readily available and by not supporting (i.e., not purchasing their software) developers who do not follow the specifics and the spirit of the guidelines developers will be forced to create consistent software. Although developers are being "forced" to make an interface consistent, it should allow the developers to streamline the development process and limit the need for future revisions. Developers can then spend more time creating innovative software that is not covered by the guidelines, thus extending the functionality and usability of software.

Since the subjective and objective measures can potentially yield different results, designers need more accurate methods for defining, interpreting and testing consistency in graphical user interface design. By using subjective and not objective measures of performance when designing a user interface leads to pitfalls in the performance and acceptance of the design.

---

# CHAPTER

# 7

---

# FUTURE RESEARCH

A significant problem involves the creation of a taxonomy for assessing the impact of inconsistencies at different levels of an interface (i.e, using Robert's levels, 1988). The creation of a taxonomy to classify inconsistencies "consistently" requires a rule base that can separate the issues presented to designers of graphical user interfaces. A usable taxonomy would require a structure that would allow designers naive to the specific interface being tested to extract problems reliably. A classification scheme requires a more detailed level of knowledge about the relationship between inconsistencies and the interpretation of their impact to get reliable information.

A command language (Moran, 1981) or user action notation (Siochi, and Hartson, 1989) requires a detailed understanding of low-level building blocks. This might hamper their acceptance by developers working under time constraints, or designers unfamiliar with the methodologies. Again the best solution, at this time, would be to create guidelines with instructions on their application that developers can apply without needing complicated flow diagrams and difficult to interpret command notations. Increasing the number of existing guidelines for a specific type of interface style is one solution. Companies or groups that propose standards must be willing to create living documents that can be enhanced and extended quickly. For example, research into how to extend the consistency of function and keyboard equivalents would be a good effort. Some keys are already defined; while one could investigate creating a second tier of command sets that apply to specific type of applications (word processing, graphics, database, etc.). This would create

consistency within a specific types of applications, where it might have been less appropriate to define those functions at the system level where other keys are defined for consistency within all applications.

The long term effects of inconsistencies were not explored in this research. If subjects were exposed to the inconsistent versions of the interface for an additional few days then the effect of transfer might be more severe. Although the subjective opinions were pronounced, timed performance measures showed quick recovery from the presentation of the inconsistencies.

It would be interesting to see how inconsistencies in virtual devices<sup>2</sup> and other specific classes of inconsistencies affect performance. Some research has addressed specific types of inconsistencies and transfer of training issues (Kellogg, 1988; Kieras and Bovair, 1986; Polson, Muncher, and Engelbeck, 1986), but more research into the effect of specific classes of inconsistencies on learning is needed. Specifying the appropriateness of the device and how it is operated needs to be addressed. Extensive literature is available on the use and functionality of real input devices such as mice, trackballs, and touchscreens (Han et al., 1990; Card, English and Burr, 1978; Epps, Snyder and Muto, 1986), but these virtual devices need to be evaluated to determine what types of information are better suited to a specific device. In general, the rules and understanding of when a menu is appropriate can be found in guidelines like Smith and Mosier (1984) and in computer-dependent guidelines (Apple, 1986). These references do not discuss if other virtual devices would provide a more usable interaction style for presenting the information. Overall, it would be useful to try to determine what classes of inconsistencies cause problems in user interface design.

The current trend to allow end users to adapt an application interface to their specific needs may have important implications. An individual might customize a program, thus enhancing their subjective opinion, but also causing their objective performance to decline. If this is the case, then there will be a tradeoff between productivity and user satisfaction.

---

<sup>2</sup> In this experiment, virtual device refers to system objects acted upon by a user, such as a scrolling window, scrolling list, pull-down menu, hierarchical menu, check box, or dialog box. Users created changes in the interface with the "virtual devices." See Table C-2 in Appendix C for a complete analysis of the classes of inconsistencies seeded in the three versions of *The Personal Organizer*.

Additionally, novice users, such as temporary workers, office mates, or supervisors might need to work with the personalized version of an application, and then be temporarily slowed by the novel interface. If the users had to go from an outdated inconsistent version of an application to a current redesigned consistent version of the same application, or vice versa, then this movement between versions would be comparable to the experimental design employed in this research. The control version would be similar to Version 2.0 of the application versus 1.0 for the inconsistent version. The functionality and the user interface for software is constantly evolving. Completely redesigned versions of an application are not uncommon. If one of the inconsistent versions of the Personal Organizer had been released and then updated to the consistent version, users would have experienced a short period of adjustment, but would have overcome the problems to achieve a level of performance equal to users who were already using the consistent version. Thus, the level of consistency within an application might be due to the flexibility designers offer to allow for individual users to customize the software to fit their specific needs. The more a user customizes the interface the larger the potential impact on usability.

Researchers need to establish if there are specific classes of inconsistencies, and interactions between classes that cause poor performance. By expanding on the CLG (Moran, 1981), one could create a classification scheme that addresses the specific problems of a graphical user interface. By taking comments made by users during an evaluation period and parsing them into groups that communicate the type and location of the inconsistency, designers would be able to locate problems and eliminate inconsistencies. One also could use video recordings of the experimental sessions, as well as notes from watching subjects during sessions to create new guidelines for interface design. The experimenters can record their observations while conducting the experiment and create working theories on interface design. Review of the video tapes could be used to analyze how often and how many of the working hypotheses were correct. Individual hypotheses could then be tested to determine their feasibility in other conditions and applications. This could result in the clarification of existing guidelines, expansion of the scope of current guidelines to include new areas, or completely new guidelines.

---

## REFERENCES

- Antin, J. F. (1988). An empirical comparison of menu selection, command entry and combined modes of computer control. *Behavior and Information Technology*, 7 (2), 173-182.
- Apple Computer, Inc. (1987). *Human Interface Guidelines: The Apple Desktop Interface*. New York: Addison-Wesley.
- Apple Computer, Inc. (1989). *HyperCard Stack Design Guidelines*. New York: Addison-Wesley.
- Bachman, R. (1989). A methodology for comparing the software interfaces of competitive products. *Proceedings of the Human Factors Society 33rd Annual Meeting* (pp. 1214-1217). Santa Monica, CA: Human Factors Society.
- Barnard, P. J., Hammond, N.V., Morton, J., Long, J.B., and Clark, I.A. (1981). Consistency and compatibility in human-computer dialogue. *International Journal of Man-Machine Studies*, 15, 87-134.
- Berry, R. E. (1988). Common user access – A consistent and usable human-computer interface for the SAA environments. *IBM Systems Journal*, 27(3), 281-300. Armonk, NY: IBM.
- Blake, T. (1986). *Introduction to the Art and Science of User Interface Design*. Intuitive Software and Interactive Systems, CA.
- Bouchard, J. C., and Robert, J. M. (1989, September). *Towards a consistency manager for user interfaces: Requirements and constraints*. Poster session presented at HCI International '89, Montreal, Quebec.

- Bowers, V. A. (1990). *Concurrent versus retrospective verbal protocol for comparing window usability*. Unpublished doctoral dissertation, Virginia Polytechnic Institute & State University, Blacksburg, VA.
- Briggs, P. (1989). Consistent benefits for the system designer and the end-user. *Applied Ergonomics*, 20(3), 160-167.
- Brown, C. M. (1988). *Human-Computer Interface Design Guidelines*. Norwood, NJ: Ablex.
- Card, S. K., English, W. K., and Burr, B. J. (1978). Evaluation of mouse, rate controlled isometric joystick, step keys, and text keys for text selection on a CRT. *Ergonomics*, 21, 601-613.
- Card, S. K., Moran, T. P., and Newell, A. (1983). *The psychology of human-computer interaction*. Hillsdale, NJ: Lawrence Erlbaum Associates.
- Carroll, J. M., Mack, R. L., and Kellogg, W. A. (1988). Interface metaphors and user interface design. In M. Helander (Ed.), *Handbook of Human-Computer Interaction*. Amsterdam: Elsevier Science Publishers, B.V.
- Casali, J. G., Lam, S. T., and Epps, B. W. (1987). Perceptions of hearing protector comfort and usability: an overview and application of a rating/ranking methodology. *Sound and Vibration*, 21 (12), 10-18.
- Chin, J. P., Diehl, V. A., and Norman, K. L. (1988). Development of an instrument measuring user satisfaction of the human-computer interface. In *CHI '88 Conference Proceedings, Human Factors in Computing Systems* (pp. 213-218). New York: Association for Computing Machinery.
- Cohen, M., and Schirmer, E. (1989, September). *Human factors in the success of a commercial software package: Lotus 1-2-3*. Paper presented at HCI International, Boston, MA.

- Coleman, W. D., Williges, R. C., and Wixon, D. R. (1985). Collecting detailed user evaluations of software interfaces. In *Proceedings of the Human Factors Society 29th Annual Meeting* (pp. 240-244). Santa Monica, CA: Human Factors Society.
- Digital Equipment Corporation. (1988, December). *XUI Style Guide*. [Computer manual]. Maynard, MA: Author.
- Epps, B. W., and Casali, J. G. (1985). Hearing protection device comfort and user preference: an investigation and evaluation methodology. In *Proceedings of the Human Factors Society 29th Annual Meeting* (pp. 814-818). Santa Monica, CA: Human Factors Society.
- Epps, B. W., Snyder, H. L., and Muto, W. H. (1986). *Comparison of six cursor devices on a target acquisition task* (Tech. Report). Blacksburg, VA: Human Factors Engineering Center, Virginia Polytechnic Institute and State University.
- Greenhouse, S. W., and Geisser, S. (1959). On methods in the analysis of profile data. *Psychometrika*, 83, 314-320.
- Grudin, J. (1989). The case against user interface consistency. *Communications of the ACM*, 32 (10), 1164-1173.
- Grudin, J., Ehrlich, S. F., and Shriener, R. (1987). *Positioning human factors in the user interface development chain*. In *CHI '87 Conference Proceedings, Human Factors in Computing Systems* (pp. 125-131). New York: Association for Computing Machinery.
- Ham, M. (1987). Software design rules. *Dr. Dobb's Journal*, (7), 112-116.
- Hammer, M., Kunin, J. S., and Schoichet, S. (1983). What makes a good user interface? *Office Automation Conference* (pp. 121-130). Washington: AFIPS Press.

- Han, S. H., Jorna, G. C., Miller, R. H., and Tan, K. C. (1990). A comparison of four input devices for the Macintosh interface. In *Proceedings of the Human Factors Society 34th Annual Meeting* (pp. 267-271). Santa Monica, CA: Human Factors Society.
- IBM (1987, First Edition). *Common User Access Panel Design and User Interaction*. [Computer Manual]. Boca Raton, FL: Author.
- Kellogg, W. A. (1987). Conceptual consistency in the user interface: Effects on user performance. In H.J. Bullinger and B. Shackel (Eds.), *Human Computer Interaction - Interact '87, Proceedings of the IFIP Conference* (pp. 389-394). New York: Elsevier Science Publishers B.V.
- Kieras, D. E., and Bovair, S. (1986). The acquisition of procedures from text: A production-system analysis of transfer of training. *Journal of Memory and Language*, 25, 507-524.
- Koritzinsky, I. H. (1989). New ways to consistent interfaces. In J. Nielsen (Ed.), *Coordinating User Interfaces for Consistency* (pp. 93-106). New York: Academic Press.
- Miller, R. H. (1989). *Analysis of Hardware Rapid Prototyping Software Available for Display and Control Development*. Virginia Polytechnic Institute and State University; Department of Industrial and Systems Engineering.
- Mish, F. C. (1983). *Webster's ninth new collegiate dictionary*. Springfield, MA: Merriam-Webster.
- Moran, T. P. (1981). The command line grammar: A representation for the user interface of interactive computer systems. *International Journal of Man-Machine Studies*, 15, 3-50.

- Park, M. Y. (1989). *Laboratory investigation of in-field influences on spectral noise attenuation and comfort of insert and circumaural hearing protectors*. Unpublished masters thesis, Virginia Polytechnic Institute and State University, Blacksburg, VA.
- Payne, S. J., and Green T. R. G. (1986). Task-action grammars: A model of the mental representation of task languages. *Human Computer Interaction*, 2, 93-133.
- Polson, P. G. (1989, December 11). Personal communication.
- Polson, P. G. (1988). The consequences of consistent and inconsistent interfaces. In R. Guindon (Ed.), *Cognitive science and its applications for human-computer interaction* (pp. 59-108). Hillsdale, NJ: Lawrence Erlbaum Associates.
- Polson, P. G., Muncher, E., and Engelbeck, G. (1986). A test of common elements theory of transfer. In *CHI '86 Conference Proceedings, Human Factors in Computing Systems*, (pp. 78-83). New York: Association for Computing Machinery.
- Reisner, P. (1981). Formal grammar and human factors design of an interactive graphics system. *IEEE Transactions in Software Engineering*, SE-7, March, 229-240.
- Robert, J. M., and Wang, W. (1989, September). *A statistical method to evaluate the relative importance of inconsistencies in user interfaces and measure the benefits of solutions*. Paper presented at the 2nd International Scientific Conference on "Work with Display Units," Montreal, Quebec.
- Robert, J. M. (1988). Detecting inconsistencies in user interfaces. Analysis, design, and evaluation of man-machine systems. *Selected Papers From the Third IFAC/IFIP/IEA/IFORS Conference* (pp. 357-363). Oslo, Finland: Man-Machine Systems.
- Roberts, T. L., and Moran, T. P. (1982). Evaluation of text editors. In *Proceedings Human Factors in Computer Systems Conference* (pp. 136-141). New York: Association for Computing Machinery.
- Shneiderman, B. (1987). *Designing the user interface*. Reading, MA: Addison-Wesley.

- Smith, S. L., and Mosier, J. N. (1984). *Design guidelines for the user interface for computer-based information systems*. Bedford, MA: MITRE Corporation, Electronics Systems Division. (Available from NTIS).
- Siochi, A. C., and Hartson, H. R. (1989). Task-Oriented Representation of Asynchronous User Interfaces, In *CHI '89 Conference Proceedings, Human Factors in Computing Systems* (pp. 183-188). New York: Association for Computing Machinery.
- Tombaugh, J. W., Paynter, B., and Dillon, R. F. (1989). Command and graphic interfaces: User performance and satisfaction. In G. Salvendy and M.J. Smith (Ed.) *Designing and Using Human-Computer Interfaces and Knowledge Based Systems* (pp. 369-375). Amsterdam: Elsevier Science Publishers B.V.
- Tullis, T. S. (1983). The formatting of alphanumeric displays: A review and analysis. *Human Factors*, 25(6), 657-682.
- Whiteside, J., Jones, S., Levy, P. S., Wixon, D. (1985). User performance with command, menu, and iconic interfaces. In *CHI '85 Conference Proceedings, Human Factors in Computing Systems* (pp. 185-191). New York: Association for Computing Machinery.
- Wolf, R. (1989). Consistency as process. In J. Nielsen (Ed.), *Coordinating User Interfaces for Consistency* (pp. 89-92). New York: Academic Press.

---

# APPENDIX A: Macintosh Screening Test for Users

- (1) Have you used ResEdit?
- (1) Have you used StuffIt?
- (2/2) Have you used HyperCard? - Have you written scripts in HyperCard?
- (2/2) What menu choices are usually found in the EDIT menu? What are their quick keys?
- (2) Where is HELP usually found in a program and how is it accessed?
- (2) How do you delete a file that the Finder reports as being locked?
- (2) Have you used a color Macintosh?
- (2) What does the command "Find" usually do in an application? and how is it activated?
- (2) How can you scroll through the contents of a window?
- (3) Name three styles for fonts found in a typical word processor and their quick keys?
- (2) How do you save changes made in HyperCard?
- (2) Name two buttons found in an open dialog box?
- (3) What is a floating palette? How do you show and hide palettes?
- (2) What % transfer would you say that have for going from one program to another?

In general, what would you guess to be the quick keys for these commands

- (4) Undo Cut Copy Paste Goto Help Find Close the window Cancel
- (1-1) If you have a database program or any program that contains multiple fields what would pressing the TAB key do? what about shift-TAB?
- (2-1) If you are in a modal Dialog box how do you copy text typed in a field in the box to the clipboard? What happens if you try to click outside of the box area?

- (2) Assume you are given a program that simulates the reading of a large book with many chapters. Name some methods for navigating like using a menu choice to jump to the beginning of the last chapter.
- (2) If I gave you a piece of software and did not tell you what it did, could you install it on the Macintosh and figure out how to start to use it?
- (2) If I give you a new word processing program, how long would it take for you to figure out the program enough to be qualified to type out a simple resume? (not including the time to typing)
- (1) On a scale of 1 to 10, 1 being a complete novice and 10 being an expert in using Macintosh programs, what would you give yourself as a rating?
- (2) What does a pop-up menu do, and how can you recognize one?

Total Score is out of a possible 50 points. Questions were asked verbally for all subjects, to allow for scoring to be flexible.

---

## APPENDIX B: Informed Consent Form

You have been solicited as a research participant for our evaluation involving the use of a new software application for the Macintosh computer. Prior to the actual session a vision test and Macintosh interface analysis needs to be performed. Because you have certain rights, this informed consent has been drawn-up to delineate those rights and responsibilities of both the research participants and the experimenters. Please read through this document before you decide whether or not to participant. The evaluation is being run by; **Dr. Robert Beaton**, Faculty Advisor (703-231-5936) and **Richard Miller**, Graduate Student (703-951-3496).

First you will be given a vision test. The purpose of the vision test is to determine whether your vision meets the criteria for participating in the evaluation. The vision test given to you is not a professional eye test; therefore, the results should not be considered an accurate description of your vision. A professional eye doctor should be consulted for an accurate examination of your vision.

The eye test consists of two parts. If you pass both tests you will be asked to work with a Macintosh application. Even if you are knowledgeable with the Macintosh you might have some difficulty working with the program. This is normal, as the program is not perfect. If you can finish eight out of the ten tasks you will be asked to participate in this evaluation. The application is called the "Personal Organizer" and includes such items as a datebook, telephone directory, and other office items.

These tests will take a total of fifteen minutes. You will not be paid for the vision and Macintosh test. However, if you do participate in the evaluation, you will be compensated for the time you spend in the evaluation.

There are no known risks associated with this evaluation. The only known discomfort would possibly be fatigue resulting from the length of the evaluation. Rest-breaks occur during the ninety minute evaluation. During the evaluation period you will be asked to use the application and rate the program using some short questionnaires.

You may choose monetary compensation or a set of shareware and freeware programs for you time and effort. Your rights as a participant are as follows:

1. You have the right to withdraw from the procedure at any time for any reason.
2. At the conclusion of your participation, you have the right to see your data and withdraw them if you so desire. If you decide to withdraw your data, please inform the administrator immediately. Otherwise, identification of your data will not be possible because they will have been collapsed with other data in order to ensure anonymity.
3. You are requested to refrain from discussing the evaluation with other people (especially people in your area, from where other evaluators might be drawn). We expect all data to be collected by the end of July, 1990. Following that date feel free to discuss the evaluation with anyone you wish.
4. If you wish to receive a summary of the results of this evaluation, please include you address (where you will be six months from now) with your signature below. If you should desire more detailed information about the evaluation, you may contact the above mentioned faculty or graduate student who will provide you with a full report. They will also be able to answer any question you may have regarding your participation.
5. If you do not wish to contact the researchers, you may contact **Dr. E. R. Stout** who is Chairman of the Institutional Review Board. His telephone number is (703) 231-5281. Dr. Stout will be able to handle any complaints or concerns you have regarding the evaluation or your participation.

Finally, we greatly appreciate your invaluable time and effort for participating in this evaluation. Remember, you cannot fail any part of this session, there are no right or wrong answers. The session is to view problems with the software not with you. Your signature below indicates that you have read this in its entirety, and that you consent to participate in this evaluation. Are there any questions?

---

Name

---

Address

---

Signature

---

## APPENDIX C: Seeded Consistencies and Inconsistencies

Each version of the application had a set of characteristics that were defined by the experimenter as inconsistencies. Most of the inconsistencies defined in the two inconsistent versions were “inconsistent” with the control version of the interface. Each area that was known by the experimenter as to be possibly defined by subjects as an inconsistency was placed into one of twenty-two topics.

This section in conjunction with the screen images in Appendix J can be used to gain an understanding of how the application worked and looked. It is difficult to explain some inconsistencies without looking at the menus, icons, layouts, lexicon, and devices used in the three interfaces.

Every inconsistency in this section was categorized into one of the eight levels of the classification scheme. The letter after the explanation of the inconsistency explains where the item was classified. Items were only classified into one category even if the item could apply to many categories. One should review all items before determining that an item might be classified in the “wrong” category. Some items might appear in slightly different forms in different topics and with different categories. Changes in how inconsistencies are identified by a participant can play an important role in categorization. The classifications were based on Moran’s CLG (Table C-2). Although the classes were not addressed in the current research, future analysis of the data will be completed. The class assigned to each inconsistency is presented in parentheses after the item to assist readers in establishing how the inconsistencies were integrated into the interfaces. A summary of the inconsistencies, grouped by class, is presented in Table C-3 at the end of Appendix C. The inconsistencies are also presented, grouped by Topic, in Table C-4 at the end of Appendix C.

TABLE C. Seeded Inconsistencies

Consistent Version 1 (Control)	Inconsistent Version A	Inconsistent Version B
<p>Type of Inconsistency: Physical Location - P Memory - M Virtual Device - D Semantic - S            Unavailable Task - T Syntactic - Y System Status (Feedback) - F Lexical - L</p> <p>Key to Location of Inconsistency: Address Roledex - AR Phone Setup - PS Calendar - CAL To Do Lists - TDL            Area Code Finder - AC Weekly Planner - WP Overview - OV Phone Dial Window - PDW</p> <p>Note: Any item that is not defined by a consistency type is defined as "consistent" in context of the system. These consistencies were not defined by type, only the ones that would affect the number of seeded "inconsistencies" were defined.</p>		
<p><b>1) Menu Structure</b></p>		
<p>a) Styles appeared in their style (Bold looks Bold).</p> <p>b) The Special Copy menu item, available in the AR, was hierarchical. It was placed after the Select All menu item.</p> <p>c) One can move from one menu to another while the mouse button was down.</p> <p>d) Menu choices follow the Apple guidelines.</p> <p>e) The edit commands were sometimes available on the Edit menu. (Y) The menu and delete commands were available in the menu and on appropriate functions.</p> <p>f) Menu titles were in Title style.</p>	<p>a) Style menu choices were not shown in their "Style" (i.e. Bold was not Bold). (L)</p> <p>b) When in the AR the Special copy menu was added to the end of the menu palette, and the standard menu</p> <p>c) Movement from one menu to another could occur in the menu bar, but not the floating palette when the mouse button was down. (Y)</p> <p>d) Style menu choices were arranged alphabetically. (P)</p> <p>e) The edit commands were shown on the palette as icons. (Y) The new and delete card commands were not available on the palette, but they were on the pull-down menu and on the card as icons. (Y)</p> <p>f) Menu titles were in Uppercase style on the menu bar and Title style on the menu palette (L) (except the Apple menu which was in Uppercase style for the Menu Palette). (L)</p>	<p>a) Style menu choices were not shown in their "Style" (i.e. Bold was not Bold). (L)</p> <p>b) Special Copy was hierarchical and also had its own menu on the AR window.</p> <p>c) One could not move from one menu to another while the mouse button was down. (Y)</p> <p>d) Style menu choices were arranged alphabetically (P), except for the Plain Text choice that was positioned first. (P)</p> <p>e) Same as the Control version. (Y)</p> <p>f) Menu titles were in Uppercase style. (L)</p>

TABLE C (continued). Seeded Inconsistencies

Consistent Version 1 (Control)	Inconsistent Version A	Inconsistent Version B
<b>2) Icon (and Field) Placement</b>		
<p>a) Word spacing and alignment was smooth and well integrated.</p> <p>b) The PS fields were lined up on the right side of the function in a logical progression, identical to the HyperCard Phone Dial stack</p> <p>c) The WP had M-W on the left page and Th-Sun on the right page.</p> <p>d) The CAL pages read top to bottom for the order of the months then going to the next page for the next 3 months (Six months shown per two visible pages).</p> <p>e) The TDL has one field for each visible page.</p> <p>f) The phone numbers were aligned in a row with the location buttons to their right, in the AR.</p> <p>g) Function specific icons were placed in the specified menu bar area.</p> <p>h) The Area Code fields were identical with those in the original HyperCard stack</p>	<p>a) Labels and some fields were not aligned. (P)</p> <p>b) Input fields were distributed around the window in small groups of 1, 2, or 3 items. (P)</p> <p>c) The WP had M, Th, F on the left page and Tu, W, Sat, and Sun on the right page. (P)</p> <p>d) This version had the months going from left to right across pages then down. Jan. and Feb. on the top row, March, April on the next row, Etc...) (P)</p> <p>e) Page numbers, page specific icons and arrow keys were placed along the bottom of the main window. (P)</p> <p>f) The AR had the location buttons on the left of the phone number. (P)</p> <p>g) The menu bar area was not used (Y). Icons were placed differently for each function. (P) The Fax icon was larger, and was next to the large phone icon. (P)</p> <p>h) The Area Code fields were identical with those in the original HyperCard stack</p>	<p>a) Labels and some fields were not aligned. (P)</p> <p>b) Input fields were distributed around the window in small groups of 1, 2, or 3 items. (P)</p> <p>c) Same as version A, but the spiral binding graphic was not used. (F)</p> <p>d) Same as version A, but the spiral binding graphic was not used. (F)</p> <p>e) The page numbers for the TDL were moved off the page, and the spiral binding graphic was not used. (P)</p> <p>f) Same as version A, but the "Roledex" graphic was not used. (F)</p> <p>g) Function specific icons were placed in the card/page area. (P)</p> <p>h) The fields in the Area Code function were varied to place the least important information into the space previously occupied by the most important information. (S) The field titles were aligned on the windows left side. (P)</p>

TABLE C (continued). Seeded Inconsistencies

Consistent Version 1 (Control)	Inconsistent Version A	Inconsistent Version B
	<b>3) Edit Keys</b>	
<p>a) Editing could be performed with the X/C/V (Cut, Copy, and Paste) keyboard equivalents.</p> <p>b) Undo was not supported. (T)</p> <p>c) They were available (when appropriate) in the Standard Apple Menu.</p> <p>d) The function keys F2-F4 were supported (Cut,Copy, and Paste, respectively).</p>	<p>a) Editing could be performed with the X/C/V (Cut, Copy, and Paste) keyboard equivalents.</p> <p>b) Undo was not supported. (T)</p> <p>c) The edit menu was placed at a different location for each function. (P) The CAL and OV functions did not have an edit menu. (Y)</p> <p>d) The F1 to F3 keys (Copy,Paste, and Cut, respectively) were supported. (P)</p>	<p>a) Editing could be performed with the X/C/V (Cut, Copy, and Paste) keyboard equivalents.</p> <p>b) Undo was not supported. (T)</p> <p>c) The edit menu was placed at a different location for each function. (P) The CAL and OV functions did not have an edit menu. (Y)</p> <p>d) The F1 to F3 keys (Copy,Paste, and Cut, respectively) were supported. (P)</p>
	<b>4) Menu Item Order</b>	
<p>a) Menus were arranged in a standard format (Apple, File, Edit, Font, Size, Style)</p> <p>b) The six main menus were always available in the same order.</p> <p>c) The menu bar was used, and all menus were available.</p> <p>d) Menu items were greyed out when appropriate, except in the CAL where they were available but not functional. (S)</p> <p>e) Some menu items were not available due to programming problems. (S)</p>	<p>a) Menus were available on a Floating palette in a top to bottom fashion arranged; Font, Style, Size, Font, Apple.</p> <p>b) The six main menus were always available.</p> <p>c) The menu bar, floating palette, and icon bar were used. (P)</p> <p>d) Menu items were greyed out when appropriate.</p> <p>e) Some menu items were not available due to programming problems. (S)</p>	<p>a) Menus were not used in the menu bar. (P) The menus were ordered differently in each function. (P)</p> <p>b) Menus were only available when appropriate. (Y)</p> <p>c) The menu bar was not used (Y). Only menus (not just menu items) appropriate to the function were available. (P)</p> <p>d) Menu items were greyed out when appropriate.</p> <p>e) Some menu items were not available due to programming problems. (S)</p>

TABLE C (continued). Seeded Inconsistencies

Consistent Version 1 (Control)	Inconsistent Version A	Inconsistent Version B
<p>a) Help followed a Noun-Verb metaphor . Point and click on the object, with the option key down, to brings up a Help window next to the object.</p> <p>b) Each object had its own help window, so multiple windows could be viewed at once. Some help windows overlapped. (P)</p> <p>c) Help was also available in "Complicated dialog boxes (but not simple 'OK' - 'Cancel' dialog boxes), and will work with menus. (S)</p> <p>d) CMD-H was supported.</p> <p>e) Help windows disappear when moving between functions or changing pages within the function.</p> <p>f) Each Help window was removed when the user clicked on the Help-info field.</p> <p>g) Some objects did not have any Help information. (S)</p> <p>h) Some objects responded to clicks for help (instead of option-clicks). (Y)</p> <p>i) Option-Clicks on objects opened the Help windows</p>	<p><b>5) Help System</b></p> <p>a) Help uses a Verb-Noun metaphor. (Y) Clicking on the Help icon changed the cursor to a 'Help' cursor. Help windows opened for all objects that were clicked Closing the Help window or pressing the shift key exited the help mode.</p> <p>b) Each object has its own help window, so multiple windows can be viewed at once. Some help windows overlapped. (P)</p> <p>c) Help was also available in "Complicated" dialog boxes (but not simple 'OK' - 'Cancel' dialog boxes), and worked with menus. Both were activated with option-click. (S)</p> <p>d) CMD-H was supported.</p> <p>e) Help windows disappear when moving between functions or changing pages within the function.</p> <p>f) All Help windows were cleared after the next non-help action. Holding the option key down continuously while clicking on an icon would have kept the help box in view. (Y)</p> <p>g) Some objects did not have any Help information. (S)</p> <p>h) Some objects responded to clicks for help (instead of option-clicks). (Y)</p> <p>i) Option-Clicks on objects opened the Help windows for a brief moment before they would close. (Y)</p>	<p>a) Help uses a Verb-Noun metaphor. (Y) The Help window that appears was not "Linked" to the object. It will always appear in the bottom right corner of the screen (more consistent in placement, but not consistent in context).</p> <p>b) Each object has its own help window, but all overlapped, so multiple windows could not be viewed at once. (P)</p> <p>c) Help was available in "Complicated dialog boxes (but not simple 'OK' - 'Cancel' dialog boxes), and worked with menus. Both were activated with option-click. (S)</p> <p>d) CMD-H was supported.</p> <p>e) Help windows disappear when moving between functions or changing pages within the function.</p> <p>f) Help windows were removed on the next click of the mouse following the click that showed the help-info field. (Y)</p> <p>g) Some objects did not have any Help information. (S)</p> <p>h) Some objects responded to clicks for help (instead of option-clicks). (Y)</p> <p>i) Option-Clicks on objects opened the Help windows for a brief moment before they would close. (Y)</p>

TABLE C (continued). Seeded Inconsistencies

Consistent Version 1 (Control)	Inconsistent Version A	Inconsistent Version B
<p>a) Arrow icons were greyed out when not available (see User Feedback)</p> <p>b) Arrow icons were always placed in the iconic menu bar.</p>	<p><b>6) Navigation Arrows</b></p> <p>a) Arrow icons were never greyed out (see User Feedback). (F)</p> <p>b) Arrow icons were always placed on the card/page of the function. (P) They were not used on the PS or OV functions. (Y)</p>	<p>a) Arrow icons were never greyed out (see User Feedback). (F)</p> <p>b) Arrow icons were always available on the Navigator palette.</p>
<p>a) The WP function icon always navigated to the page that contains the current day.</p> <p>b) The Address Book and AC icons return the subject to the previous location or the beginning for the first visit in a session.</p> <p>c) The OV, and PS icons always navigate to the same place since there was only one page for each of these functions.</p> <p>d) The TDL icon navigated to the first page in the function. (M)</p>	<p><b>7) Icons for Functions</b></p> <p>a) The CAL and WP function icons were used to navigate to the last location visited. (M)</p> <p>b) The Address Book and AC icons return the subject to the previously known location.</p> <p>c) Same as in Consistent version.</p> <p>d) The TDL icon navigated to the first page in the function. (M)</p>	<p>a) The CAL and WP function icons were used to navigate to the last location visited. (M)</p> <p>b) The Address Book and AC icons return the subject to the previously known location.</p> <p>c) Same as in Consistent version.</p> <p>d) The TDL icon navigated to the first page in the function. (M)</p>

TABLE C (continued). Seeded Inconsistencies

Consistent Version 1 (Control)	Inconsistent Version A	Inconsistent Version B
<p><b>8) Find within Function</b></p> <p>a) Available in the Roledex, Area Codes, WP, and To Do functions.            b) Words that were found were saved for use in the next search.            c) CMD-F and the Find icon were available.            d) A find initiated with a selection of text will use the selected text. The text string could not contain returns or quotes. (T)            e) The Find feature was not available in the Phone Dial window. (S)            f) The Find feature was not available in the OV or PS.</p>	<p>a) Same as consistent version.            b) Find did not remember the last word searched. (M)            c) CMD-F and the Find icon were available.            d) It did not support the selection of text. (T)            e) Same as in consistent version. (S)            f) Same as consistent version.</p>	<p>a) Same as consistent version.            b) Find did not remember the last word searched. (M)            c) CMD-F and the Find icon were available.            d) It did not support the selection of text. (T)            e) Same as in consistent version. (S)            f) Same as consistent version.</p>
<p><b>9) Find between Functions</b></p> <p>a) Words that were found were saved for use in the next search.            b) Find searches the visible fields in that function. In the WP or TDL, Find searched these plus the CAL. If the string was found in the current function the program navigated to the found string, but if it was found somewhere else it would return a message informing the user that the string did not appear in the current function, but did appear in the {other} function. Find always searched the entire current function before searching in the other two functions. Note: A Find could not be started from the CAL.</p>	<p>a) Words that were found were not saved for use in the next search. (M)            b) Same as consistent version.</p>	<p>a) Words that were found were not saved for use in the next search. (M)            b) Same as consistent version.</p>

TABLE C (continued). Seeded Inconsistencies

Consistent Version 1 (Control)	Inconsistent Version A	Inconsistent Version B
<p>a) The function keys were as follows            1-4 Undo/Cut/Copy/Paste            5-8 First Card/Prev Card/Next Card/ First card            9-12 Prev Function/AR/PS/Area Codes            13-15 Yearly CAL/WP/TDL</p> <p>b) The F1-F4 were the standard Apple edit keys, the next four were the within function navigation keys, followed by the return to previous function button and then the six functions (in the same order as they appear on the top of the main window).</p>	<p><b>10) Function Key Layout</b></p> <p>a) The function keys were as follows;            1-4 AR/Area Codes/Copy/Cut            5-8 First card/Last Card/Next card/Paste            9-12 PS/Prev Card/Prev Function/TDL            13-15 Undo/WP/Yearly CAL</p> <p>b) They were ordered alphabetically. (P)</p>	<p>a) The function keys were as follows;            1-4 Copy/Paste/Cut/Prev Card            5-8 Next Card/First Card/Last Card/AR            9-12 Yearly CAL/WP/Area Codes/TDL            13-15 Prev Functions/Undo/PS</p> <p>b) They use the same basic functional grouping as in version 1, except they were ordered left to right by how often the function was used. The undo and PS were placed at the end since they were almost never used. (P)</p>
<p>a) Click in any field to edit.</p>	<p><b>11) Phone Setup</b></p> <p>a) Fields were locked and changes could only be made by using dialog boxes (D), except for the Local Area Code and Prefix fields which could be edited directly. (Y)</p>	<p>a) Same as Inconsistent version A. (D,Y)</p>
<p>a) Icons were available to copy and paste from the field without selecting the text.            b) The edit keys were supported.            c) The layout of the buttons was organized left to right.</p>	<p><b>12) Dialing a Number</b></p> <p>a) Same as in the consistent version.            b) The edit keys were supported.            c) The layout of the buttons was organized in small chunks. (P)</p>	<p>a) Same as in the consistent version.            b) The edit keys were supported.            c) The layout of the buttons was organized in small chunks. (P)</p>

TABLE C (continued). Seeded Inconsistencies

Consistent Version 1 (Control)	Inconsistent Version A	Inconsistent Version B
<b>13) Error and Other Messages</b>		
<p>a) All error messages were presented in the message box.                      b) Trying to use the Return arrow to return to a To Do page that had been deleted resulted in an error messages and a beep.</p>	<p>a) All error messages were presented in the message box.                      b) N/A, a return key was not available.</p>	<p>a) All error messages were presented in the message box.                      b) N/A, a return key was not available.</p>
<b>14) Auditory Feedback</b>		
<p>a) The standard 'Bad' beep Macintosh beep was used to signal that something was wrong. Good beeps were used to signal the completion of a successful long action (like adding a card, or page, or finishing a task).                      b) If a " Find" was performed incorrectly there was a bad beep.</p>	<p>a) Audio feedback was not given at all during the tasks (F), but a 'Bad' beep was used to signal the completion of a task. (S)                      b) If a " Find" was performed incorrectly there was a bad beep.</p>	<p>a) Audio feedback was not given at all during the tasks (F), but a 'Bad' beep was used to signal the completion of a task.(S)                      b) If a " Find" was performed incorrectly there was a bad beep.</p>
<b>15) Visual Feedback</b>		
<p>a) Watch cursor appears during long (&gt;1 sec) waits                      b) Icons highlighted to show the initiation of an action.                      c) Icons that were consistent across functions show no movement during transitions. Icon placement was exact, so no apparent motion was visible.                      d) Help fields were outlined in a 2 pt. shadow box                      e) Pop-up menus highlighted.</p>	<p>a) A watch cursor was not used. (F)                      b) Some icons did not highlight. (F)                      c) Some apparent motion (up to two pixel offset) was present. (P)                      d) Help fields have a variety of shadow effects from none to 3 points. (F)                      e) Pop-up menus highlighted.</p>	<p>a) The cursor changes to a small version of current function icon and not the icon for where the subject was going to in the application. (F)                      b) Some icons did not highlight. (F)                      c) The navigator was always available so icon jitter was not possible for the function icons. Other icons were placed in distinct locations on each function, so jitter was not applicable. (P)                      d) Help fields have a variety of shadow effects from none to 3 points. (F)                      e) Pop-up menus highlighted.</p>

TABLE C (continued). Seeded Inconsistencies

Consistent Version 1 (Control)	Inconsistent Version A	Inconsistent Version B
<p>a) A Clock was always available in the same position in all functions.</p>	<p>a) The time was available in the CAL and in the WP. (F)</p>	<p>a) The time was available on the Navigator and the CAL and WP. (F)</p>
<p><b>16) Clock Function</b></p>		
<p><b>17) Wordings</b></p>		
<p>a) The wording of the Find prompt was always "Find . . . in the {function name goes here} function?"</p> <p>b) Location menu choices, from the AR, were all in uppercase style to differentiate them from words types into address fields. The Dial window also used uppercase for the locations.</p> <p>c) "NO LOCATION" was presented as the button name when a phone number was entered, but before the location was set. It was only available when a phone number was present in the corresponding field.</p> <p>d) The Location menu allowed users to type in a string to be used as the location setting, if an existing menu choice was not appropriate. The application truncated the string to 11 characters, and formatted the text to uppercase.</p> <p>e) Text feedback was limited to the size of the message box. (F)</p>	<p>a) The wording of the Find function varied; "Search" (To Do function), "Locate" (Area Code function), "Find" (WP), and "Go to" (AR) began the Find prompt. (S)</p> <p>b) Location choices were in uppercase. Once one was selected the choice was shown in title style. (L) The Dial window used both uppercase and title style for the location settings. (L)</p> <p>c) The "NO LOCATION" button was always available until it was changed, even if no phone number was present (S)</p> <p>d) The Location menu allowed users to type in a string to be used for the location setting, if an existing menu choice was not appropriate. The application truncated the string to 11 characters, and formatted the text to uppercase.</p> <p>e) Text feedback was limited to the size of the message box. (F)</p>	<p>a) The wording of the Find function varied; "Search" (To Do function), "Locate" (Area Code function), "Find" (WP), and "Go to" (AR) began the Find prompt. (S)</p> <p>b) Location choices were in uppercase. Once one was selected the choice was shown in title style. (L) The Dial window used just uppercase for the location settings.</p> <p>c) Same as the consistent version.</p> <p>d) The Location menu allowed users to type in a string to be used for the location setting, if an existing menu choice was not appropriate. The application truncated the string to 11 characters, and formatted the text to uppercase.</p> <p>e) Text feedback was limited to the size of the message box. (F)</p>
<p>a) The wording of the Find prompt was always "Find . . . in the {function name goes here} function?"</p> <p>b) Location menu choices, from the AR, were all in uppercase style to differentiate them from words types into address fields. The Dial window also used uppercase for the locations.</p> <p>c) "NO LOCATION" was presented as the button name when a phone number was entered, but before the location was set. It was only available when a phone number was present in the corresponding field.</p> <p>d) The Location menu allowed users to type in a string to be used as the location setting, if an existing menu choice was not appropriate. The application truncated the string to 11 characters, and formatted the text to uppercase.</p> <p>e) Text feedback was limited to the size of the message box. (F)</p>	<p>a) The wording of the Find function varied; "Search" (To Do function), "Locate" (Area Code function), "Find" (WP), and "Go to" (AR) began the Find prompt. (S)</p> <p>b) Location choices were in uppercase. Once one was selected the choice was shown in title style. (L) The Dial window used both uppercase and title style for the location settings. (L)</p> <p>c) The "NO LOCATION" button was always available until it was changed, even if no phone number was present (S)</p> <p>d) The Location menu allowed users to type in a string to be used for the location setting, if an existing menu choice was not appropriate. The application truncated the string to 11 characters, and formatted the text to uppercase.</p> <p>e) Text feedback was limited to the size of the message box. (F)</p>	<p>a) The wording of the Find function varied; "Search" (To Do function), "Locate" (Area Code function), "Find" (WP), and "Go to" (AR) began the Find prompt. (S)</p> <p>b) Location choices were in uppercase. Once one was selected the choice was shown in title style. (L) The Dial window used just uppercase for the location settings.</p> <p>c) Same as the consistent version.</p> <p>d) The Location menu allowed users to type in a string to be used for the location setting, if an existing menu choice was not appropriate. The application truncated the string to 11 characters, and formatted the text to uppercase.</p> <p>e) Text feedback was limited to the size of the message box. (F)</p>

TABLE C (continued). Seeded Inconsistencies

Consistent Version 1 (Control)	Inconsistent Version A	Inconsistent Version B
<b>18) Using Tab to Navigate</b>		
<p>a) Click in the field or use the Tab and Shift-Tab keys to enter a field.</p> <p>b) The tab keys moves the cursor into fields from top to bottom of the left page and then top to bottom on the right page, or just top to bottom is the function was only one page/card.</p> <p>c) Shift Tab work opened fields in reverse order.</p> <p>d) After deleting or adding a card to the AR the cursor was positioned in the first field (the name/address field).</p>	<p>a) Click in the field or use the Tab and Shift-Tab keys to enter a field.</p> <p>b) Tab movement was supported from left to right in the WP, down to up in the Area Code function, and in random order in the AR. (S) The PS field had only two fields which could be edited, so the tab key would toggle between those tow fields and would not work with the other fields. (D)</p> <p>c) Shift Tab work opened fields in reverse order.</p> <p>d) After deleting or adding a card to the AR the cursor was randomly positioned in a field. (S)</p>	<p>a) Click in the field or use the Tab key to enter a field.</p> <p>b) Tab movement was supported from left to right in the WP, down to up in the Area Code function, and in random order in the AR. (S) The PS field had only two fields which could be edited, so the tab key would toggle between those tow fields and would not work with the other fields. (D)</p> <p>c) Shift Tab work opened fields in reverse order.</p> <p>d) After deleting or adding a card to the AR the cursor was randomly positioned in a field. (S)</p>
<b>19) Color</b>		
<p>a) Color was not used in this version.</p>	<p>a) Color was used only on the opening screen and on the menu palette for background color. (F)</p>	<p>a) The palette used color coding in the background of the icons. The previous and next arrows were light blue, the first and last arrow keys were green. The function icons were colored a shade of green. The other icons had different background colors. (F)</p>

TABLE C (continued). Seeded Inconsistencies

Consistent Version 1 (Control)	Inconsistent Version A	Inconsistent Version B
<b>20) Floating Palettes</b>		
<p>a) Prompt boxes always appear in the center of the screen</p> <p>b) The Quick jump windows always appeared under the quick jump icon.</p> <p>c) The Quick key list window used the standard zoom box.</p> <p>d) Clicking and holding down the mouse on the window's title bar allowed the user to move the palette.</p>	<p>a) Prompt boxes appear in a randomly generated position on the screen. (P)</p> <p>b) The Quick jump windows always appeared under the quick jump icon.</p> <p>c) The Quick key list window used an unusual icon in place of the standard zoom box. (D)</p> <p>d) Clicking and holding down the mouse on the window's title bar allowed the user to move the palette. (D)</p>	<p>a) Prompt boxes appear in a randomly generated position on the screen. (P)</p> <p>b) The Quick jump windows always appeared under the quick jump icon.</p> <p>c) The Quick key list window used the standard zoom box.</p> <p>d) Clicking and holding down the mouse on the window's title bar did nothing, a special icon was used to allow the user to move the palette. (D)</p> <p>e) An animated "Moving Hand" icon was used when the window was being moved. (F)</p>
<b>21) Function Specific Actions</b>		
<p>a) Fields in the TDL and WP were for 'extended writing'. They were standard fields and allow one to write past the bounds, resulting in information being outside the visible area.</p>	<p>a) Scrolling fields were used in the Planner, but not in the TDL. (D)</p>	<p>a) Scrolling fields were used in both the WP and the TDL (D)</p>
<b>22) Pop-up Menus/ Icons</b>		
<p>a) The 3 location menus, on each page of the AR, were only available once a phone number was entered into the corresponding field.</p> <p>b) A "(no location)" message was only visible if a phone number was not entered into the corresponding field.</p> <p>c) Sort used a standard dialog box to choose the method of sorting.</p>	<p>a) The 3 location menus were always available. (S)</p> <p>b) N/A, the location menus were always available.</p> <p>c) Same as in consistent version.</p>	<p>a) Same as in the Consistent version.</p> <p>b) Same as in the Consistent version.</p> <p>c) Sort used a pop-up menu top choose the method of sorting (First or Last name). (D)</p>

TABLE C (continued). Seeded Inconsistencies

Consistent Version 1 (Control)	Inconsistent Version A	Inconsistent Version B
<p><b>Other Layout Items</b></p> <p>The items were covered previously, all rules pertaining to the menu bar area are recapped.</p>		
<p>a) The center top area of the menu bar contained the functions was ordered from left to right; AR,PS,AC (related to telephones), CAL,WP, TDL (related to scheduling). The currently active icon was greyed out.</p> <p>b) Navigation icons from left to right; First, Previous, Next, and Last page of function, and Most Recent Page of Previous Function. They were greyed out when not appropriate.</p>	<p>a) The same as in the Consistent version.</p> <p>b) Navigation icons were placed in the card areas consistent with the function and not the global rule. In the PS and OV did not have any navigation icons. Navigation icons were never greyed out, and produced no feedback.</p> <p>c) The same as in the Consistent version.</p>	<p>The OWN was used in place of the icon bar.</p> <p>a) The icons were placed on the OWN (in a 2 by 7, or 7 by 2 arrangement). Since it was always visible the function icons did not change place or have any jitter when moving between functions.</p> <p>b) Navigation icons were placed on the navigator. They were never greyed out and did not produce feedback.</p>
<p>c) System Functions (top right) Help and Quit, were in the right corner of the window.</p> <p>d) General icons (Dial and Return to OV) were placed to the right of the navigation icons. They were always available. The time was also placed in this area.</p> <p>e) Function specific icons were placed to the left of the function icons. These pertained to all current cards or pages. The 'Find' function, when available, was always placed closest to the function icons. Icons were added from right to left in order of importance.</p> <p>f) The page or card specific icons were placed to the left. These icons were for the current page or card. The Quick jump icon, when available, was placed in this section.</p>	<p>d) Time was available of the menu palette and in the CAL and WP functions. The other icons were placed on the floating palette.</p> <p>e) They were placed in open spaces in each function, so they do not appear in the same location on every function.</p> <p>f) These were placed in the card area.</p>	<p>c) System functions do not have the button outline and were smaller and not aligned correctly on the OWN.</p> <p>d) Time was available of the navigator and in the CAL and WP functions. The other icons were placed on the floating palette.</p> <p>e) Sort used a pop-up menu for sorting by last or first name.</p>
		<p>f) These icons were available in the main window, but related icons (delete or ad d) did not appear in the same location between functions.</p>

TABLE C-D. Defining Classes of Inconsistencies

<p>Originally the CLG was going to be used to define the problems in the interface, but the author did not feel it was robust enough to define the incon. noted by participants. The CLG does not make distinctions for some objects and incon. specific to a graphical user interface.</p> <p>The six levels of Moran's CLG were broken down into eight classes. The divisions were formulated by the author after observing subjects using the Personal Organizer. Moran's grammar did not directly apply to some incon., thus new definitions and classes were created. The author does not imply that this scheme is tested or validated, but it did provide a method for defining the incon. in this experiment.</p> <p>The bounds of each of these classes can overlap a great deal depending on the inconsistency. Note how the lexical level also includes changes in iconic representations. In the same manner that the word "Find" might be substituted for "Search", a different icon might be used to represent actions of similar meanings. This could also be classified as a semantic inconsistency. For example, if the participant mentioned that the icons represent the wrong concept or metaphor then it would be defined in the semantic class, but if the participants only determined that two icons did the same function, or that two similar icons (such as the phone icons) performed different function, then it would be classified into the Lexical class. Research into the application of this classification scheme is currently underway.</p>	<p><b>Physical Location (P)</b> - This is equivalent to the Spatial level defined by Moran. Incon. in the variation in the location of an object would be included. It also could include the relative positioning of menu choices or how they were grouped (or separated), similar objects having different locations, or different objects having similar locations.</p> <p><b>Memory (M)</b> - When the program does not "remember" an event, object position, system state, or other activity that a subject expects to be remembered, it was classified as a memory inconsistency. Other memory incon. could occur with window locations, previous search strings, and the storage of entered keystrokes or mouse clicks.</p> <p><b>Semantic (S)</b> - This level includes the memory group in the CLG, but also would include items that had different meaning for similar words/icons or different words /icons with similar meanings. If the current system state implied that a specific function or option of a function is available or not available, then these changes would be considered semantic inconsistencies. This level enumerates the concepts but not how to access them.</p> <p><b>Syntactic (Y)</b> - The syntactic level includes high level objectives, methods for entering a command or initiating an action, and shortcuts. The rules for when items were presented (or not presetted) and when one method for initiating a command was available (or unavailable) would be defined in the syntactic class. The level specifies how the system operations are activated.</p>	<p><b>Virtual Device (D)</b> - Device refers to the method of activation or usage. It refers to system objects that a user acts upon, such as scrolling window, scrolling list, pull down menu, hierarchical menu, check box or dialog box. Users created changes in the state of the interface with the "virtual devices." This level specifies the appropriateness of the device and how it is operated.</p> <p><b>System State, or Feedback (F)</b>- Any feedback that gives the user some information about the current state of the system, or what state the system would be in if an action was initiated. The arrows, used to indicate to the user that a pull down menu was available, would be present in this level. Changes in the cursor, other visual feedback, audio feedback, window boxes, or other properties that make the user aware of the current system status were included in this level. Changes in the graphic layout of the screens would also be included. Removing the "look" of a Roledex card would be included here.</p> <p><b>Lexical (L)</b> - This level includes all changes in the wording or representation of any of the application's language. This also includes pictographic (iconic) representations. This could include error messages, prompts, and icons used to represent system properties or functions, abbreviations, and capitalization.</p> <p><b>Task (T)</b> - (unavailable functions) - This contains system features that were not implemented in the application, but should have been available to the user (actual missing functions and not functions available and simply not discovered by the user). Unavailable functions were "equally" unavailable in all versions.</p>
----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

TABLE C-C. Summary of Inconsistencies by Class

Class of Inconsistency	Seeded Inconsistencies By Versions Used by Groups		
	Group 1 (Consistent)	Group 2 (I-Ver A)	Group 3 (I- Ver. B)
Physical Location	1	18	18
Memory	1	5	5
Syntactic	2	11	10
Semantic	5	10	9
Virtual Device	0	5	5
System Status (Feedback)	1	8	12
Lexical	0	5	3
Totals *	10	60	62

\* Totals do not include the number of Unavailable Tasks.

**NOTE:**

A summary of the seeded inconsistencies is presented in Table C-C. These reflect the cumulative total of all inconsistencies, defined by the author, in Table C. The seeded inconsistencies are also summed by Topic in Table C-T. An effort to determine the effect of a specific type of inconsistency is underway.

TABLE C-T. Summary of Inconsistencies by Topic

Topic	Seeded Inconsistencies By Versions		
	Group 1 (Consistent)	Group 2 (I-Ver A)	Group 3 (I- Ver. B)
1- Menu Structure	0	7	6
2- Icon Placement	0	9	9
3- Edit Keys	0	3	3
4- Menu Item Order	2	2	6
5- Help System	4	7	8
6- Navigation Arrows	0	2	2
7- Icons for Function	1	2	2
8- Find within Function	1	2	2
9- Find between Functions	0	1	1
10- Function Key Layout	0	1	1
11- Phone Setup	0	2	2
12- Dialing a Number	0	1	1
13- Error and Other Messages	0	0	0
14- Auditory Feedback	0	2	2
15- Visual Feedback	0	4	4
16- Clock Function	0	1	1
17- Wording	1	5	3
18- Using Tab Keys to Nav.	0	3	3
19- Color	0	1	1
20- Floating Palettes	0	3	3
21- Function Specific Actions	0	1	1
22- Pop-up Menus/Icons	0	1	1
Totals	10	60	62

---

# APPENDIX D: Post-test Questionnaire

Please rate the interface you just used on these scales. Circle a number to represent your impressions of what you feel is appropriate. Note: The scales do not always go from “Good” to “Bad” (left to right), so please read each pair carefully!

The interface refers to the objects, actions, and methods used to work with the application. The interface includes every action you perform and input you make while using the application.

**The interface was**

Pleasing							Irritating
1	2	3	4	5	6	7	

Unfriendly							Friendly
1	2	3	4	5	6	7	

Complete							Incomplete
1	2	3	4	5	6	7	

Cooperative							Uncooperative
1	2	3	4	5	6	7	

Undependable							Dependable
1	2	3	4	5	6	7	

Complicated							Simple
1	2	3	4	5	6	7	
Inconsistent							Consistent
1	2	3	4	5	6	7	
Natural							Unnatural
1	2	3	4	5	6	7	
Intelligent							Unintelligent
1	2	3	4	5	6	7	
Uninterpretable							Interpretable
1	2	3	4	5	6	7	
Fast							Slow
1	2	3	4	5	6	7	
Adaptive							Unadaptive
1	2	3	4	5	6	7	
Useful (functions and commands)							Useless
1	2	3	4	5	6	7	
Redundant (functions)							Concise
1	2	3	4	5	6	7	

Cluttered (Screen layout)

1      2      3      4      5      6      7

Uncluttered

Unsafe (Data security)

1      2      3      4      5      6      7

Safe

Maintainable

1      2      3      4      5      6      7

Unmaintainable

Easy to learn

1      2      3      4      5      6      7

Difficult to learn

The application consists of the interface and the underlying functionality and processing that creates the information available to you.

**The application made me feel**

Not frustrated

1      2      3      4      5      6      7

Frustrated

Unproductive

1      2      3      4      5      6      7

Productive

that it is not worth buying

1      2      3      4      5      6      7

that it is worth buying

Relaxed							Tense
1	2	3	4	5	6	7	
Optimistic							Pessimistic
1	2	3	4	5	6	7	
Competent							Incompetent
1	2	3	4	5	6	7	
Pleased							Disgusted
1	2	3	4	5	6	7	
Bored							Engaged
1	2	3	4	5	6	7	
Unhappy							Happy
1	2	3	4	5	6	7	
Intelligent							Stupid
1	2	3	4	5	6	7	
Pleasant							Annoyed
1	2	3	4	5	6	7	

What about the interface did you like?

---

---

---

---

What do you think we should change in the interface in future revisions? ( What could be changed in this program that would make it more “Mac-like”)

---

---

---

---

Can you think of any addition features or functionality that should be added to the application or the interface, if so what?

---

---

---

---

---

# APPENDIX E: Post-Experiment Questionnaire

Note: Actual Questionnaire had more room for free response answers.

**Please ask the investigator to open the second version of the application.**

Did you find the second program easy to learn?

Yes, very much

1

2

3

4

5

No, not at all

6

7

When you worked with the program was it consistent with how you expected it to act?

Yes, very much

1

2

3

4

5

No, not at all

6

7

Do you think if the interface in the second program was more consistent it would have been easier to use?

Yes, very much

1

2

3

4

5

6

No, not at all

7

Did the inconsistencies in the second version affect the time it took to complete the tasks?

Yes, very much

1

2

3

4

5

No, not at all

6

7

How many minutes do you think it took to complete each of the six sessions?

1 \_\_\_\_\_ 2 \_\_\_\_\_ 3 \_\_\_\_\_ 4 \_\_\_\_\_ 5 \_\_\_\_\_ 6 \_\_\_\_\_

How would you compare the third version of the program to the second?

Better			Second Version				Worse
1	2	3	4	5	6	7	
Confusing			Second Version				Understandable
1	2	3	4	5	6	7	
Easier			Second Version				Harder
1	2	3	4	5	6	7	
More Consistent			Second Version				Less Consistent
1	2	3	4	5	6	7	
Simpler			Second Version				Complicated
1	2	3	4	5	6	7	

What, in general, did you like about the second program you used?

---

What, in general, did you **not** like about second program you used?

---

Overall, how would you compare the three versions of the program?

First & Second:	Equal						Different
	1	2	3	4	5	6	7
Second & Third:	Equal						Different
	1	2	3	4	5	6	7
First & Third:	Equal						Different
	1	2	3	4	5	6	7

Try to compare the Consistency or Inconsistency (circle one) of the second versions features to the other two versions as well as other Macintosh and non-Macintosh programs that you have used.

Example:

The Undo Command (Con - Incon) Why? It would not let me undo my most recent command and it used a different function key than the standard Apple programs

1. Menu Structure ( Con - Incon ) Why? \_\_\_\_\_

2. Icon Placement ( Con - Incon ) Why? \_\_\_\_\_

3. Edit Keys ( Con - Incon ) Why? \_\_\_\_\_

4. Menu items order ( Con - Incon ) Why? \_\_\_\_\_

5. Help system ( Con - Incon ) Why? \_\_\_\_\_

6. Navigation arrows ( Con - Incon ) Why? \_\_\_\_\_

7. Icons for the Functions ( Con - Incon ) Why? \_\_\_\_\_

8. 'Find' within a function ( Con - Incon ) Why? \_\_\_\_\_
9. 'Find' between functions ( Con - Incon ) Why? \_\_\_\_\_
10. Function Key layout ( Con - Incon ) Why? \_\_\_\_\_
11. Phone Setup input ( Con - Incon ) Why? \_\_\_\_\_
12. Dialing a number ( Con - Incon ) Why? \_\_\_\_\_
13. Error and other Messages ( Con - Incon ) Why? \_\_\_\_\_
14. Auditory feedback ( Con - Incon ) Why? \_\_\_\_\_
15. Visual feedback ( Con - Incon ) Why? \_\_\_\_\_
16. Clock function ( Con - Incon ) Why? \_\_\_\_\_
17. Wording (uppercase/lowercase etc...) ( Con - Incon ) Why? \_\_\_\_\_
18. Using Tab keys to navigate ( Con - Incon ) Why? \_\_\_\_\_
19. Color ( Con - Incon ) Why? \_\_\_\_\_
20. Floating Palettes ( Con - Incon ) Why? \_\_\_\_\_
21. Function specific actions (Sorting, Find, etc...) ( Con - Incon ) Why? \_\_\_\_\_
22. Pop-up Menus/ Icons ( Con - Incon ) Why? \_\_\_\_\_

---

## APPENDIX F: Tasks Performed By Subjects

The text for each of the 48 tasks used in this experiment is presented. The Block number and task type referenced (Block #-Task Type) can be compared to the general tasks explained in Table 4.

1-A. See if you are available for dinner on August 21, 1990 in the Weekly Planner. If you are not available reschedule what is there for the same time in the previous week. Then put down "Dinner with the Stevens family" for that day. •

1-B. Find Alan Chao's telephone number and put his name and lab number with "Call about weekly parts supply" in the Weekly Planner on September 7 and 14, 1990. Do not go to the Address Roledex to find this information! •

1-C. Go to the Phone Setup function and add "724" to the list of prefixes, then change the Long Distance code to "28" (Please keep a space between the prefixes). •

1-D. Delete the last two pages of the To Do function (even if they are empty). Put "Birthday Party List" in 14 pt italic type at the top of the visible page on the right. Put "Jamie, Karen, Steve, and Rob" under it in 14 pt. plain type. Insert this information before anything on the current page. •

1-E. Go to "Kentucky" (area code 606) in the Area Code function. Place "Baseline, Sweet Briar, Waterpoint" in the list of locations, in their correct alphabetical order. •

1-F. Delete (not Erase) the cards in the Address Roledex for Ira Ungermann and Steve Jones, and Royce Walthrop. •

1-G. Find the meeting with Jake Burns in the Weekly Planner. Verify that the name of his company is correctly identified as compared to what is in the Address Roledex. Correct the Planner entry (not the Address Roledex), if necessary. •

1-H. Get Help for the “Sort” icon in the Address Roledex function. You will verify that you are getting Help for the correct function by reading the information out loud. Do not actually perform a “Sort.” •

2-I. Add “Meeting with Boss at noon” in bold text for October 5th and 19th, 1990 in the Weekly Planner. •

2-J. Find the area code for Carmel, California in the Area Code function. Put it in front of both telephone numbers on the two address cards in the Address Roledex that show Carmel, California for an address. •

2-K. Go to the last card in the Address Roledex and sort the cards by last name. •

2-L. Go to pages 5 and 6 of the To Do function and add two more pages. Put “Phase 2 work group” in 14 pt underline type at the top of the visible page on the right. Put “Scully, Jobs, Bush, Miller, Swede, Willis, Jones” under it in 14 pt. plain type. •

2-M. Add a card to the Address Roledex with “George Bush White House Washington D.C. 20345.” The first telephone field should have “345-0567”, with the ‘Location’ button set to “OFFICE“. •

2-N. Dial Major Greer’s lab number with the speaker setting from the Weekly Planner page with his name on it. Do Not go to the Address Roledex. •

2-O. Find the meeting in Montreal with Robert Stevens in your Weekly Planner. Make sure his Montreal phone number has the correct area code, as compared to what is found in the Area Code function. Correct the Planner entry, if necessary. •

2-P. Get Help for the 'Toll Call' option from within the Phone Setup function. You will verify that you are getting Help for the correct function by reading the information outloud. Do not actually change the 'Toll Call' option. •

3-A. See if you are available for lunch on September 13, 1990 in the Weekly Planner. If you are not available reschedule what is there for the same time in the following week. Then put down "Lunch with Apple Computer" for that day. •

3-B. Find Betty Carr's telephone number and put her name and office phone number with "Phase-2 Overview Meetings" in the Weekly Planner on October 3 and 10, 1990. Do not go to the Address Roledex to find this information! •

3-C. Go to the Phone Setup function and add "725" to the list of prefixes, then change the Outside Line to "9" (Please keep a space between the prefixes). •

3-D. Delete the last two pages of the To Do function (even if they are empty). Put "Mac User Group" in 14 pt bold type on the visible page on the right. Put "Jay, Sarah, Barbara, Irwin" under it in 14 pt. plain type. Insert this information before anything on the current page. •

3-E. Go to "Georgia" (area code 404) in the Area Code function. Place "Cobb, East Point, and Vinnings" in the location list, in their correct alphabetical order. •

3-F. Delete (not Erase) the cards in the Address Roledex for Jerry Garcia and Roy Swain, and Karl Daly. •

3-G. Find the meeting with Andy Cohen in the Weekly Planner. Verify that the name of his company is correctly identified as compared to what is in the Address Roledex. Correct the Planner entry (not the Address Roledex), if necessary. •

3-H. Get Help for the "Phone Setup" icon in the Weekly Planner function. You will verify that you are getting Help for the correct function by reading the information outloud. Do not actually use the "Find" operation. •

4-I. Add “Ski with club if there’s snow” in italic text for January 6th and 20th, 1991 in the Weekly Planner. •

4-J. Find the area code for Mobile, Alabama in the Area Code function. Put it in front of both telephone numbers of the two Address Roledex cards that show Mobile, Alabama for an address. •

4-K. Go to the last card in the Address Roledex and sort the cards by last name. •

4-L. Go to pages 5 and 6 of the To Do function and add two more pages. Put “Groceries for Party” in 18 pt bold type at the top of the left visible page. Put “Grapes, Milk, Cheese, Chips, Coke, Beer, and Eggs” under it in 18 pt. plain type. •

4-M. Add a card to the Address Roledex with “Kay Whitmore Eastman Kodak Company Rochester, NY 14653.” The first telephone field should have “724-5150”, with the ‘Location’ button set to “OFFICE.” •

4-N. Dial Dorris Schwartz’s Office number with the modem (tone) setting from the Shopping List Page of the To Do list’s. Do Not go to the Address Roledex. •

4-O. Find the meeting and phone number for Sierra Capitol in your Weekly Planner. Make sure the phone number has the correct area code, as compared to what is found in the Area Code function. Correct the Weekly Planner entry, if necessary. •

4-P. Get Help for the “Dial Number” icon in the Area Code function. You will verify that you are getting Help for the correct function by reading the information that is presented outloud. Do not actually bring up the “Dial” window. •

5-A. See if you are available for dinner on August 28, 1990 in the Weekly Planner. If you are not available reschedule what is there for the same time in the following week. Then put down “Dinner with the In-laws” for that day. •

5-B. Find Ted Holt's telephone number and put his name and lab number with "Schedule group meeting" in the Weekly Planner on September 3 and 10, 1990. Do not go to the Address Roledex to find this information! •

5-C. Go to the Phone Setup function and add "726" to the list of prefixes, then change the International code to "011." (Please keep a space between the prefixes). •

5-D. Delete the last two pages of the To Do function (even if they are empty). Put "Basketball Team" in 14 pt italic type on the visible page on the left. Put "Me, John, Susan, Gloria, Jack" under it in 14 pt. plain type. Insert this information before anything on the current page. •

5-E. Go to "Florida" (area code 813) in the Area Code function. Place "Bradington, Coconut Grove, Willmount" in the location list, in their correct alphabetical order. •

5-F. Delete (not Erase) the cards in the Address Roledex for Dean Diffie and Beverley Richie, and Glenn Kelley. •

5-G. Find the meeting with Steve Arronan in the Weekly Planner. Verify that the name of his company is correctly identified as compared to what is in the Address Roledex. Correct the Planner entry (not the Address Roledex), if necessary. •

5-H. Get Help for the "Extend Planner" icon in the Weekly Planner function. You will verify that you are getting Help for the correct function by reading the information outloud. Do not actually "Extend" the planner. •

6-I. Add "Meeting with Copier team" in underline text for January 4th and 18th, 1991 in the Weekly Planner. •

6-J. Find the area code for Winchester, Virginia in the Area Code function. Put it in front of the both telephone numbers on the two Address Roledex cards that show Winchester, Virginia for an address. •

6-K. Go to the last card in the Address Roledex and sort the cards by last name. •

6-L. Go to pages 5 and 6 of the To Do function and add two more pages. Put “Movies to Rent” in 18 pt bold type at the top of the left visible page. Put “Die Hard, Mask, Batman, Star Wars, Animal House” under it in 18 pt. plain type. •

6-M. Add a card to the Address Roledex with “Jim Wilstern Eastman Kodak Company Rochester, NY 14653.” The first telephone field should have “726-9302”, with the ‘Location’ button set to “OFFICE.” •

6-N. Dial Dorris Schwartz’s Office number with the speaker setting from the Weekly Planner page with her name on it. Do Not go to the Address Roledex. •

6-O. Find the meeting with Bobby Heisman in your Weekly Planner. Make sure the phone number has the correct area code, as compared to what is found in the Area Code function. Correct the Weekly Planner entry, if necessary. •

6-P. Get Help for the ‘Extend Calendar’ icon in the Yearly (Six-Month) Calendar. You will verify that you are getting Help for the correct function by reading the information that is presented outloud. Do not actually “Extend” the calendar. •

---

# APPENDIX G: General Questionnaire

If you need additional space please ask the investigator for more paper. The more you say the better the results of this study.

What is your Age? \_\_\_\_\_ Male or Female? \_\_\_\_\_ Department ? \_\_\_\_\_

Faculty - Staff - Graduate Student - Undergraduate (circle one). If a student, what year? \_\_\_\_\_

How many years of experience with all computers have you had? \_\_\_\_\_

How many years of experience with Macintosh computers have you had? \_\_\_\_\_

What kind of computers do you use? \_\_\_\_\_

What kind of computers do you own? \_\_\_\_\_

About how many hours per week do you use a Macintosh computer? (circle one)

0-4 hrs/wk

4-8 hrs/wk

8-12 hrs/wk

12-16 hrs/wk

16-20 hrs/wk

20-24 hrs/wk

24-28 hrs/wk

more than 28 hrs/wk

In general, do you like working with computers?

Yes, very much

1 2

3

4

5

6

No, not at all

7

Do you find most programs on the Macintosh easy to learn?

Yes, very much

1 2

3

4

5

6

No, not at all

7

Why or why not?

---

---

# APPENDIX H: Instructions for Users

Thank-you for participating in this study of computer usage. This research is done, in an effort to better understand the human-computer interface. You are being asked to be part of the development process of a computer based application. Your thoughts, comments, and actions will help shape the development of the system.

There will be two sections to this experiment, and you will take approximately two hours to complete the tasks. The first section is a screening and you must pass this screening in order to participate in this experiment.

[ Administer Screening before continuing with instructions / Set Keyboard Speed]

To Live in a World of Love is to Live in a World of Peace

What you do today might have been done better yesterday

*This is a test, as life is a test*

I am the color ??      Note: These text strips were presented in the colors  
I am the color ??      Red, Green, Blue, Yellow, and Black, respectively.  
I am the color ??  
I am the color ??  
I am the color ??

Please Type Here..

[ If they do not pass the tests, thank the participant and excuse them from the evaluation.]

Remember all data is kept strictly confidential, and there is no way for the experimenter to separate your data from previously taken data after the completion of your session. The

VCR camera behind you shoulder is set to record your use of the computer and to help record your comments made during an interview at the end of the session.

Please look at the program presented on the computer, it is called The Personal Organizer. The computer will prompt you with a single instruction in a separate window. Please follow the instructions and complete the task as outlined. If you cannot figure a way to complete a task inform the instructor. Try your best to complete each and every task in a timely manner. Please verify that you have completed everything that the instructions ask for before continuing to the next instruction. It is more important that you complete each and every instruction that to go fast. Use the HELP system if you are having problems. In many cases the HELP system will be faster than randomly experimenting with the interface. Once you start a new task you will not be allowed to go back to the previous task. Remember, you are assisting in the program development process. Once you have completed the task you will need to return to the main screen and click the "DONE!" button. You return to the main screen by clicking the Main Screen button in the application. Clicking "Done!" will end the trial and start the next trial. After 8 trials you may take a break. There will be six sets of trials and you may take a break after each set of trials.

Here are some rules to follow during the experiment:

- Do not try to quit the program or switch out of the application using Multifinder,
- Try to use the Tab keys and Quick keys to speed your performance,
- Do not use the message bar, you should be able to perform all tasks without typing any commands. Use the menus, icons, quick keys, function keys and arrow keys to perform actions,
- Unless the instructions specially request a certain font or style, do not waste time making text match other text styles when editing,
- Searching and Navigation is very important to completing the instructions. If you find yourself clicking arrow keys over and over again to move around, try using the "Find" function or another use a different method to move around the quickly. Look for pop-up menus.
- Please read over the list of quick keys that is provided, this will give you an idea of some of the most common function and actions,

Are there any questions? Are you ready to begin?

[Administer General Questionnaire ]

[Start program]

[ Once the Tasks are completed Continue... Do not allow subject to use the interface while answering the questionnaire

Please answer the questions on the questionnaire provided. This will take a few minutes. Please try to consider each question carefully, your expert opinions are very valuable.

[Administer the Questionnaire ]

[Rest Break ]

**[Repeat next two sections for the last two sets of tasks]**

Please continue to use the interface by following the instructions in the window. The sets of tasks used in this interface are similar to those provided in the first test. Please try to complete each and every task in a timely manner. Note the new list of quick keys, and the new keyboard template.

[ Once the Tasks are completed Continue... Do not allow subject to use the interface while answering the questionnaire

Please answer the questions on the questionnaire provided. This will take a few minutes. Please try to consider each question carefully, as they are not the same as found in the first questionnaire. Please base you decisions on the last two sets of instructions.

[Administer the Questionnaire ] [Rest Break ]

Thank-you for participation in this study. You have completed using all three interfaces. Please take some time to complete this final questionnaire. You may use and manipulate the interface in order to answer the questions.

[Administer Final Questionnaire ]

---

# APPENDIX I: Correlation Matrix: Preference Index

TABLE I. Spearman Correlations and  $p$  values for the items in the PI

Note: The numbers correspond to the items listed in Table 21. Number 7 was the Consistent-Inconsistent Item.

	1	2	3	4	5	6	7	8	9	10
1	1.000 <.001	0.540 0.001	0.305 0.085	0.483 0.004	0.376 0.031	0.322 0.068	0.445 0.009	0.343 0.051	0.398 0.022	0.483 0.004
2	0.540 0.001	1.000 <.001	0.560 0.001	0.466 0.006	0.355 0.042	0.296 0.095	0.273 0.125	0.449 0.009	0.420 0.015	0.531 0.002
3	0.305 0.085	0.560 0.001	1.000 <.001	0.227 0.203	0.442 0.010	0.259 0.145	0.141 0.435	0.331 0.060	0.388 0.026	0.225 0.208
4	0.483 0.004	0.466 0.006	0.227 0.203	1.000 <.001	0.520 0.002	0.368 0.035	0.463 0.007	0.261 0.143	0.504 0.003	0.325 0.065
5	0.376 0.031	0.355 0.042	0.442 0.010	0.520 0.002	1.000 <.001	0.228 0.201	0.618 <.001	0.273 0.125	0.257 0.148	0.123 0.496
6	0.322 0.068	0.296 0.095	0.259 0.145	0.368 0.035	0.228 0.201	1.000 <.001	0.577 <.001	0.409 0.018	0.331 0.060	0.539 0.001
7	0.445 0.009	0.273 0.125	0.141 0.435	0.463 0.007	0.618 <.001	0.577 <.001	1.000 <.001	0.324 0.066	0.298 0.092	0.388 0.026
8	0.343 0.051	0.449 0.009	0.331 0.060	0.261 0.143	0.273 0.125	0.409 0.018	0.324 0.066	1.000 <.001	0.300 0.090	0.474 0.005
9	0.398 0.022	0.420 0.015	0.388 0.026	0.504 0.003	0.257 0.148	0.331 0.060	0.298 0.092	0.300 0.090	1.000 <.001	0.407 0.019
10	0.483 0.004	0.531 0.002	0.225 0.208	0.325 0.065	0.123 0.496	0.539 0.001	0.388 0.026	0.474 0.005	0.407 0.019	1.000 <.001
11	0.379 0.030	0.294 0.096	0.060 0.739	0.110 0.542	0.073 0.687	0.218 0.223	0.101 0.575	0.274 0.123	0.335 0.057	0.180 0.317

TABLE I (cont.). Spearman Correlations and *p* values for the Preference Index Items

	1	2	3	4	5	6	7	8	9	10
12	0.263 0.139	0.088 0.626	0.323 0.066	0.318 0.071	0.066 0.716	0.186 0.299	0.001 0.995	0.132 0.465	0.517 0.002	0.028 0.879
13	0.503 0.003	0.401 0.021	0.334 0.057	0.243 0.174	0.188 0.295	0.241 0.177	0.301 0.089	0.295 0.096	0.504 0.003	0.316 0.074
14	0.125 0.490	0.078 0.668	0.300 0.090	0.099 0.583	0.194 0.280	0.120 0.507	0.206 0.250	0.301 0.089	0.011 0.953	0.020 0.911
15	0.287 0.106	0.170 0.345	0.068 0.707	0.069 0.702	0.139 0.441	0.353 0.044	0.166 0.356	0.160 0.374	0.024 0.894	0.312 0.077
16	0.003 0.987	0.160 0.374	0.242 0.175	0.019 0.916	0.176 0.328	0.117 0.518	0.021 0.909	0.060 0.740	0.303 0.086	0.015 0.935
17	0.251 0.159	0.463 0.007	0.387 0.026	0.528 0.002	0.495 0.003	0.466 0.006	0.353 0.044	0.310 0.079	0.374 0.032	0.436 0.011
18	0.412 0.017	0.607 <.001	0.429 0.013	0.347 0.048	0.380 0.029	0.530 0.002	0.474 0.005	0.516 0.002	0.362 0.039	0.537 0.001
19	0.604 <.001	0.399 0.022	0.384 0.027	0.522 0.002	0.450 0.009	0.466 0.006	0.463 0.007	0.468 0.006	0.396 0.022	0.407 0.019
20	0.679 <.001	0.658 <.001	0.373 0.033	0.413 0.017	0.405 0.020	0.308 0.081	0.482 0.005	0.455 0.008	0.495 0.003	0.567 0.001
21	0.667 <.001	0.504 0.003	0.388 0.026	0.450 0.009	0.287 0.105	0.328 0.062	0.344 0.050	0.463 0.007	0.397 0.022	0.403 0.020
22	0.359 0.040	0.536 0.001	0.352 0.045	0.407 0.019	0.309 0.080	0.420 0.015	0.340 0.053	0.577 <.001	0.358 0.041	0.461 0.007
23	0.576 0.001	0.570 0.001	0.388 0.026	0.630 <.001	0.458 0.007	0.441 0.010	0.525 0.002	0.557 0.001	0.426 0.014	0.492 0.004
24	0.474 0.005	0.603 <.001	0.480 0.005	0.573 0.001	0.513 0.002	0.492 0.004	0.429 0.013	0.629 <.001	0.430 0.013	0.509 0.003
25	0.582 <.001	0.630 <.001	0.627 <.001	0.478 0.005	0.620 <.001	0.468 0.006	0.458 0.007	0.643 <.001	0.399 0.022	0.344 0.050
26	0.635 <.001	0.544 0.001	0.325 0.065	0.255 0.153	0.240 0.179	0.211 0.239	0.296 0.095	0.227 0.204	0.382 0.028	0.335 0.057
27	0.527 0.002	0.425 0.014	0.275 0.122	0.291 0.101	0.263 0.140	0.159 0.378	0.054 0.767	0.175 0.330	0.345 0.049	0.163 0.366
28	0.470 0.006	0.343 0.051	0.465 0.006	0.464 0.007	0.336 0.056	0.441 0.010	0.203 0.258	0.258 0.148	0.373 0.032	0.331 0.060
29	0.655 <.001	0.656 <.001	0.390 0.025	0.624 <.001	0.506 0.003	0.418 0.016	0.463 0.007	0.602 <.001	0.517 0.002	0.536 0.001

TABLE I (cont.). Spearman Correlations and *p* values for the Preference Index Items

	11	12	13	14	15	16	17	18	19	20
1	0.379 0.030	0.263 0.139	0.503 0.003	0.125 0.490	0.287 0.106	0.003 0.987	0.251 0.159	0.412 0.017	0.604 <.001	0.679 <.001
2	0.294 0.096	0.088 0.626	0.401 0.021	0.078 0.668	0.170 0.345	0.160 0.374	0.463 0.007	0.607 <.001	0.399 0.022	0.658 <.001
3	0.060 0.739	0.323 0.066	0.334 0.057	0.300 0.090	0.068 0.707	0.242 0.175	0.387 0.026	0.429 0.013	0.384 0.027	0.373 0.033
4	0.110 0.542	0.318 0.071	0.243 0.174	0.099 0.583	0.069 0.702	0.019 0.916	0.528 0.002	0.347 0.048	0.522 0.002	0.413 0.017
5	0.073 0.687	0.066 0.716	0.188 0.295	0.194 0.280	0.139 0.441	0.176 0.328	0.495 0.003	0.380 0.029	0.450 0.009	0.405 0.020
6	0.218 0.223	0.186 0.299	0.241 0.177	0.120 0.507	0.353 0.044	0.117 0.518	0.466 0.006	0.530 0.002	0.466 0.006	0.308 0.081
7	0.101 0.575	0.001 0.995	0.301 0.089	0.206 0.250	0.166 0.356	0.021 0.909	0.353 0.044	0.474 0.005	0.463 0.007	0.482 0.005
8	0.274 0.123	0.132 0.465	0.295 0.096	0.301 0.089	0.160 0.374	0.060 0.740	0.310 0.079	0.516 0.002	0.468 0.006	0.455 0.008
9	0.335 0.057	0.517 0.002	0.504 0.003	0.011 0.953	0.024 0.894	0.303 0.086	0.374 0.032	0.362 0.039	0.396 0.022	0.495 0.003
10	0.180 0.317	0.028 0.879	0.316 0.074	0.020 0.911	0.312 0.077	0.015 0.935	0.436 0.011	0.537 0.001	0.407 0.019	0.567 0.001
11	1.000 <.001	0.282 0.112	0.326 0.064	0.120 0.508	0.052 0.774	0.168 0.351	0.173 0.336	0.168 0.349	0.269 0.130	0.188 0.294
12	0.282 0.112	1.000 <.001	0.370 0.034	0.315 0.074	0.136 0.451	0.291 0.101	0.146 0.418	0.207 0.247	0.378 0.030	0.193 0.282
13	0.326 0.064	0.370 0.034	1.000 <.001	0.364 0.037	0.455 0.008	0.208 0.246	0.143 0.427	0.484 0.004	0.486 0.004	0.649 <.001
14	0.120 0.508	0.315 0.074	0.364 0.037	1.000 <.001	0.449 0.009	0.017 0.927	0.125 0.489	0.156 0.387	0.172 0.340	0.224 0.209
15	0.052 0.774	0.136 0.451	0.455 0.008	0.449 0.009	1.000 <.001	0.069 0.705	0.235 0.189	0.280 0.114	0.166 0.357	0.344 0.050
16	0.168 0.351	0.291 0.101	0.208 0.246	0.017 0.927	0.069 0.705	1.000 <.001	0.011 0.951	0.082 0.649	0.050 0.783	0.077 0.672
17	0.173 0.336	0.146 0.418	0.143 0.427	0.125 0.489	0.235 0.189	0.011 0.951	1.000 <.001	0.428 0.013	0.348 0.047	0.394 0.023
18	0.168 0.349	0.207 0.247	0.484 0.004	0.156 0.387	0.280 0.114	0.082 0.649	0.428 0.013	1.000 <.001	0.739 <.001	0.669 <.001
19	0.269 0.130	0.378 0.030	0.486 0.004	0.172 0.340	0.166 0.357	0.050 0.783	0.348 0.047	0.739 <.001	1.000 <.001	0.661 <.001

TABLE I (cont.). Spearman Correlations and *p* values for the Preference Index Items

	11	12	13	14	15	16	17	18	19	20
20	0.188	0.193	0.649	0.224	0.344	0.077	0.394	0.669	0.661	1.000
	0.294	0.282	<.001	0.209	0.050	0.672	0.023	<.001	<.001	<.001
21	0.288	0.392	0.587	0.193	0.276	0.099	0.279	0.367	0.539	0.730
	0.105	0.024	<.001	0.283	0.120	0.583	0.117	0.036	0.001	<.001
22	0.236	0.144	0.349	0.035	0.083	0.164	0.393	0.767	0.734	0.605
	0.187	0.425	0.046	0.847	0.645	0.361	0.024	<.001	<.001	<.001
23	0.123	0.226	0.440	0.226	0.256	0.137	0.570	0.638	0.671	0.651
	0.495	0.206	0.011	0.206	0.151	0.446	0.001	<.001	<.001	<.001
24	0.156	0.123	0.345	0.132	0.099	0.051	0.521	0.770	0.698	0.629
	0.385	0.494	0.049	0.465	0.583	0.777	0.002	<.001	<.001	<.001
25	0.240	0.296	0.479	0.303	0.121	0.059	0.425	0.765	0.685	0.623
	0.179	0.094	0.005	0.087	0.502	0.743	0.014	<.001	<.001	<.001
26	0.391	0.136	0.505	0.083	0.399	0.158	0.256	0.293	0.238	0.489
	0.024	0.449	0.003	0.645	0.021	0.379	0.150	0.098	0.182	0.004
27	0.429	0.417	0.452	0.172	0.336	0.300	0.204	0.331	0.278	0.401
	0.013	0.016	0.008	0.340	0.056	0.090	0.254	0.060	0.117	0.021
28	0.455	0.370	0.410	0.100	0.078	0.176	0.314	0.522	0.627	0.350
	0.008	0.034	0.018	0.579	0.665	0.327	0.075	0.002	<.001	0.046
29	0.520	0.254	0.485	0.112	0.141	0.042	0.388	0.698	0.775	0.629
	0.002	0.155	0.004	0.534	0.434	0.818	0.026	<.001	<.001	<.001

TABLE I (cont.). Spearman Correlations and *p* values for the Preference Index Items

	21	22	23	24	25	26	27	28	29
1	0.667 <.001	0.359 0.040	0.576 0.001	0.474 0.005	0.582 <.001	0.635 <.001	0.527 0.002	0.470 0.006	0.655 <.001
2	0.504 0.003	0.536 0.001	0.570 0.001	0.603 <.001	0.630 <.001	0.544 0.001	0.425 0.014	0.343 0.051	0.656 <.001
3	0.388 0.026	0.352 0.045	0.388 0.026	0.480 0.005	0.627 <.001	0.325 0.065	0.275 0.122	0.465 0.006	0.390 0.025
4	0.450 0.009	0.407 0.019	0.630 <.001	0.573 0.001	0.478 0.005	0.255 0.153	0.291 0.101	0.464 0.007	0.624 <.001
5	0.287 0.105	0.309 0.080	0.458 0.007	0.513 0.002	0.620 <.001	0.240 0.179	0.263 0.140	0.336 0.056	0.506 0.003
6	0.328 0.062	0.420 0.015	0.441 0.010	0.492 0.004	0.468 0.006	0.211 0.239	0.159 0.378	0.441 0.010	0.418 0.016
7	0.344 0.050	0.340 0.053	0.525 0.002	0.429 0.013	0.458 0.007	0.296 0.095	0.054 0.767	0.203 0.258	0.463 0.007
8	0.463 0.007	0.577 <.001	0.557 0.001	0.629 <.001	0.643 <.001	0.227 0.204	0.175 0.330	0.258 0.148	0.602 <.001
9	0.397 0.022	0.358 0.041	0.426 0.014	0.430 0.013	0.399 0.022	0.382 0.028	0.345 0.049	0.373 0.032	0.517 0.002
10	0.403 0.020	0.461 0.007	0.492 0.004	0.509 0.003	0.344 0.050	0.335 0.057	0.163 0.366	0.331 0.060	0.536 0.001
11	0.288 0.105	0.236 0.187	0.123 0.495	0.156 0.385	0.240 0.179	0.391 0.024	0.429 0.013	0.455 0.008	0.520 0.002
12	0.392 0.024	0.144 0.425	0.226 0.206	0.123 0.494	0.296 0.094	0.136 0.449	0.417 0.016	0.370 0.034	0.254 0.155
13	0.587 <.001	0.349 0.046	0.440 0.011	0.345 0.049	0.479 0.005	0.505 0.003	0.452 0.008	0.410 0.018	0.485 0.004
14	0.193 0.283	0.035 0.847	0.226 0.206	0.132 0.465	0.303 0.087	0.083 0.645	0.172 0.340	0.100 0.579	0.112 0.534
15	0.276 0.120	0.083 0.645	0.256 0.151	0.099 0.583	0.121 0.502	0.399 0.021	0.336 0.056	0.078 0.665	0.141 0.434
16	0.099 0.583	0.164 0.361	0.137 0.446	0.051 0.777	0.059 0.743	0.158 0.379	0.300 0.090	0.176 0.327	0.042 0.818
17	0.279 0.117	0.393 0.024	0.570 0.001	0.521 0.002	0.425 0.014	0.256 0.150	0.204 0.254	0.314 0.075	0.388 0.026
18	0.367 0.036	0.767 <.001	0.638 <.001	0.770 <.001	0.765 <.001	0.293 0.098	0.331 0.060	0.522 0.002	0.698 <.001
19	0.539 0.001	0.734 <.001	0.671 <.001	0.698 <.001	0.685 <.001	0.238 0.182	0.278 0.117	0.627 <.001	0.775 <.001

TABLE I (cont.). Spearman Correlations and *p* values for the Preference Index Items

	21	22	23	24	25	26	27	28	29
20	0.730 <.001	0.605 <.001	0.651 <.001	0.629 <.001	0.623 <.001	0.489 0.004	0.401 0.021	0.350 0.046	0.629 <.001
21	1.000 <.001	0.399 0.022	0.456 0.008	0.457 0.008	0.502 0.003	0.572 0.001	0.354 0.043	0.388 0.026	0.505 0.003
22	0.399 0.022	1.000 <.001	0.722 <.001	0.809 <.001	0.685 <.001	0.192 0.285	0.383 0.028	0.542 0.001	0.755 <.001
23	0.456 0.008	0.722 <.001	1.000 <.001	0.653 <.001	0.690 <.001	0.359 0.040	0.385 0.027	0.472 0.006	0.748 <.001
24	0.457 0.008	0.809 <.001	0.653 <.001	1.000 <.001	0.833 <.001	0.321 0.069	0.396 0.023	0.587 <.001	0.727 <.001
25	0.502 0.003	0.685 <.001	0.690 <.001	0.833 <.001	1.000 <.001	0.398 0.022	0.487 0.004	0.597 <.001	0.723 <.001
26	0.572 0.001	0.192 0.285	0.359 0.040	0.321 0.069	0.398 0.022	1.000 <.001	0.505 0.003	0.368 0.035	0.420 0.015
27	0.354 0.043	0.383 0.028	0.385 0.027	0.396 0.023	0.487 0.004	0.505 0.003	1.000 <.001	0.488 0.004	0.534 0.001
28	0.388 0.026	0.542 0.001	0.472 0.006	0.587 <.001	0.597 <.001	0.368 0.035	0.488 0.004	1.000 <.001	0.676 <.001
29	0.505 0.003	0.755 <.001	0.748 <.001	0.727 <.001	0.723 <.001	0.420 0.015	0.534 0.001	0.676 <.001	1.000 <.001

---

## APPENDIX J: Sample Screen Displays from the Personal Organizer

The three versions of The Personal Organizer contain a variety of icons, screen layouts, menu structures, virtual devices, and other attributes. It is difficult to understand the differences without using pictures. Many consistencies and inconsistencies discussed in Appendix C can be understood by looking through this Appendix. Some inconsistencies, like auditory and visual feedback cannot be displayed in these static displays. The screens are printed to correspond precisely with the look of the display. The “jaggies” in the fonts and icons in these screen shots accurately depict the look of the actual screens.

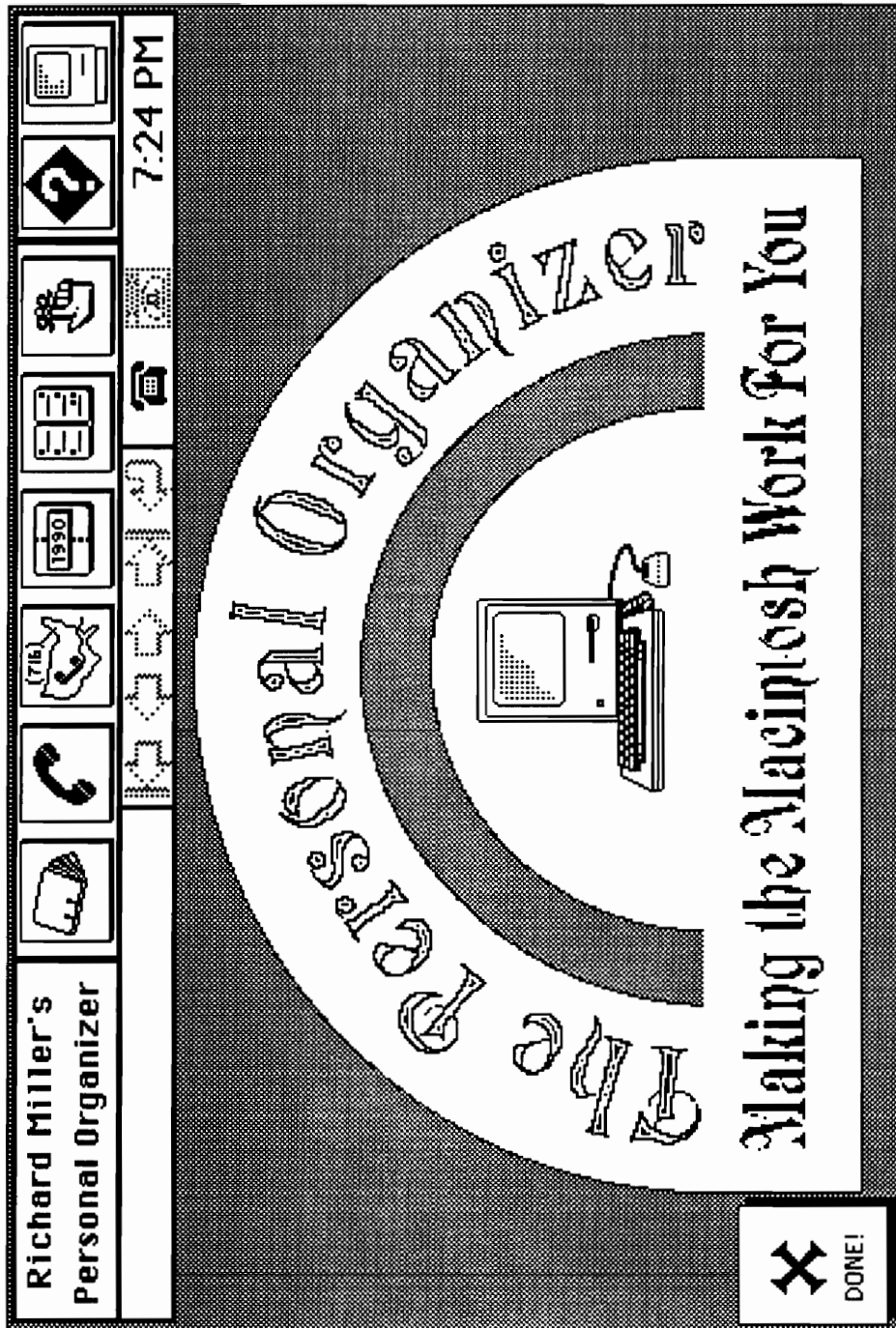


Figure J-1. Consistent version- Main Screen function.

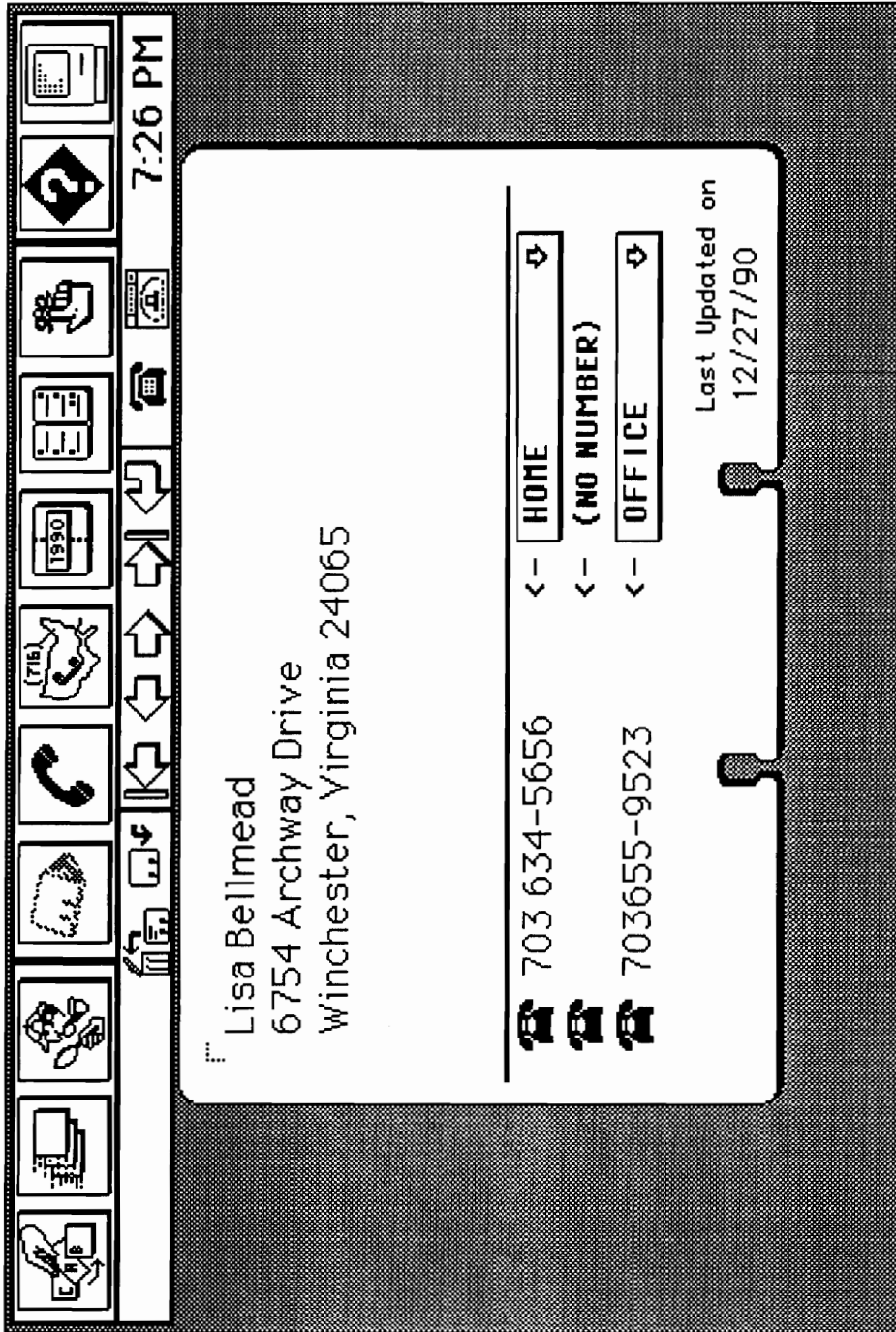


Figure J-2. Consistent version- Address Rolodex function.

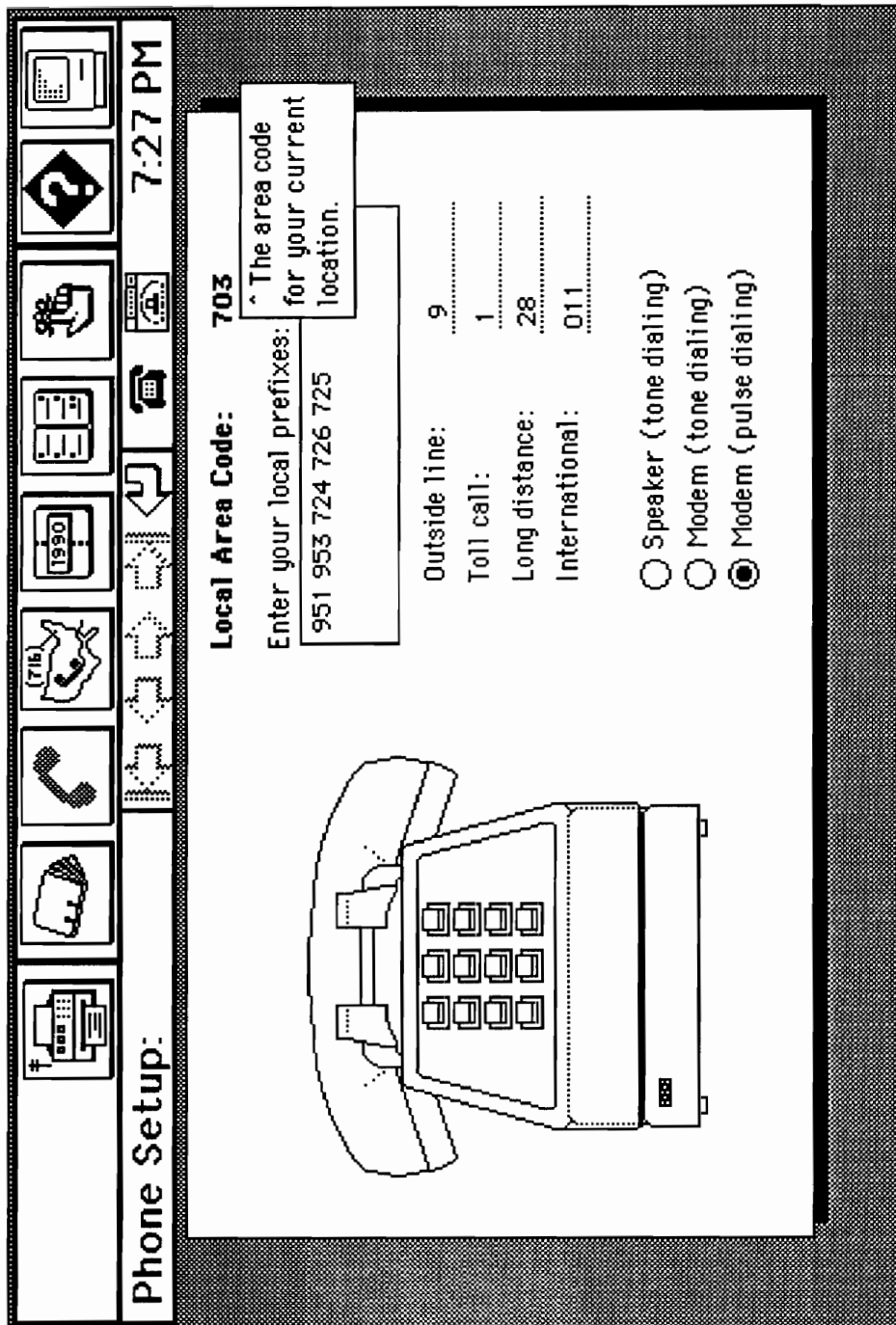


Figure J-3. Consistent version- Phone Setup function.

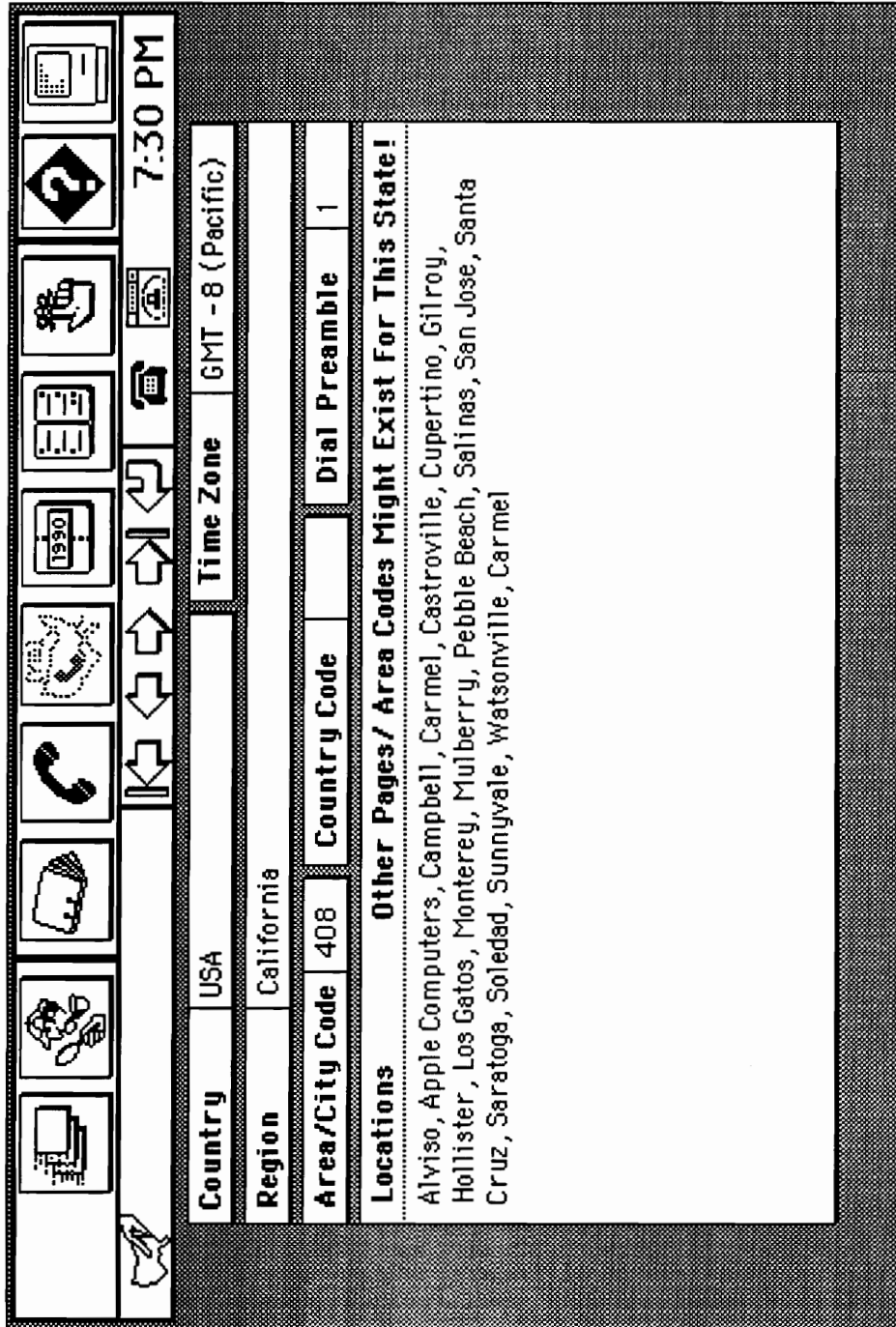


Figure J-4. Consistent version- Area Code function.

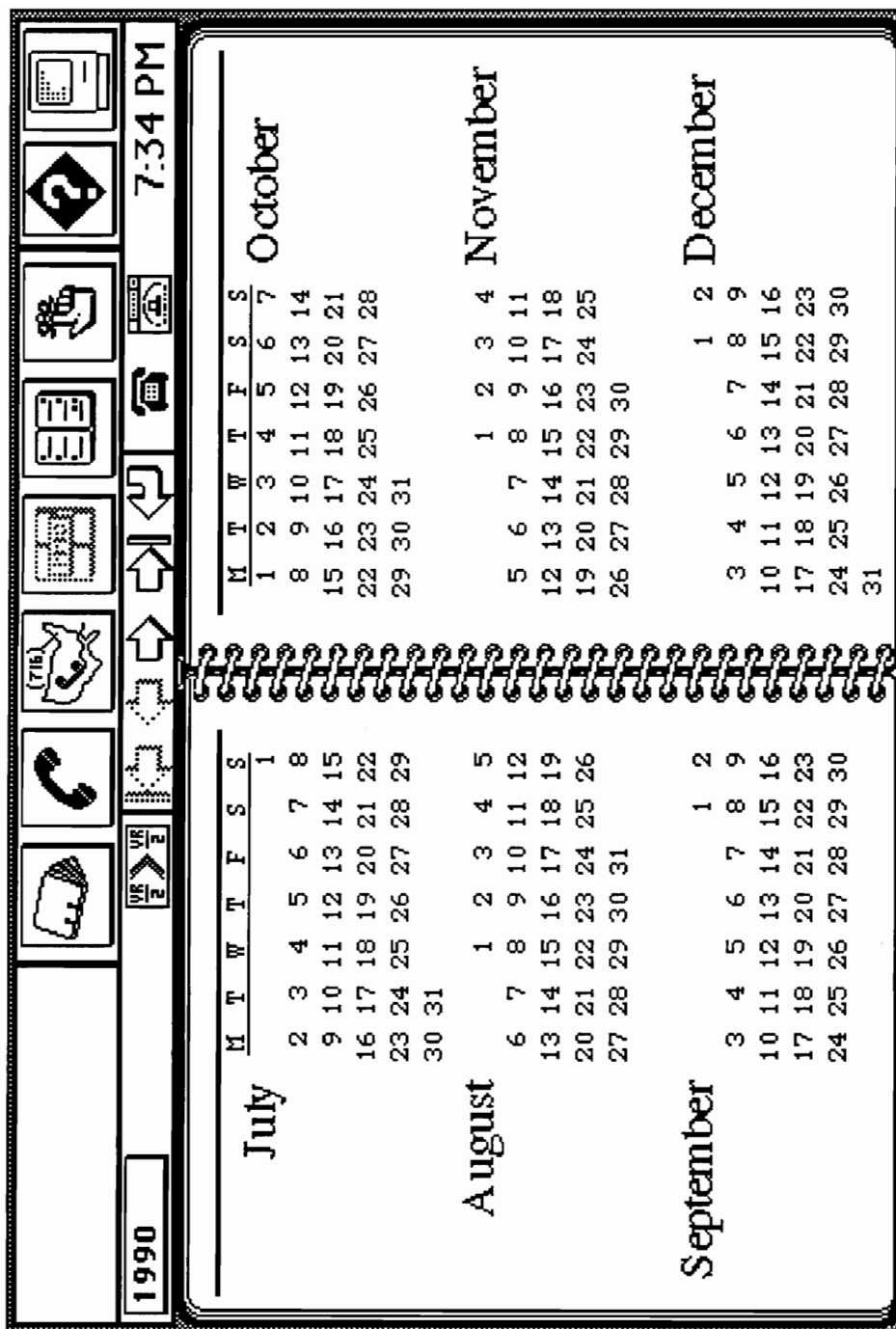


Figure J-5. Consistent version- Six-month (Yearly) Calendar.

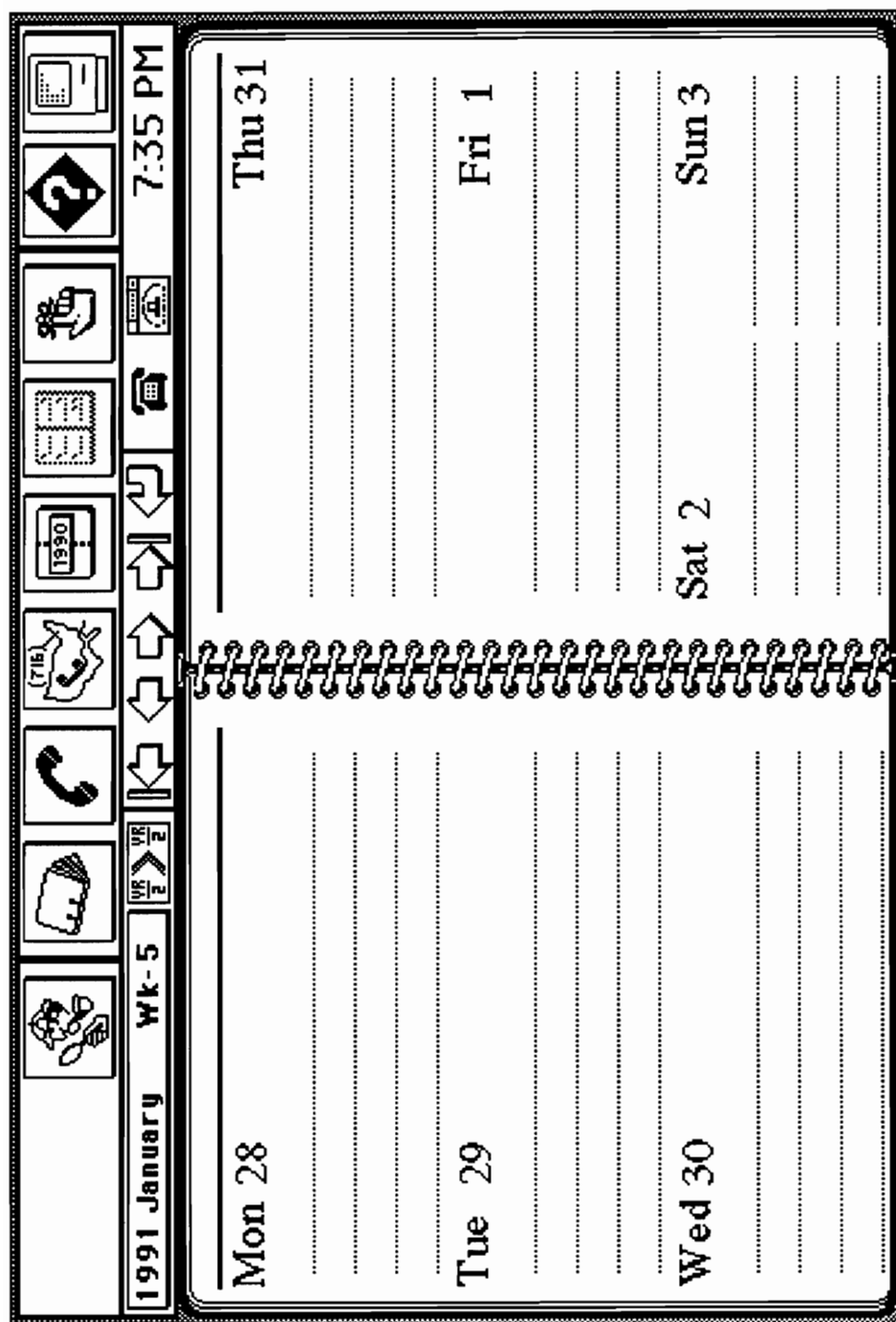


Figure J-6. Consistent version- Weekly Planner function.

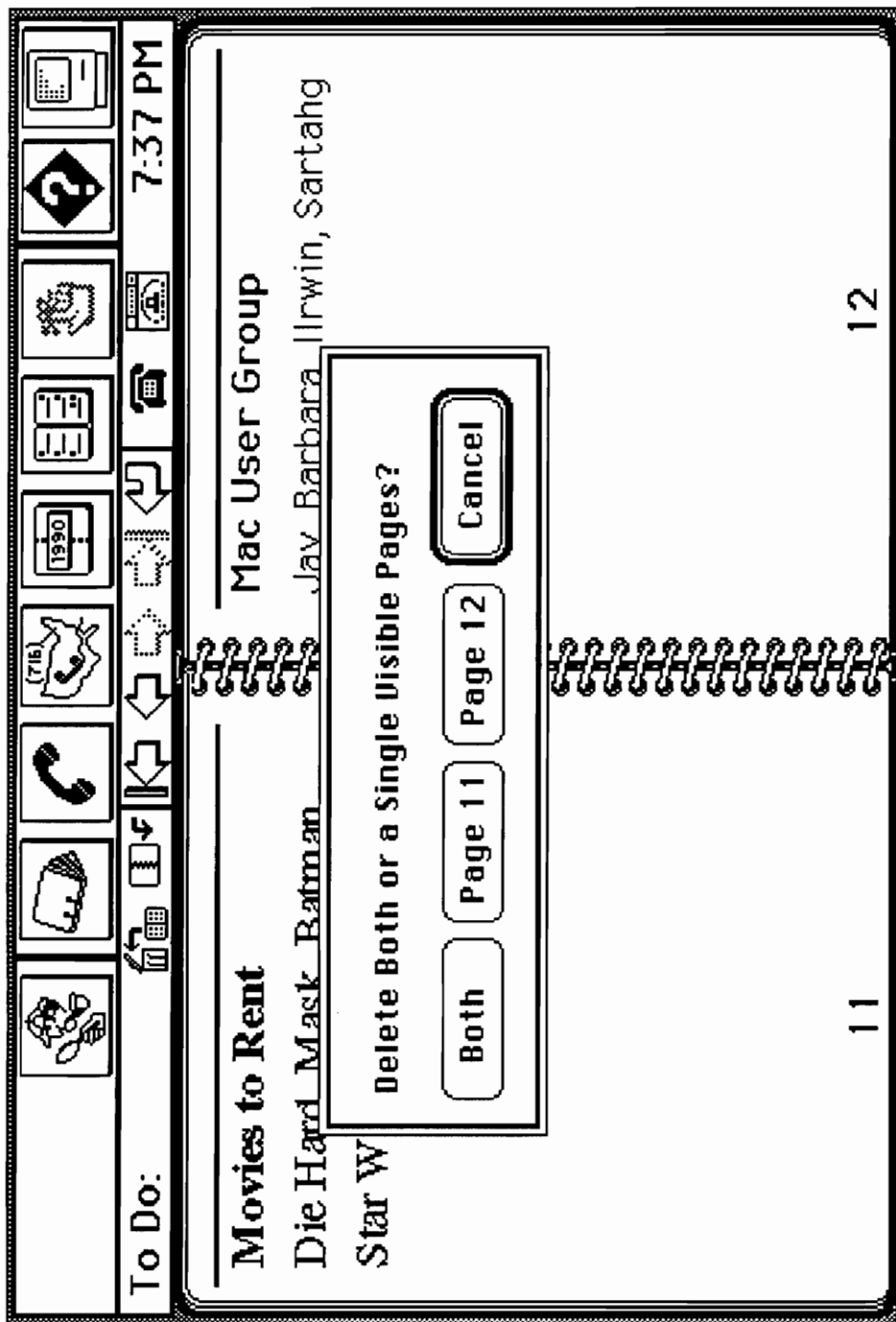


Figure J-7. Consistent version- To Do list function (with delete box open).



Figure J-8. Inconsistent version A - Main Screen function.

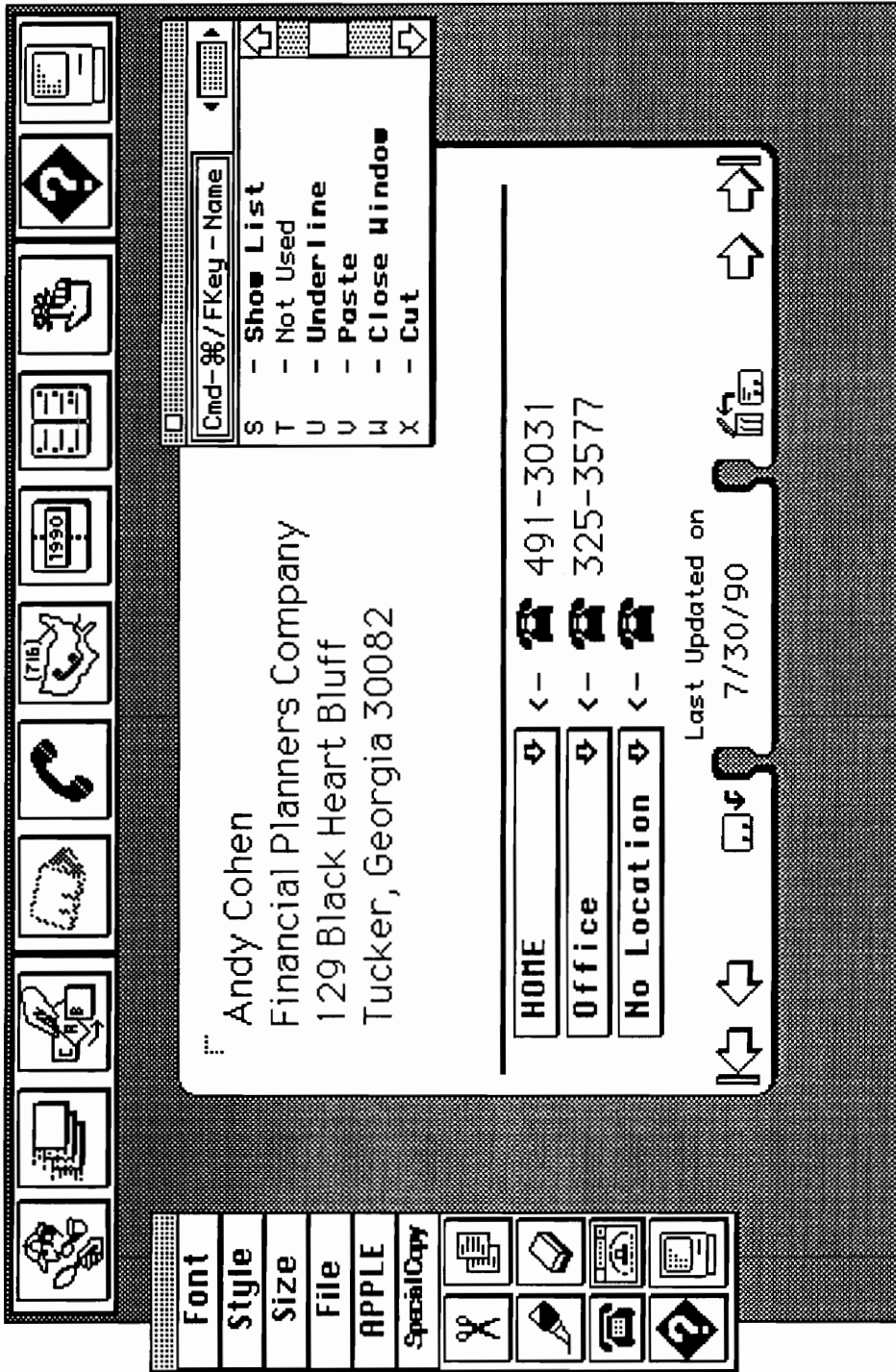


Figure J-9. Inconsistent version A- Address RoleDEX function.

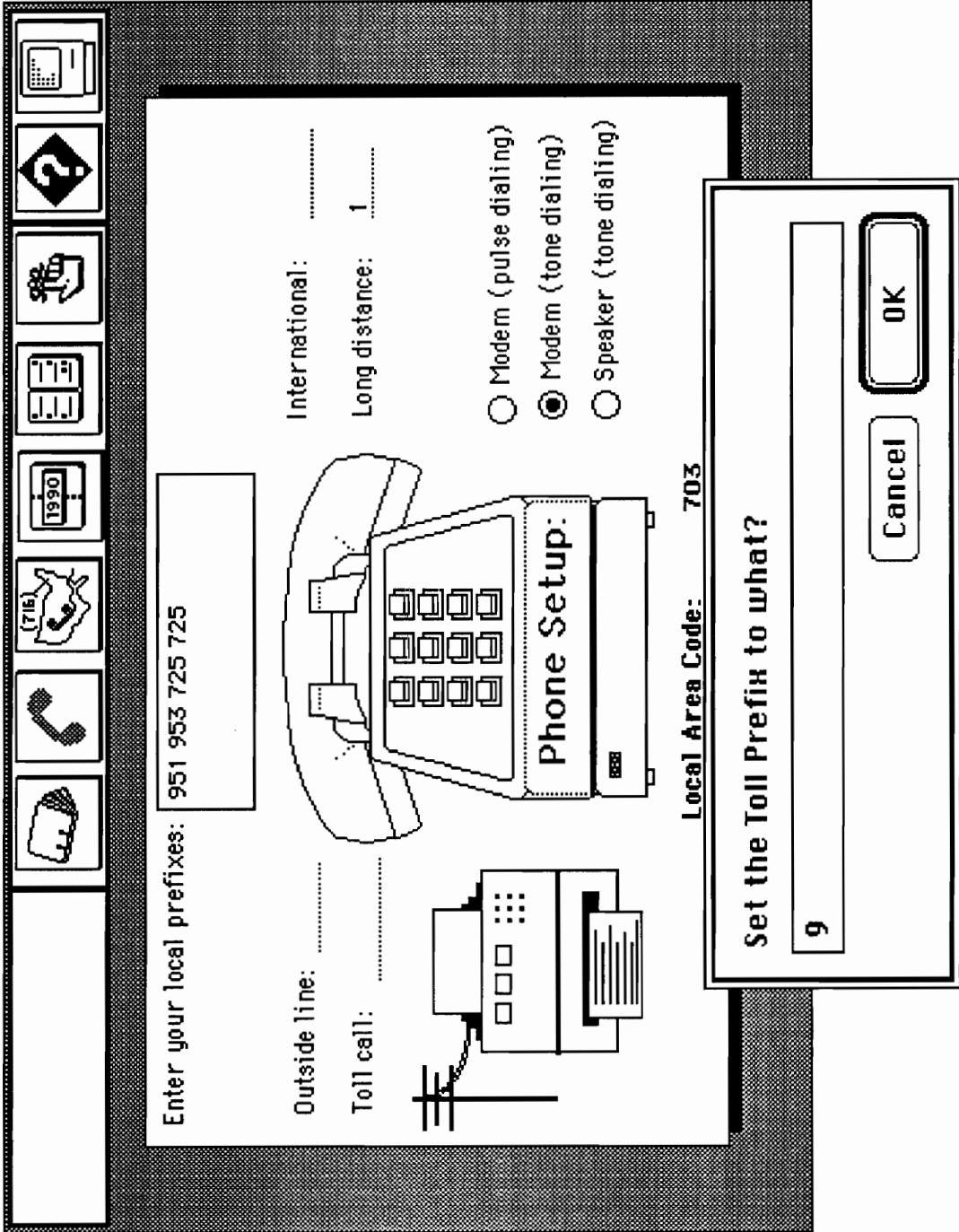


Figure J-10. Inconsistent version A- Phone Setup function.

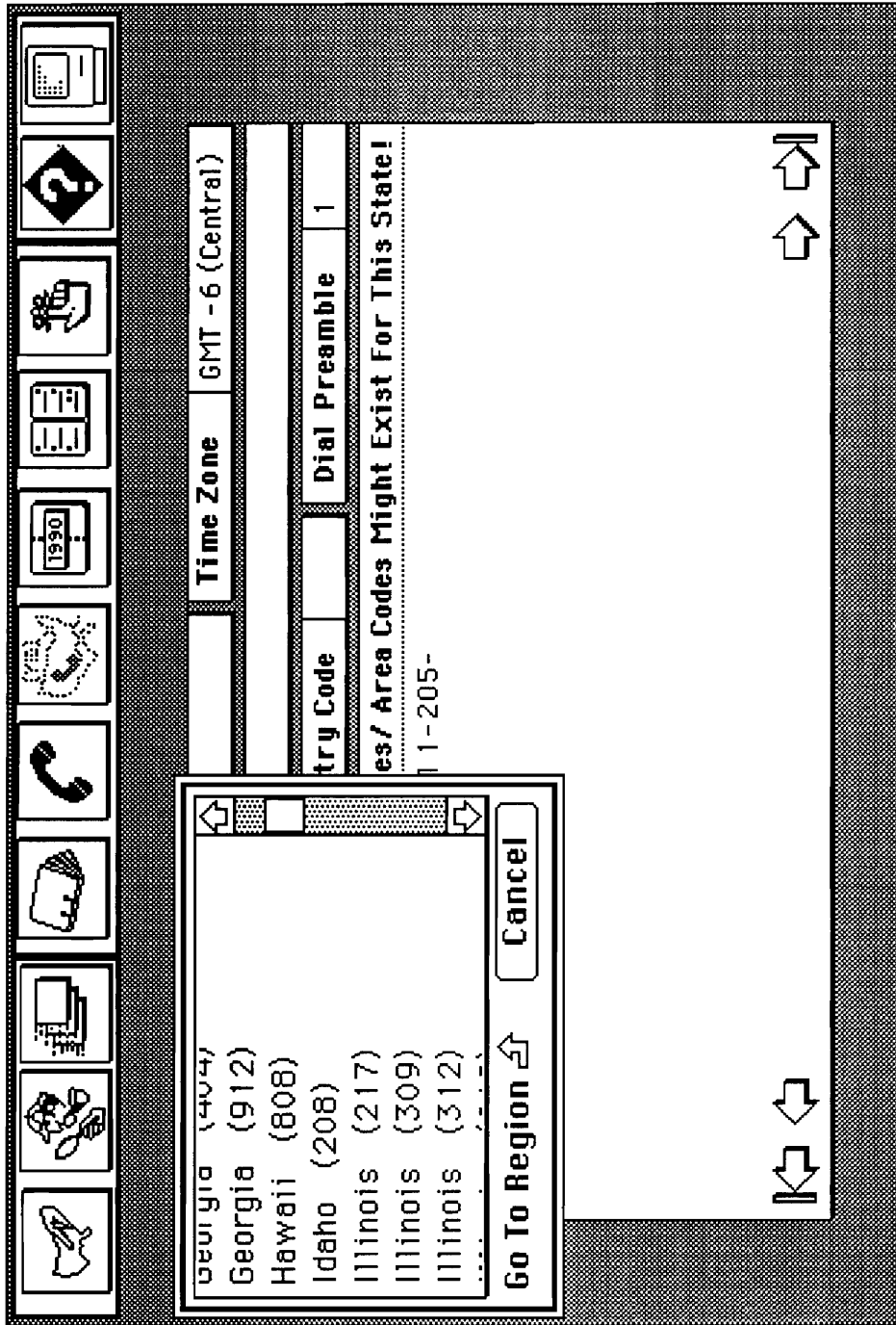


Figure J-11. Inconsistent version A- Area Code function.

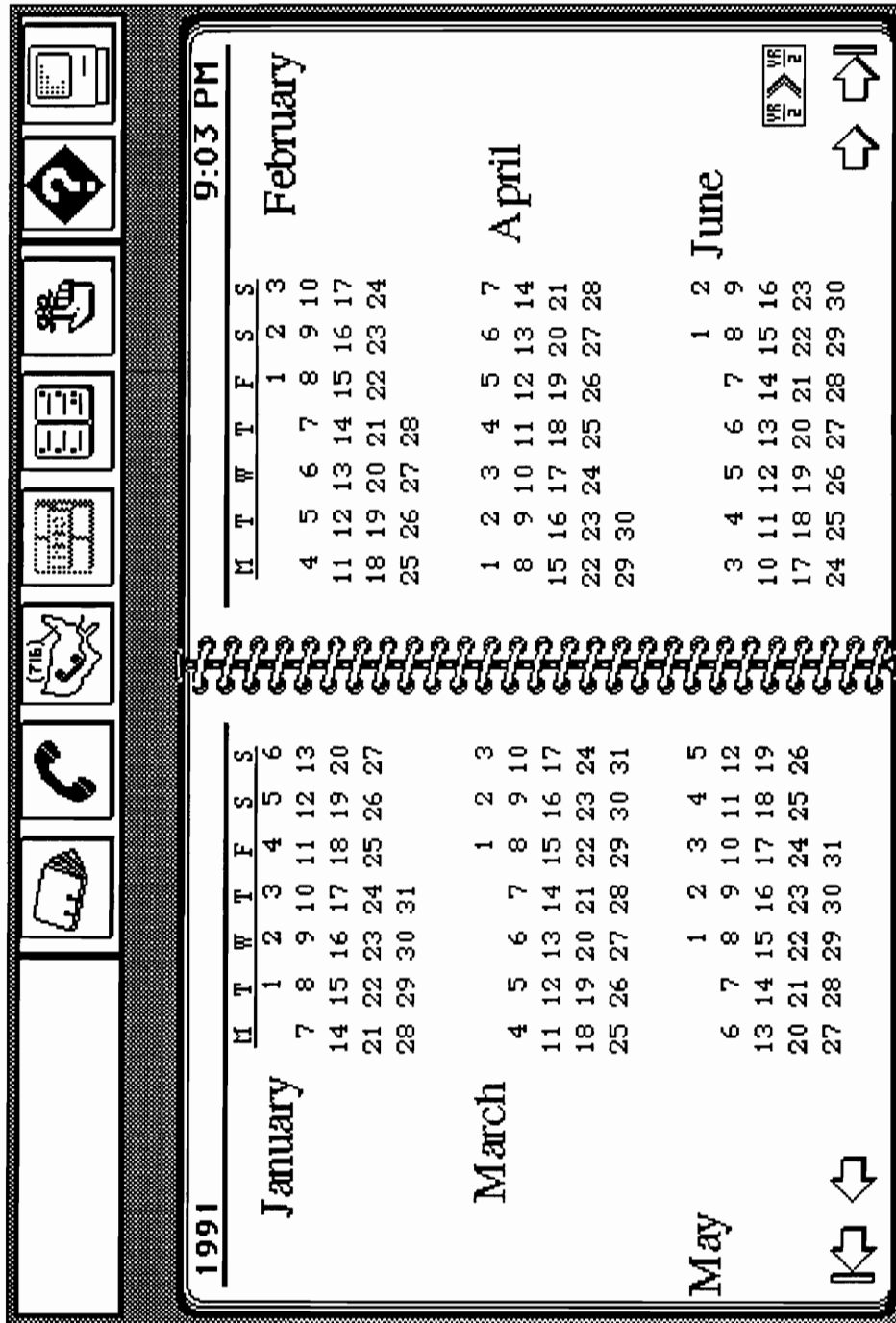


Figure J-12. Inconsistent version A- Six-month (yearly) Calendar.



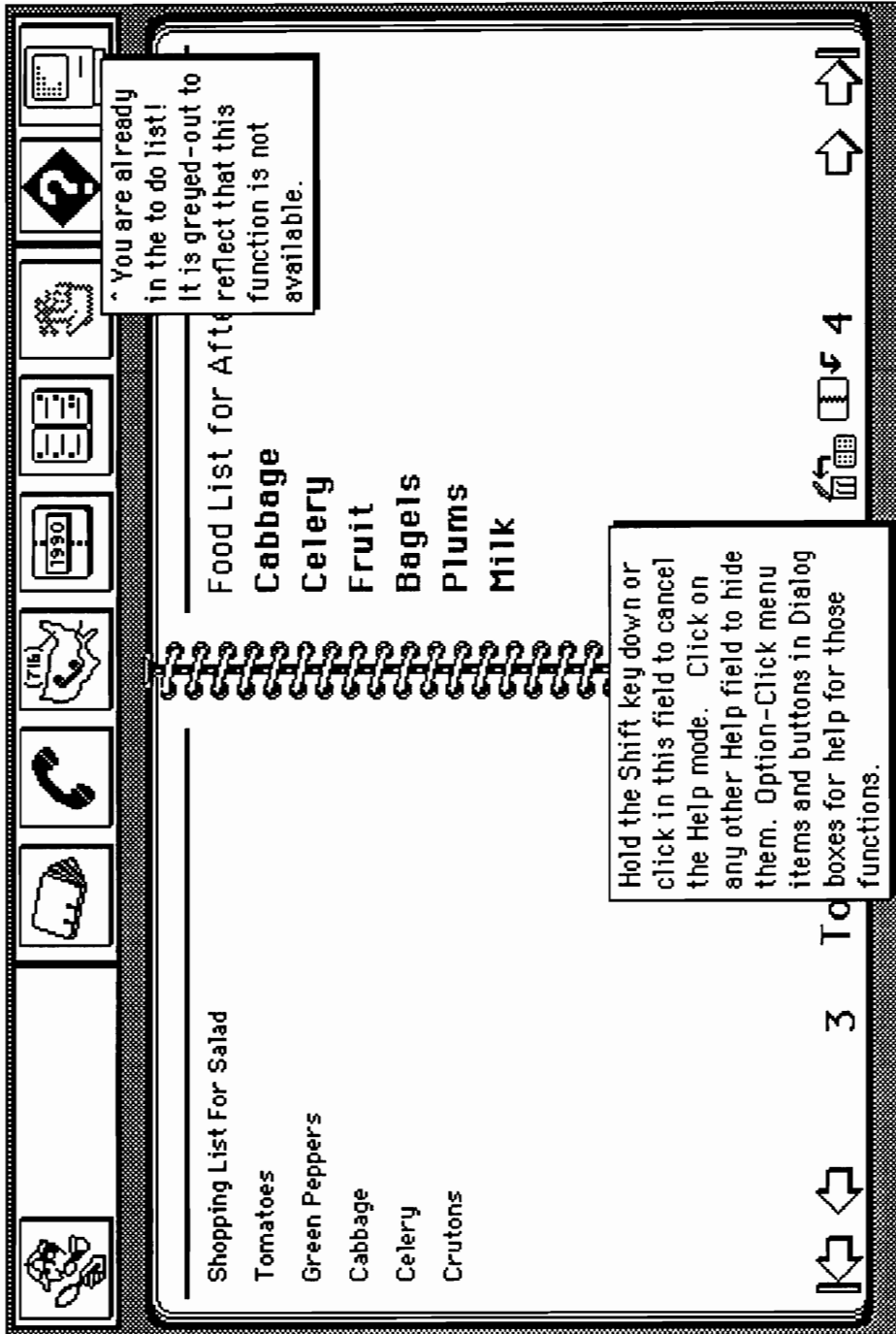


Figure J-14. Inconsistent version A- To Do list function (with delete box open).

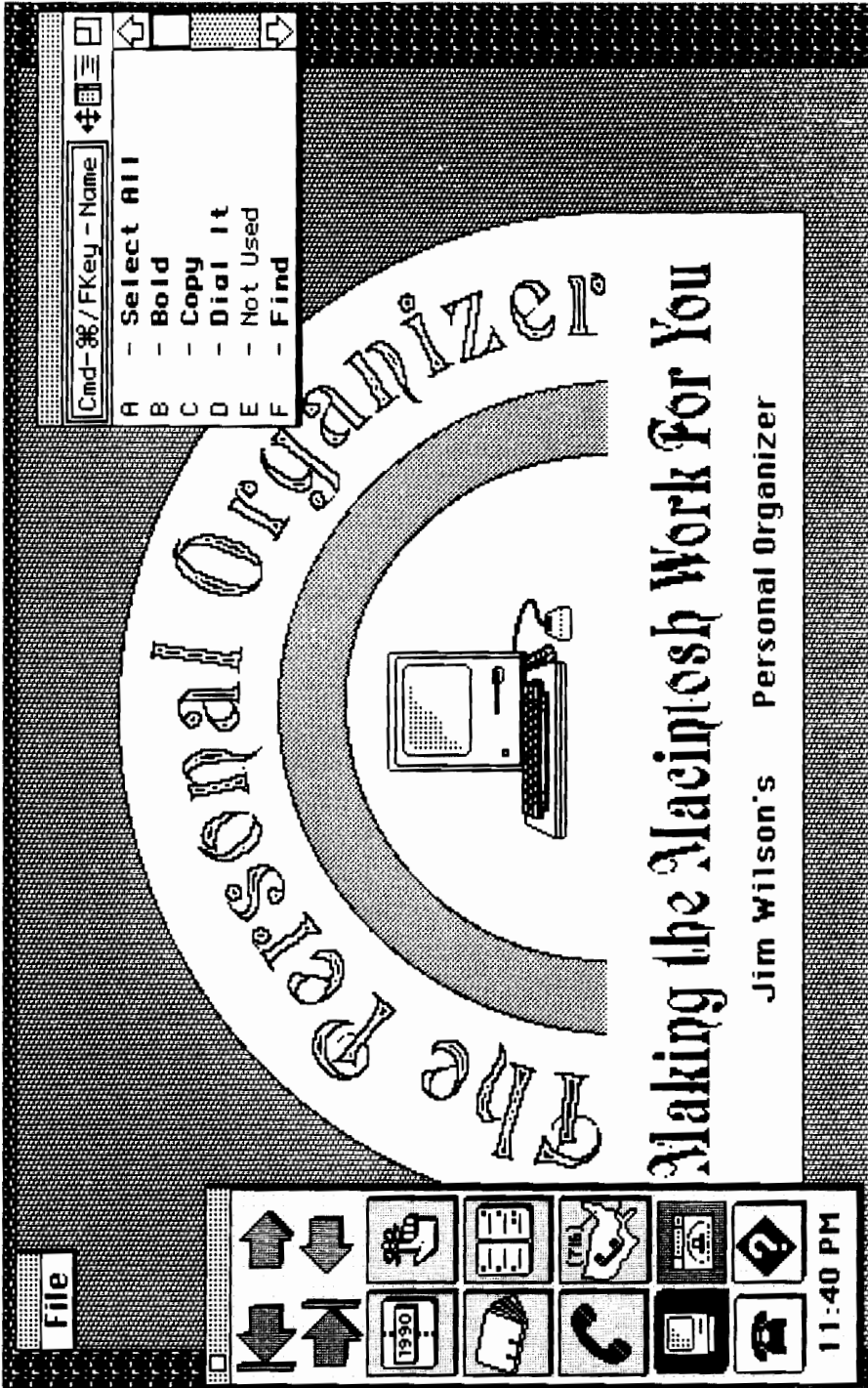


Figure J-15. Inconsistent version B- Main Screen function.

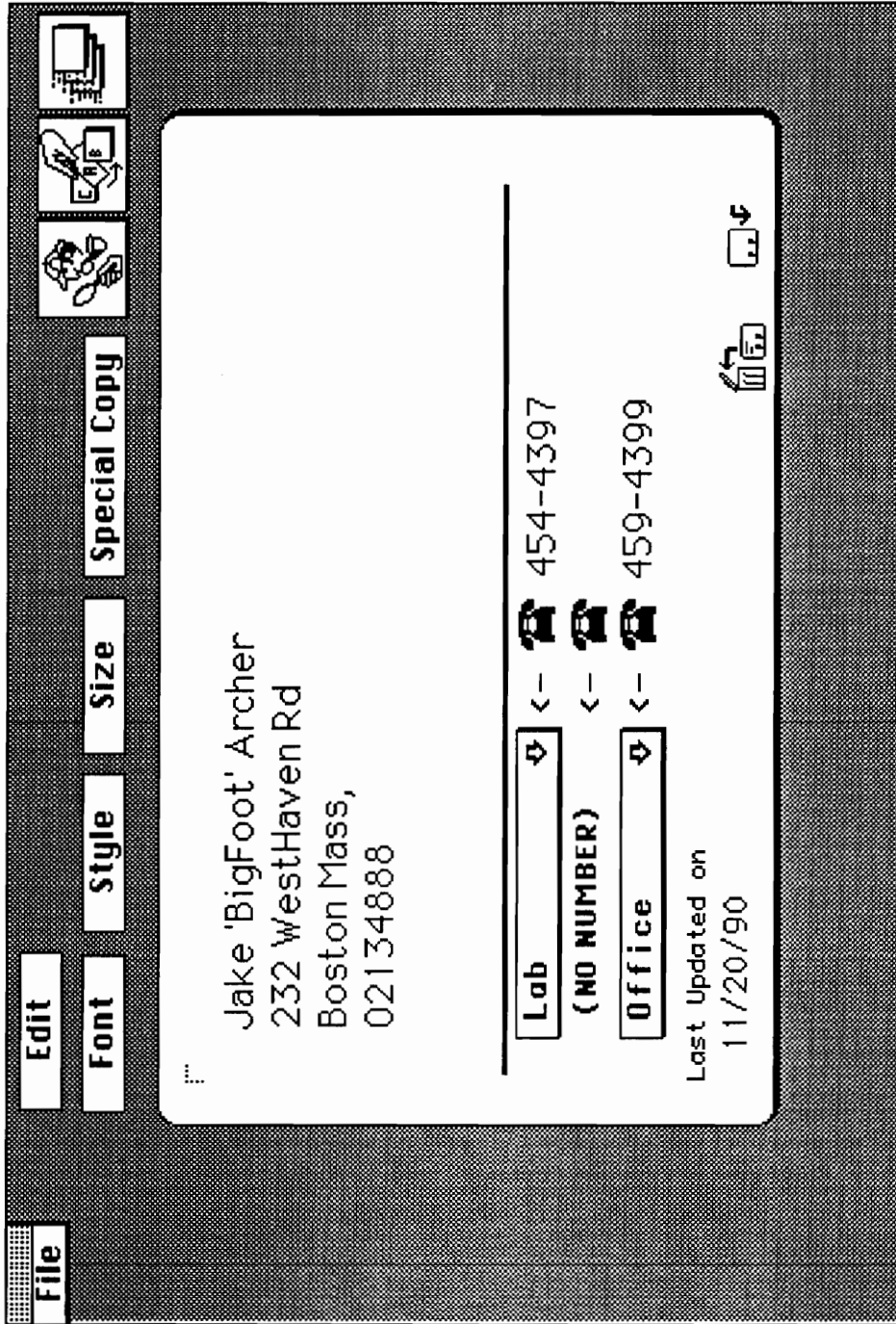


Figure J-16. Inconsistent version B- Address RoleDEX function.

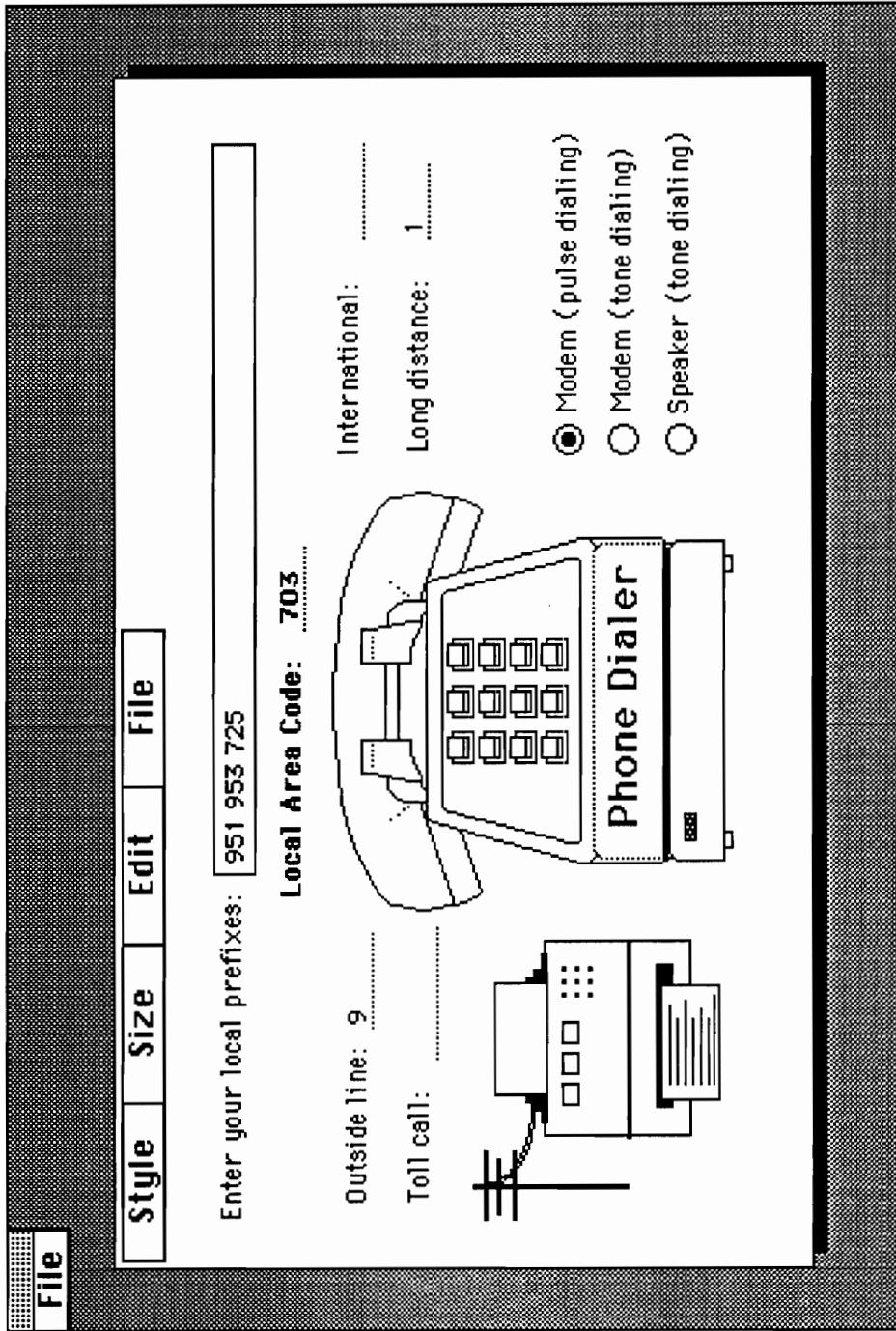


Figure J-17. Inconsistent version B- Phone Setup function.

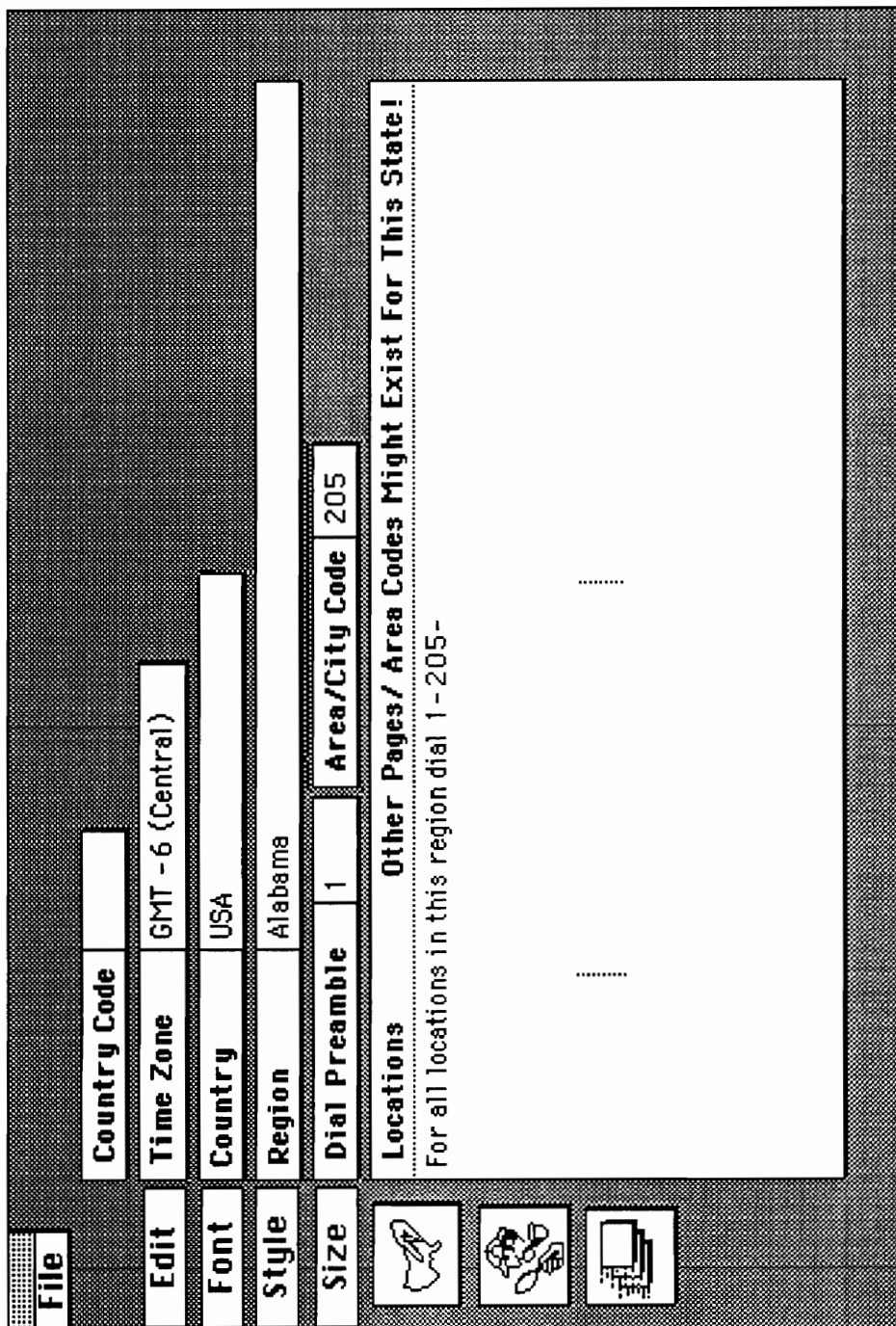


Figure J-18. Inconsistent version B- Area Code function.

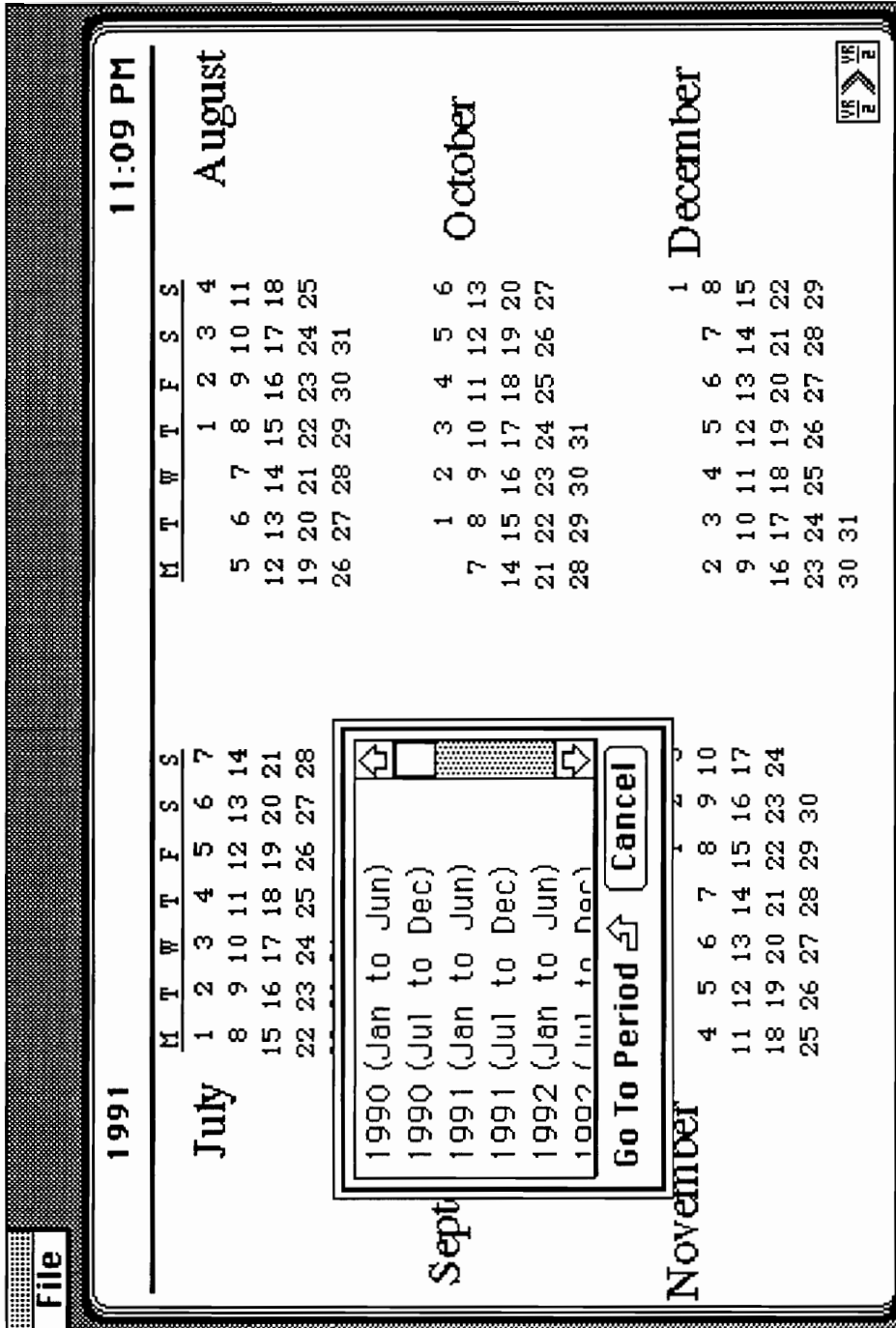


Figure J-19. Inconsistent version B- Six-month (Yearly) Calendar

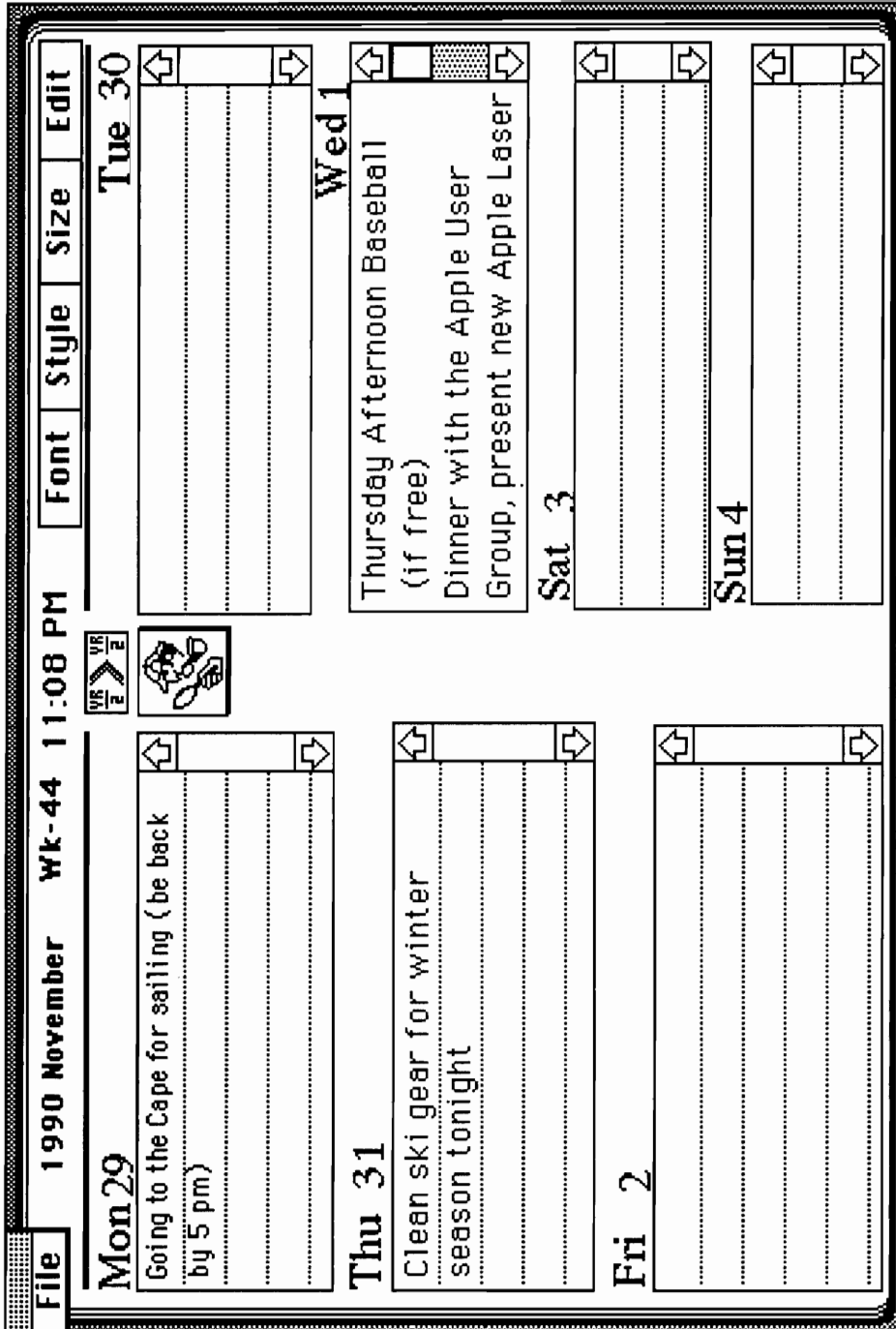


Figure J-20. Inconsistent version B- Weekly Planner.

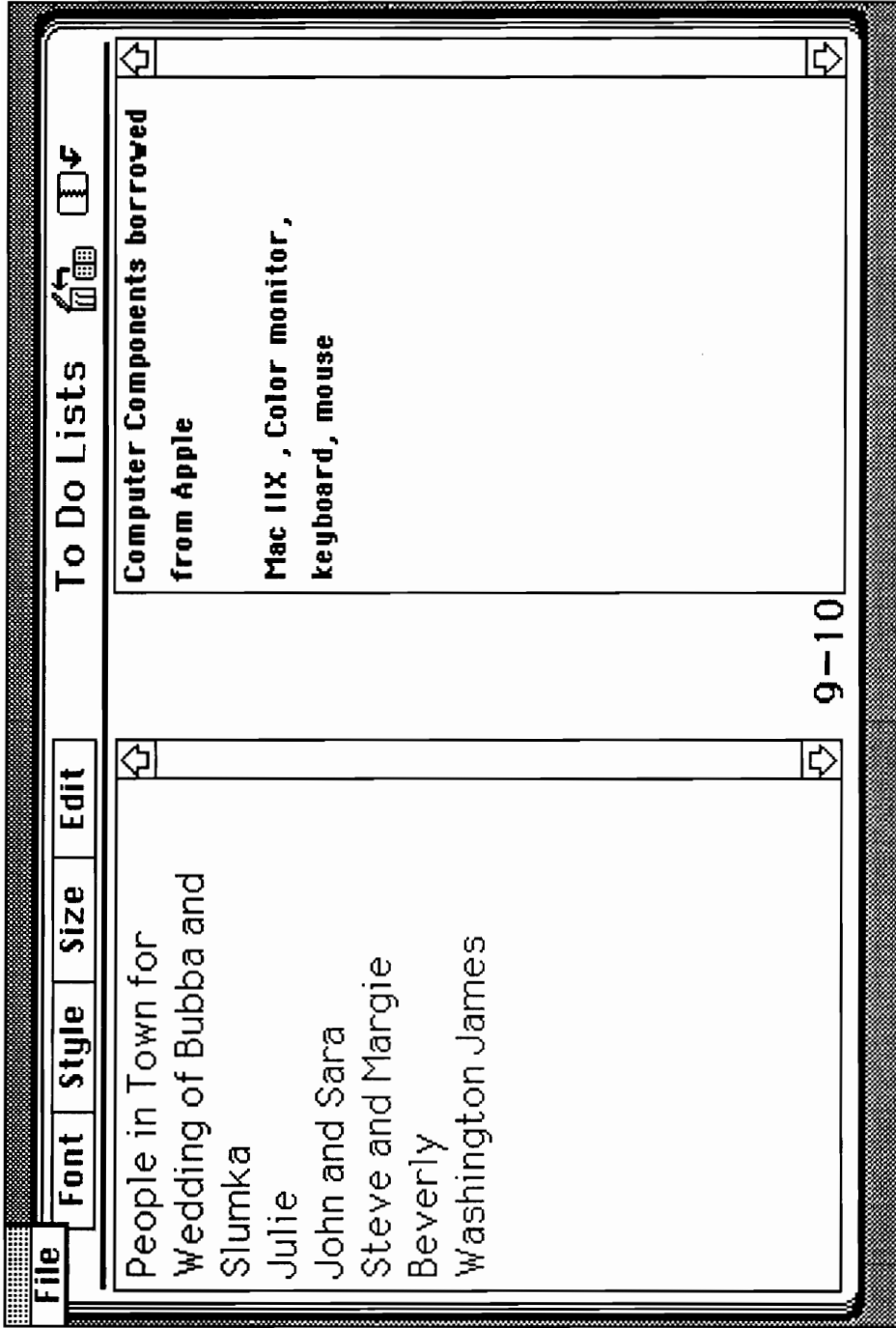


Figure J-21. Inconsistent version B- To Do list function (with delete box open).



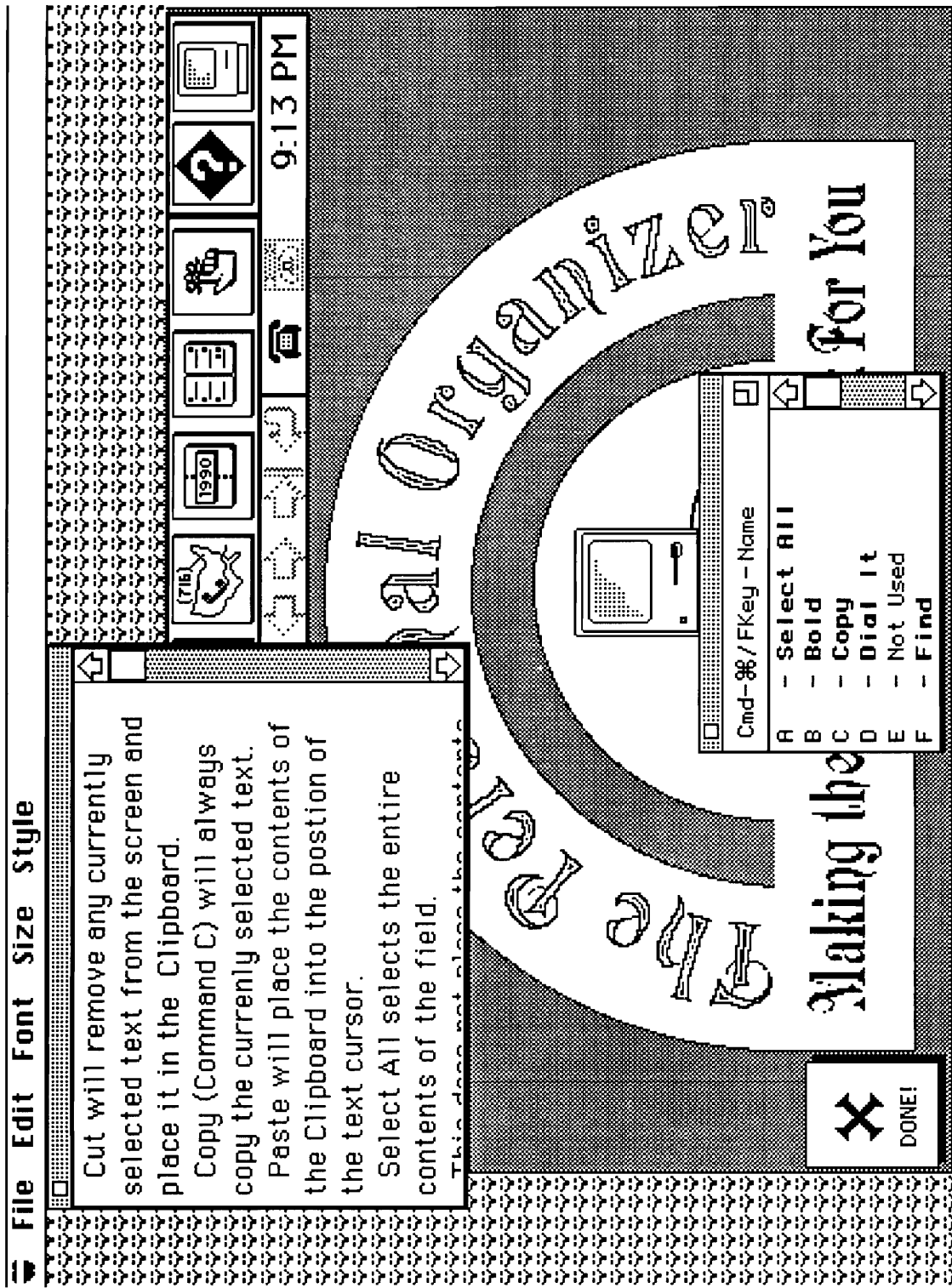


Figure J-23. A demonstration of the help system for the menu bar.

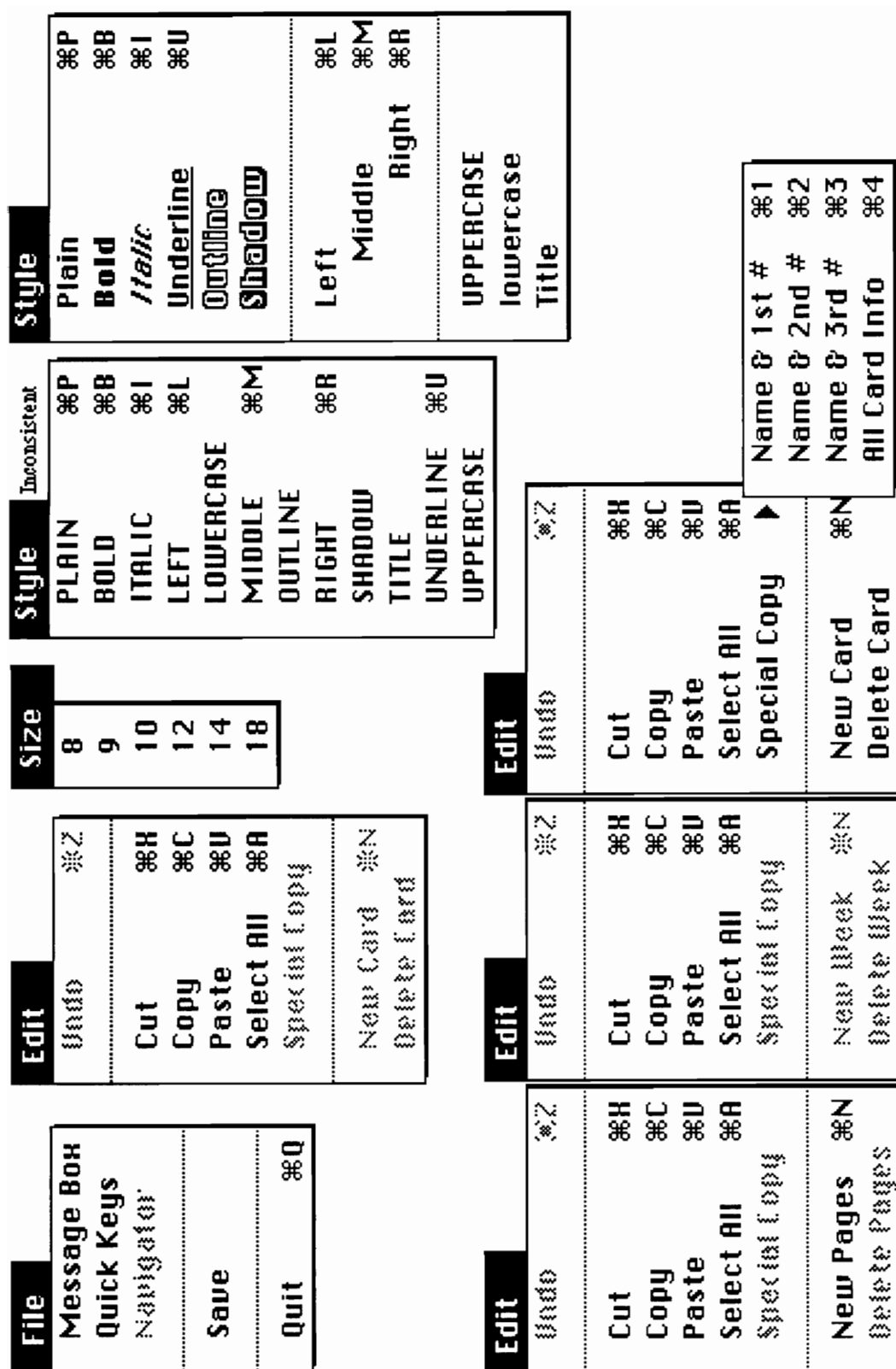


Figure J-24. The menus used in all versions of the application (except the style menu).

---

## APPENDIX K: Technical Information

The research was conducted on a Macintosh IIfx with an Radius high resolution color monitor (1152 x 882 pixel addressability), an 8-bit color board, 8 megabytes of RAM, and an Apple mouse for user input. The Macintosh was operated in the MultiFinder mode running system software 6.0.5. Apple HyperCard 1.2.5 was used for original software development and ported to Silicon Beach's SuperCard 1.5 to complete the project.

This report was generated using the same hardware as previously mentioned. The body of the text was set in 12 point Times on a QMS-PS 410, final copy was set on a Varityper VT300 using Microsoft Word 4.0D. Color prints were generated on an Océ' color printer.

Non-parametric tests were calculated in StatView II by Abacus Software, while ANOVA procedures were run on an IBM 3084 running VMS/SAS 6.0.6 and SuperANOVA by Abacus Software. Charts, Graphs, and Figures were developed using Excel 2.2, Canvas 2.1, MacDraw II, Nisus 3.05, and Cricket Graph.

The Personal Organizer was developed by using the original HyperCard Address, Phone, To Do, and Area Code Stacks as a starting point. These stacks with extensive modifications were eventually combined into a single application. Each version contained about 300 pages of code, and graphics and system overhead, for a total size of 1.5 megabytes per version. They each ran under a Multifinder partition of 3000K, to eliminate memory problems. The application recorded the time, action, place in which the action occurred, and the method of initiation (using an icon, menu or keyboard shortcut). This and summary data (total time, number of Help requests, etc..) was written to a variable, and then written to a file at the end of each session. Performance was unaffected by the time for writing to disk.

Richard H. Miller was born in Milwaukee, Wisconsin, on June 12, 1965. He received his Bachelor of Science degree in Applied Psychology with a minor in Drama and Film from the Georgia Institute of Technology in June of 1987. While at Georgia Tech, he led the human factors effort on an award winning design of a Formula One style sports car for a Society of Automotive Engineer's competition. He also was active in community projects, a Dean's List student, played intercollegiate lacrosse, and was a member of Alpha Epsilon Pi.

In 1987, Richard entered Virginia Polytechnic Institute and State University to pursue a Master of Science degree in Human Factors Engineering. He spent a summer working for the FMC Corporation where he designed the computer interface for an air-defense tank, as well the user interface guidelines for FMC's subcontractors to use when designing interface components for the tank. Richard then served as the principal investigator on a project for assessing the usability and functionality of rapid prototyping software in the Vehicle and Simulation Laboratory at VPI&SU. Interest in rapid prototyping led to research in graphical user interface design.

As one of VPI&SU's technical coordinators for Apple Computers, Inc., Richard designs computer software and hardware solutions for local projects. He helped to install and maintain parts of a state-of-the-art computer network while serving as an intern at the Design Resource Center of Eastman Kodak, and continues to do network administration duties for the Displays and Controls Laboratory and the Virginia Productivity Center at VPI&SU. Richard teaches training classes for computer-based applications, and has been an invited speaker at VPI&SU for sessions on multimedia solutions in the classroom, writing of HyperCard applications, and interface design.

Mr. Miller enjoys the theater, designing gadgets, traveling with his family and friends, playing sports, and reading the hundreds of novels that he has accumulated since starting graduate school. Richard plans on completing his Ph.D. requirements during the 92-93 school year and will pursue a career in human factors and the management of engineering projects.