PRIMING THE COGNITIVE PUMP:
IMPLICIT MEMORY AND
NAVIGATING MULTIPLE WINDOW INTERFACES

by

GARY LORNE MACISAAC

B.Sc. (Honours), The University of British Columbia, 1975
B.Sc., The University of Calgary, 1981

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF

THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE

in

THE FACULTY OF GRADUATE STUDIES

Department of Computer Science

We accept this thesis as conforming
to the required standard

THE UNIVERSITY OF BRITISH COLUMBIA

October 1994

In presenting this thesis in partial fulfilment of the requirements for an advanced degree at the University of British Columbia, I agree that the Library shall make it freely available for reference and study. I further agree that permission for extensive copying of this thesis for scholarly purposes may be granted by the head of my department or by his or her representatives. It is understood that copying or publication of this thesis for financial gain shall not be allowed without my written permission.

(Signature

Department of Computer Science

The University of British Columbia
Vancouver, Canada

Date October 13, 1994.

DE-6 (2/88)

Abstract

When navigating through large-scale information spaces, users may lose track of their location and experience the sensation of being "lost in hyperspace". A common solution applied to this problem is the graphical user interface, using windows to keep track of the pages visited, to show overview maps of the information space, and to highlight information "landmarks", all of them serving the user as reminders in support of the task at hand. My thesis focuses on two different types of memory or memory processes--implicit and explicit--and explores how these may be harnessed in the design of human-computer interfaces. Explicit memory is reflected in tests of conscious recall or recognition of a past event or experience. Implicit memory is revealed through improved performance on tasks that do not require conscious or intentional recollection of previously studied information. I carried out an experiment that used a menu-item selection task, with window arrangement and text justification as the independent variables, where both implicit and explicit memory were assessed. Implicit memory testing focused on the acquisition of a menu-item selection skill under different menu-item mapping conditions; explicit memory testing required recollecting the correct position of target menu-items with the aid of previously seen window displays. Forty undergraduate student volunteers served as subjects. The implicit memory test results showed no effects due to various window arrangement and text justification manipulations. By contrast, explicit memory test performance showed an overall increase in menu-item selection accuracy, an improvement in accuracy across trials, and in addition, it also showed a significant difference in selection accuracy with left- versus right-justified text. The discussion focuses on various aspects of these findings; it explores limits of the present study and outlines plans for future investigations, as well as implications for window management and information presentation.

TABLE OF CONTENTS

List of Tables

List of Figures

Figure		Page

Acknowledgement

Many people provided me with support in various forms during the course of my studies. I thank Dr. Richard Tees, Dr. Anthony Phillips, and many other members of the Department of Psychology for their flexibility in the scheduling of my work. The patience of my colleagues enabled me to carry out this part-time program in conjunction with my full-time position as Computer Analyst. I thank Brian Moorhead, the Psychology Department Engineering Technician, for his assistance with the hardware timing issues. Thanks to Dr. David Lowe, the second faculty reader, and to Brian Pidcock the student reader. I wish to especially thank Dr. Kellogg Booth and Dr. Peter Graf, my research co-supervisors, for their patience and support in accepting the often long intervals of time which would pass between successive stages of work on my project. Above all, I wish to thank my spouse Irene, and my son Andrew, for providing me with five years worth of evenings and weekends at home to complete this program--it would have been impossible to accomplish otherwise. Financial support was provided, in part, by the Natural Sciences and Engineering Research Council of Canada.

CHAPTER ONE

Introduction

Graphical user interfaces are a popular approach to the general problem of user disorientation while navigating through an information space. By maintaining the context for a specific task within a window, users can easily switch their attention among several tasks. However, with large-scale information spaces, such as complex hypertext and hypermedia networks, users continue to experience the "lost in hyperspace" problem as they explore these networks. In many areas, human-computer interface (HCI) research is focusing on the application of low-level visual cues to facilitate information acquisition and navigation by users. This renewed emphasis on the application of psychological principles to user interface design is driven by a desire to reduce the cognitive load on users of increasingly sophisticated information systems. To build upon the progress so far, we must look to new areas of basic research in psychology as a source of user interface design principles.

This thesis directs our attention to recent research on two different measures of memory (implicit and explicit) and focuses on how they can be used to enhance our understanding and aid our improvement of human-computer interfaces. The two measures of interest in this thesis are: an *implicit* memory test (based on a skill learning task) and an *explicit* memory test (based on a recall task). Implicit measures of memory show performance improvements resulting from a previous exposure to information, where the task involved does not require conscious recollection of the information. Explicit measures of memory reflect the conscious recall of a past experience.

A good human-computer interface should become a natural part of the user's working environment. Weiser's (1993) *ubiquitous computing* concept promotes the general notion that computers will be widespread in the users' environment, and that users' should not be aware of computer support when they engage in an activity which draws upon the resources of a computer system. At the same time, the full power of these computer systems would remain instantly available. Measures of cognitive effort may provide us with a mechanism for comparing how successful different interfaces are at achieving Weiser's goal.

Memory represents a major component of human cognitive capabilities. Interface designers have used visual, auditory and other cues to assist users with their tasks and to reduce the need for rote memorization of interface components. Menus are one of the obvious mechanisms; they reduce the need for a user to remember command names. It is known that placing user interface elements such as buttons, menus and menu-items at standard locations on the display will facilitate effective use of the interface. However, we do not know to what extent the visual appearance of the interface constrains a user's behaviour when locating objects on a display. A number of factors including the arrangement of windows, menus, and the layout of text present the user with a frequently changing visual picture on the display which may influence the speed and accuracy of users as they manipulate information. Implicit and explicit measures of memory performance may provide one mechanism for evaluating and comparing these user interface characteristics.

The purpose of the research reported in this thesis was to investigate the influence of visual context on window navigation in a human-computer interface. Specifically, the research examines the effects of different window and text displays on different measures of memory, both implicit and explicit, focusing on menu-item selection. Two hypotheses were tested by the study:

A:

- H1: The arrangement of windows and justification of text on a computer display will influence performance on implicit and/or explicit memory tests.
- H0: The arrangement of windows and justification of text has no influence on memory performance.

B:

- H1: Given that window arrangement and text justification do influence performance on implicit and/or explicit memory tests, this influence is mediated by specific aspects of the display geometry.
- H0: There is no difference in memory performance across display geometry.

This study represents the first known (to the author) investigation that uses implicit *and* explicit memory tests in the exploration of performance on human-computer interfaces.

The remainder of this chapter provides an outline for each of the following chapters which comprise this thesis.

Chapter Two discusses the literature on window and information system navigation from the field of human-computer interface studies. An historical overview of work from the first window environments to modern hypermedia systems introduces the key components of these interfaces. In particular, studies associated with window attributes, window arrangement and window system control are reviewed. The studies of menu structure, information presentation, and information navigation highlight the interdisciplinary nature of user interface design and suggest a linkage to a recently expanding area of cognitive psychology research which may prove important to the design of future interfaces: the study of implicit memory.

Chapter Three reviews relevant aspects of the psychology of human memory. Most people associate the term memory with the conscious or intentional recall of previous experiences. This type of memory is referred to as *explicit* memory. A brief review of human memory research summarizes specific studies emphasizing a second form of memory or memory process referred to as *implicit* memory. This form of memory is evident through a facilitation or change in observed task performance due to the unintentional retrieval of previously acquired information.

Three important aspects of memory research are considered. First, a short review of the commonly understood concepts of short-term and long-term memory are considered to provide a starting point for the examination of distinctions in memory. Next, important contributions from research on amnesic patients expand upon the notion of different forms of memory. By showing that performance on explicit memory tests is severely impaired, these studies demonstrate that the same amnesic patients exhibit normal levels of performance on implicit memory tests. These studies provide compelling evidence that human memory involves at least two different types of memory or memory processes. The new evidence for different distinctions in memory and the associated techniques for examining these distinctions may provide the user interface designer with new methods for assessing user interface features. Finally, studies involving explicit and implicit memory tests on normal subjects show consistent differences in measures of performance between these types of tests. Research examining performance on implicit tests using word fragments, pictures, and the acquisition of a skill show that this form of memory or memory process exists across a number of modalities. The rapidly accumulating evidence for these differences in memory form or function may open up new possibilities for enhancements to the human-computer interface.

Chapter 4 describes an experiment undertaken to measure memory performance on explicit and implicit memory tests. These tests were carried out by employing menu-item selection activities in the context of a simple window-based user interface. Different combinations of window arrangement and text layout were associated with a question answering task where item selection time and selection errors were measured.

The experiment consisted of 32 subject *sessions*. Each session included a *mouse practice phase*, a *target training phase* and a *recall test phase,* in that order. The mouse practice phase familiarized subjects with using the mouse and involved a variable number of *mouse menu-item selections* carried out by the subject until a criterion level of performance was reached. The target training phase provided the measure of implicit memory performance and was designed to expose subjects to different screen display patterns (window arrangement and text layout) each requiring a specific sequence or a random sequence of mouse menu-item selections (a menu-item mapping) across 48 *target training blocks*. The recall test phase represented the explicit measure of memory performance and was designed to test a subject's ability to accurately recall the position of the correct answer chosen on a previously viewed target training trial, and included 16 *recall test blocks*.

Chapter Five analyzes the results of the experiment in the context of the two hypotheses investigated. No significant difference in menu-item selection speed and accuracy was observed between menu-item mapping conditions during training. No significant difference in menu-item selection accuracy was observed between the menu-item mapping conditions on a recall test. The fixed-mapping condition did produce slightly better selection accuracy relative to the variable-mapping condition. Subjects in the experimental group showed significantly better selection accuracy than controls on a recall test. All experimental subjects showed a large decrease in the error associated with menu-item selections across trials on a recall test. No difference in menu-item selection performance at recall was evident between the two tested window arrangements, but selection accuracy was significantly better when text appeared in the left-justified format compared to when the right-justified format was used.

Chapter Six discusses the study in the context of window-based computer interfaces and hypermedia/hypertext navigation. The similarity of results obtained in this study with basic research from the psychology literature is examined. While the results were not conclusive, there were many trends pointing to an influence of implicit memory on the behaviour of users in the simple window-based user interface employed in this study. Future directions for this research are suggested which could refine our understanding of how implicit memory affects performance on a number of tasks in window-based computing environments.

CHAPTER TWO

Window-based Interfaces and Hypermedia Navigation

A significant issue that must be addressed by designers of human-computer interfaces is the loss of task context which can lead to user disorientation during exploration or search through a large body of information. Researchers have considered many different approaches to this question--often referred to as "the navigation problem" or "the lost in hyperspace problem"--with varying degrees of success. Graphical user interfaces (GUIs) are one of the most common solutions applied to the navigation problem. In the first section of this chapter I review the development of graphical user interfaces by highlighting significant research and design contributions associated with window[1] and hypertext[2]/hypermedia systems. Figure 2-1 and Table 2-1 summarize the major research developments associated with graphical user interfaces and hypermedia/hypertext systems. The next section provides a description of specific attributes found in modern window interfaces, outlining the interface options available to system designers. The last section examines studies of user performance in window-based GUIs, how a window manager[3] can control application displays to implement a window management policy and how these results have been combined in new window and hypertext/hypermedia interfaces.

Historical Context

The human-computer interface has changed substantially over the past 30 years. Early interactive interfaces presented users with a scrolling roll of paper fed through a typewriter-like device. This interaction method provided an immediately accessible, permanent record of

[1]A window represents a portion of the screen image which may contain graphics, text, video, pictures or a combination of these items within a bounded rectangular region of the display screen (see Figure 2-2, page 10).
[2]Hypertext/ hypermedia are terms given to a body of interconnected information where the connections or links are embedded within the information itself.
[3]A window manager is a software system which assists the user by monitoring and controlling different contexts which are physically isolated as distinct windows on the screen.

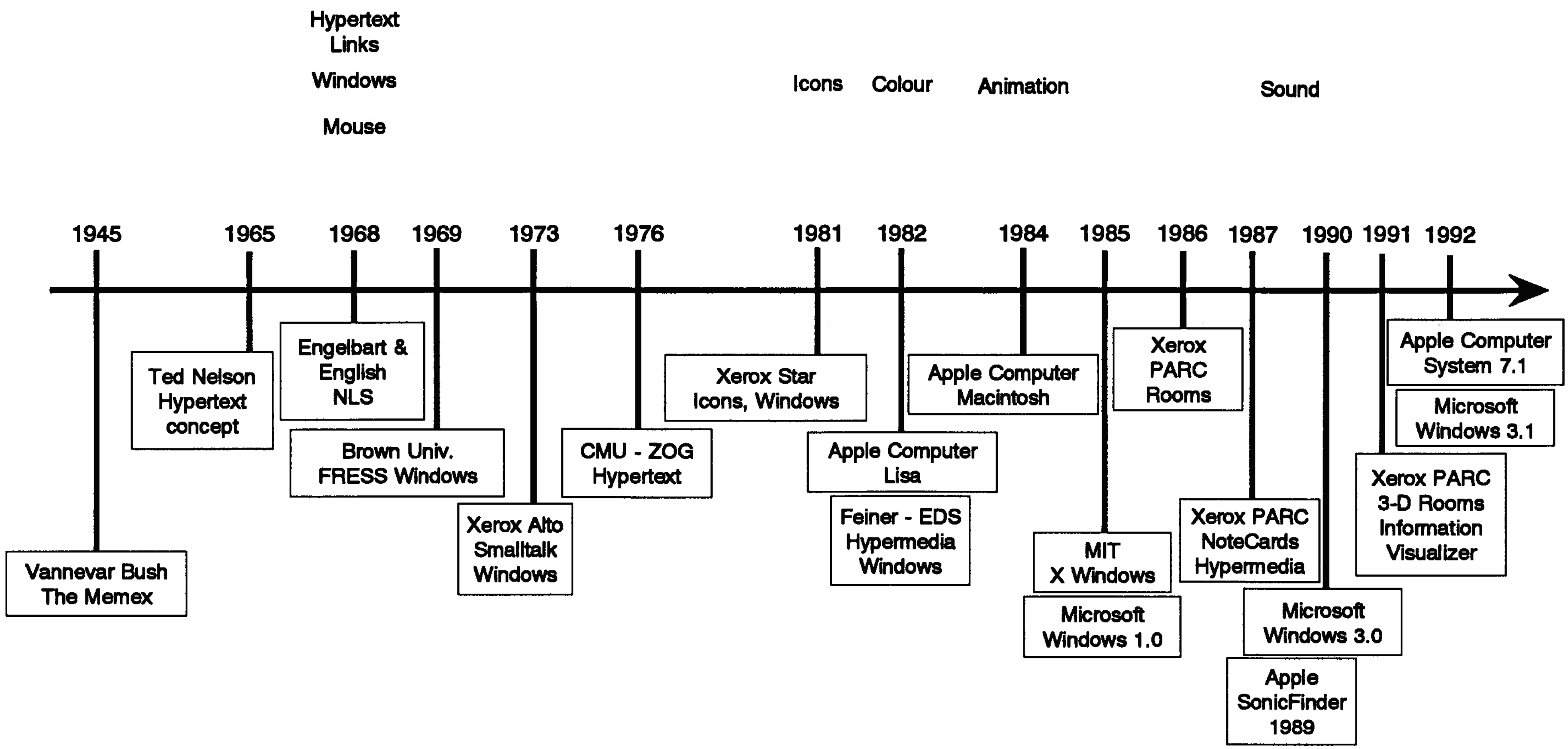

Figure 2-1. A timeline illustrating the development of window-based user interfaces and associated hypertext/hypermedia systems. This does not represent an exhaustive list of window and hypermedia systems. It does provide a concise historical view of many significant commercial and research user interfaces. The timeline is not to scale.

Table 2-1. Chronology of significant window and hypermedia system user interface developments.

System	Year	User Interface Features	References
Memex proposal	1945	Extension of human memory	Bush (1945)
The Hypertext proposal	1965	Non-sequential writing with links to other text chunks embedded in the document.	Nelson (1975)
NLS/Augment (oN-Line System)	1968	Windows, hypertext links, structured browsing, mouse	Engelbart & English (1968)
FRESS hypertext system	1969	Windows, hypertext links	Yankelovich et al., (1985)
Xerox Alto and Smalltalk	1973	Overlapping windows, pop-up menus	Lampson (1988)
ZOG hypertext system	1976	One or two windows, fixed menu location, no scrolling within windows, fast system response	Mantei (1982); McCracken & Akscyn (1984)
Xerox Star	1981	Overlapping windows, pop-up menus, use of icons as a window representation	Johnson et al., (1989)
Apple Computer Lisa	1982	Overlapping windows, pop-up menus, icons	Williams (1984)
Apple Computer Macintosh	1984	Overlapping windows, pop-up menus, icons	Williams (1984)
Sapphire Window System	1984	Overlapping windows, pop-up menus, icons with window state information, icon window, animation	Myers (1984)
X Windows	1985	Overlapping windows, separation of window management from application interface management, colour	Scheifler & Gettys (1986)
Microsoft Windows 1.0	1985	Overlapping and tiled windows, pop-up menus, icons	Shneiderman (1992)
Xerox PARC Rooms	1986	Virtual room metaphor, multiple overlapping windows, context retention in rooms	Henderson & Card (1986)
Xerox PARC NoteCards	1987	Overlapping windows, hypermedia links, overview windows	Halaz et al., (1987)

(table continues)

Table 2-1. *(continued)*

System	Year	User Interface Features	References
Apple Computer SonicFinder	1989	Auditory icons	Gaver (1989)
Microsoft Windows 3.0	1990	Overlapping and tiled windows, pop-up menus, icons, program group windows, significant use of colour	*Microsoft Windows User's Guide, Version 3.0* (1990)
Xerox PARC 3-D Rooms, Information Visualizer	1991	Cone trees, perspective wall, 3-D Rooms and animation for presentation of information detail and context.	Robertson, Mackinlay, & Card (1991); Mackinlay, Robertson, & Card (1991); Card, Robertson, & Mackinlay (1991)
Apple Computer Macintosh System 7.0	1992	Overlapping windows, pop-up menus, folders, icons, significant use of colour	*Macintosh User's Guide* (1992)
Microsoft Windows 3.1	1992	Overlapping windows, pop-up menus, icons, program group windows, colour, drag and drop object manipulation	*Microsoft Windows User's Guide for the Microsoft Windows Operating System, Version 3.1* (1992)

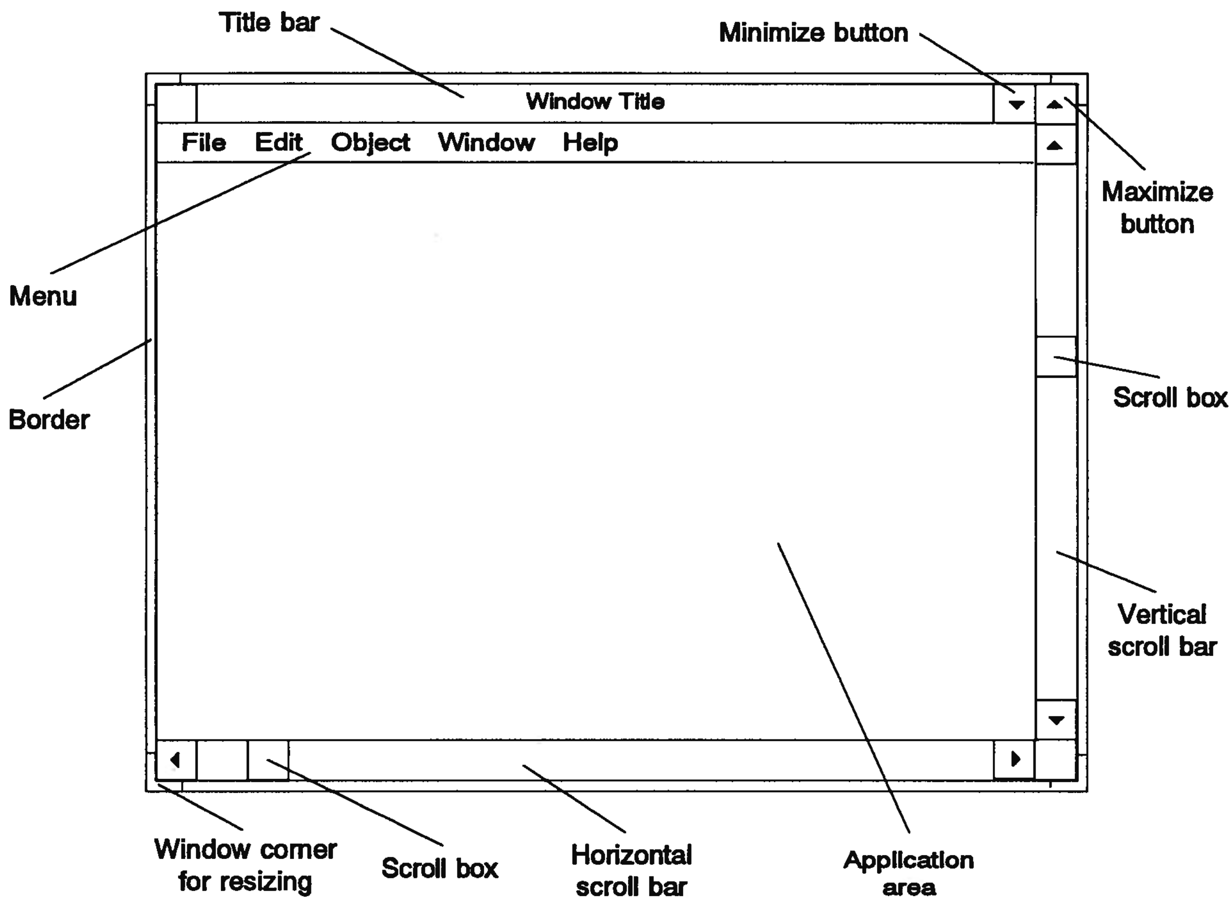

Figure 2-2. Major components of a window illustrating features found in many window systems.

sequential user requests and system responses. "Scrolling" was accomplished by physically handling the paper. The context associated with an item of interest was provided by the sequential location of information on the printed paper stream. The limited computational resources, small-scale data sets and low speed Teletype terminals did not overburden the user's ability to make decisions and maintain task context. The subsequent development of high-speed display screens (the so called "glass Teletype") allowed for electronic scrolling, backwards and forwards, through what was still a sequential record of activity. While the user could quickly move back and forth through an interaction history, the context associated with an item of interest was limited to the visible region on the display screen.

Rapidly improving computer hardware performance and more sophisticated system software capabilities changed the nature of computer applications and their user interfaces. Activities such as the writing and editing of documents, on-line searches for information and the software development process did not always follow the older sequential interaction pattern, but instead often forced users to recall information from a variety of previously viewed screens. As designers looked for ways to reduce memory demands on the user, different components of graphical user interfaces (GUIs) began to appear. Windows, pop-up menus[4], icons[5], mice[6] and hypertext links all found their way onto the user's "electronic desktop". The widespread use of Windows, Icons, Menus and a Pointing device formed the basis for the *desktop metaphor*--an electronic representation of the physical desktop--and resulted in this form of GUI becoming known as a "WIMP" interface. Table 2-2 provides detailed descriptions of these GUI components. At the same time, continued improvements to computer hardware and software provided the operating environments necessary to introduce colour, sound, animation and new metaphors for use in these user interfaces. Today, GUIs

---

[4]Menus are screen displays which include one or more objects sensitive to a selection action.
[5]Icons are small representations of windows or other objects.
[6]The mouse pointing device generally has one, two or three buttons on top of it along with a mechanical or optical method of sensing information about changes in its position relative to a physical desktop.

Table 2-2. Expanded definitions of terms used in the text.

Term	Definition
Windows	A window represents a portion of the screen image which may contain graphics, text, video, pictures or a combination of these items within a bounded rectangular region of the display screen (Figure 2-2). A window may overlap other windows, thereby partially obscuring their contents, or windows may be tiled so that every window is allocated a portion of the available display screen area without obscuring the view of any other windows (Figure 2-3a,b).
Menus	Menus are screen displays which include one or more objects sensitive to a selection action. In some cases menus may be located in a specific region of the screen. In other cases, as with pop-up menus, they may appear and disappear when the mouse cursor moves across a specific region of the display screen or a mouse button is pressed. Menu items provide a means of giving a command thereby causing the computer system to carry out some action.
Icons	Icons are small representations of windows. Sometimes icons are drawn as a miniature replica of the full-size window, in other cases they use generic symbols that represent an application category, such as word processors. Icons provide a means of reducing screen clutter by reducing the demand for display space by windows that are not in use.
Mouse pointing device	The mouse pointing device generally has one, two or three buttons on top of it along with a mechanical or optical method of sensing information about changes in its position relative to a physical desktop. By pressing the buttons, the user can signal that some action is required by the system. The mouse provides a means of moving a visible tracking symbol or cursor around the display screen, based on the physical movement of the device on a firm surface. Once the cursor has been positioned over a menu-item, the user can invoke the command by clicking one of the mouse buttons. The mouse also provides a mechanism for the selection of one or more data items which will serve as the object of a command. A WIMP interface combines the four components (Windows, Icons, Menus and Pointing device) into a coordinated dialog between the user and the computer system (Nielsen, 1993).

(*table continues*)

Table 2-2. *(continued)*

Term	Definition
Hypertext/ hypermedia	Hypertext/ hypermedia are terms given to a body of interconnected information where the connections or links are embedded within the information itself. Hypertext usually refers to linked chunks of text. Hypermedia generalizes the definition to include linked chunks of text, pictures, video sequences and sound. In the general case, hypermedia can be thought of as information with embedded menus. Each link item within the currently displayed chunk of information will, when selected, cause the next chunk of information to appear on the display, thus behaving like a menu-item selection. Hypermedia and hypertext systems are commonly implemented using a window-based user interface.
Window manager	A window manager is a software system that assists the user by monitoring and controlling the display of distinct windows on the screen. Window managers allow users to work on several tasks at the same time by maintaining the context of each task in a physically distinct region of the screen. Users can easily and quickly shift their attention among different tasks by selecting the relevant window. In this way, window managers act as a memory aid by significantly reducing the detail a user needs to recall as they switch among tasks. A window manager can also implement a user interface (UI) policy by controlling the arrangement and appearance of individual windows on the display screen. Policies can include the specification of tiled vs. overlapping windows, constraints on window layouts and the attributes associated with the window which is currently accepting input from the user.

are found on millions of computer workstations, and have a major effect on how people carry out information handling tasks.

Window Systems. In this section I will examine the development of graphical user interfaces, looking at various window system characteristics. Table 2-1 shows a chronology of notable features associated with window systems. Douglas Engelbart, while at the Stanford Research Institute, is generally credited with the invention of a system for interconnecting chunks of information, windows, and the mouse as components of the human-computer interface in his oN-Line System (NLS) (Engelbart & English, 1968). Drawing upon ideas first proposed by Vannevar Bush (Bush, 1945), Engelbart (1963) described a model for his H-LAM/T system (Human using Language, Artifacts, and Methodology, in which he/she is Trained). This work guided Engelbart's thinking as he designed a method, in NLS, for creating connections between arbitrary-sized and labeled items of information. These connections became known as hypertext links (Nelson, 1975). The NLS system provided a mechanism for managing information as entities called "statements" which were structured in a hierarchical manner. Through the NLS view generator, a subset of the linked units of information (nodes) could be "frozen" on the upper portion of the screen while other statements appeared in the lower portion of the screen. The NLS view generator would be considered a browser today, with its ability to provide different windows into data at various levels of detail. The NLS system represented the first attempt at reducing the cognitive workload of users by shifting the maintenance of information structures to the computer. It is interesting to note one particular observation of Engelbart and English (1968) concerning the use of a structured data presentation:

> We have come to write all of our documentation, notes, reports, and proposals according to these conventions, because of the resulting increase in our ability to study and manipulate them during composition, modification, and usage . . . . We have found it to be fairly universal that after an initial period of negative reaction in reading explicitly structured material, one comes to prefer it to material printed in the normal form.
>
> Engelbart and English (1968), pg. 87 of Greif (1988).

By the early 1970's, research and development work at Xerox PARC--the Xerox Palo Alto Research Center--produced the Smalltalk graphical environment on a system called the Alto, the first professional-quality personal workstation. While some of this work addressed system software issues, the researchers were also interested in understanding how they could share a scarce resource, display screen area, among several applications. The Alto system was the first to support movable, overlapping windows and pop-up menus for accepting user requests (Lampson, 1988). A window represents a portion of the screen image which may contain graphics, text, video, pictures or a combination of these items within a bounded rectangular region of the display screen. A window may *overlap* other windows, thereby partially obscuring their contents, or windows may be *tiled* so that every window is allocated a portion of the available display screen area without obscuring the view of any other windows (Figure 2-3a and b). Menus are screen displays which include one or more objects sensitive to a selection action. In some cases menus may be located in a specific region of the screen. In other cases, as with pop-up menus, they may appear and disappear as the mouse cursor moves across a specific region of the display screen or a mouse button is pressed. Menu items provide a means for specifying a command, thereby causing the computer system to carry out some desired action.

A subsequent system, the Xerox Star workstation, provided users with a high-resolution black and white display containing up to six non-overlapping windows and icons (Johnson et al., 1989). *Icons* are small representations of windows. Sometimes icons are drawn as a miniature replica of the full-size window, in other cases they use generic symbols that represent an application category, such as word processors or drawing software. Icons provide a means of reducing screen clutter by reducing the demand for display space from windows that are not in use. Introduced in 1981, the Star was the first commercial system to offer a window-based user interface. Although the Star supported overlapping windows, initial testing indicated users spent much of their time adjusting window sizes and positions to avoid overlap. This fact led the designers to constrain application windows to not overlap, forcing them instead to be tiled. Along with windows, the Star user interface included icons and menus with features based on good graphical design principles. The goal was to produce a UI which did not place significant cognitive demands on the user, making the system easy to learn and obvious to use.

Apple Computer's first try at marketing a window-based computer occurred in 1982 with the Lisa multitasking system. The Lisa interface derived many of its features from the Smalltalk environment and the Xerox Star. The Lisa was unfortunately too expensive at the time to find widespread acceptance by users. Thus, in early 1984 Apple introduced the Macintosh system (Williams, 1984). The Macintosh user interface was strongly influenced by features found in the Lisa. Using multiple overlapping windows, pop-up menus and icons, the Macintosh attracted a loyal following of new users who had previously avoided personal computers. Its one drawback was the small (9-inch diagonal) monitor which placed significant constraints on the number of windows a user could keep open or visible in the early models. However, the Macintosh was a driving force behind the growth of lower cost WIMP interfaces for personal computers and it challenged competing software vendors to develop similar graphical user interfaces.

The Sapphire system (Myers, 1984), which featured overlapping windows, introduced an innovative interface feature: an *icon window* which provided users with information about the current state of any open windows, even those not currently visible. Icons contained in this window might include descriptive symbols representing the progress of a computation in a window, the error state of a window, or whether user input was required. Sapphire was one

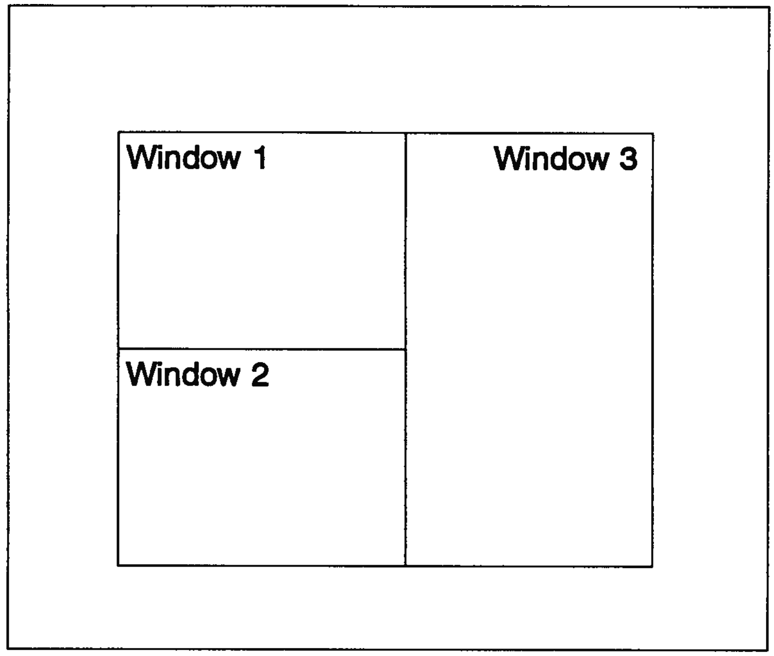

Figure 2-3a. An example of tiled windows. Visible windows must adjust size and aspect ratio to co-exist with other windows. All contents (of visible windows) can be seen simultaneously.

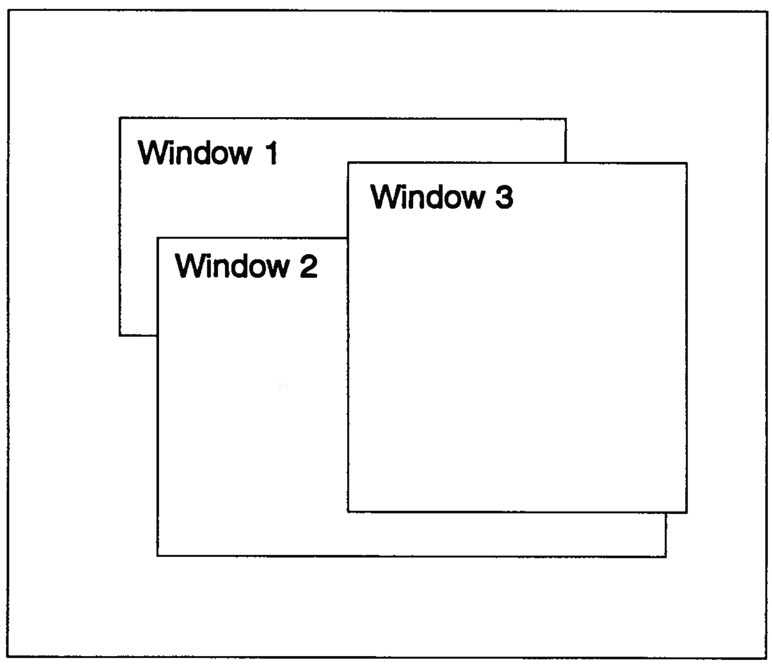

Figure 2-3b. An example of overlapping windows. Windows can take on a size and aspect ratio that is "natural" for the application, but the user cannot see all of the content.

of the first interfaces to make widespread use of animated icons to present system and application context information to the user. Together with the Macintosh, the Sapphire system furthered the application of one type of animated icon--the percent-done indicator--now common in a number of current user interfaces (Myers, 1985).

In 1985 the X Windows system was introduced. It approached graphical user interfaces from a systems architecture point of view by supporting a user-specified window management policy. X Windows represented the combination of a window manager interface and an application interface (Scheifler & Gettys, 1986). This was a significant development. The window manager interface was responsible for the organization of the overall display, assisting the user by monitoring and controlling the display of distinct windows on the screen. Window managers allow users to work on several tasks at the same time by maintaining the context of each task in a physically distinct region of the screen and coordinating visual context switches between the tasks. Users can easily and quickly shift their attention among different tasks by selecting the relevant window. In this way, window managers act as a memory aid by reducing the detail users need to recall as they switch among tasks.

Window manager behaviour is independent of application program behaviour. As a result, the window manager interface can control window attributes such as colour, title bars and borders. In contrast, the application interface retains control of the presentation and manipulation of information within an application window. A primary design goal of X was to support many different window manager and application interface features. According to the X philosophy, applications should *react* to window manager actions, such as resizing, by appropriately reformatting the window's contents. According to Scheifler and Gettys (1986), applications should not direct or influence window management policy. A window manager could implement a user interface (UI) policy by controlling the arrangement and appearance of individual windows on the display screen. Policies could include the specification of tiled vs. overlapping windows, constraints on window layouts and the assignment of various attributes associated with the window currently accepting input from the user. Although work on X was primarily targeted at systems-level software issues, the architecture provided an excellent base upon which GUI standards could evolve. With these and other developments, X provided a useful environment for additional user interface experiments (Sun, Cowan & Booth, 1990).

The user interface features of Microsoft Windows are not particularly novel, however, the very rapid growth in use of this particular GUI (tens of millions of installed systems) suggests that Windows will exert a significant influence on future UI and application development for the IBM PC and compatibles. Microsoft had the first version of its Windows environment operational in 1985. However, it was not until the 1990 release of Windows version 3.0 that the system came into widespread use on desktop personal computers. Supporting both overlapping and tiled window arrangements, user activity centred around a window called the Program Manager which functions as a "shell" or command processor (*Microsoft Windows User's Guide for the Microsoft Windows Operating System*, 1992). Besides serving as the launch point for applications, one or more windows within the Program Manager window also provided a mechanism for categorizing and grouping applications. These program group windows could appear on the screen as icons, tiled windows or overlapping windows, allowing the user some flexibility in arranging screen layout. Windows 3.0 provided users with a much desired GUI on top of the character-based DOS operating system found on the majority of IBM-compatible personal computers. Unfortunately, the Windows system did not sufficiently constrain the repositioning of tiled windows within the Program Manager. The selection of a particular subwindow followed by a request to retile the display could often result in considerable movement and resizing of many adjacent subwindows. After retiling, the display may look quite different from its appearance just moments earlier--often forcing the user to carry out a search to locate and select a desired application icon. This example shows how easy it is to overlook a disorienting aspect of a GUI which may subsequently affect the performance of millions of computer users.

Throughout the 1980's window systems continued to attract the attention of GUI researchers who wished to push the limits of the technology. For additional historical information on these user interfaces, the reader is directed to a survey by Myers (1988) which provides a detailed taxonomy of window presentation methods, window interaction methods and window manager functions. The studies and system developments considered so far illustrate a number of new approaches to the problem of navigation through an information space. In parallel with window system research, work on hypertext and hypermedia systems investigated the ability of windows and other user interface components to support the browsing and searching of large interconnected sets of information. The following section briefly examines the evolution of hypertext and hypermedia solutions to the "lost in hyperspace" problem.

Hypertext and Hypermedia. The late 1960's through the late 1970's produced a considerable body of research into hypertext and hypermedia systems (see Figure 2-1 and Table 2-1). In 1968 Ted Nelson, working at Brown University with Andries van Dam and several students, built the Hypertext Editing System (HES) which was used for the Apollo Space Program documentation. Nelson, who created the term "hypertext", considered *documents* to be the fundamental units upon which a system would operate (Conklin, 1987). A document could include "windows" into other documents with access to these windows implemented via a linking mechanism. Hypertext usually refers to linked chunks of text. Hypermedia generalizes the definition to include linked chunks of text, pictures, video sequences and sound. In the general case, hypermedia can be thought of as information with embedded menus. Each link item within the currently displayed chunk of information will, when selected, cause the next chunk of information to appear on the display thus behaving like a menu-item selection. While Ted Nelson continued to promote the concepts of hypertext and hypermedia, he never really focused on specific user interface navigation issues associated with these systems. Subsequent work by van Dam and others (Yankelovich, Meyrowitz, & van Dam, 1985) expanded upon HES to produce the File Retrieval and Editing System

(FRESS). That system used multiple windows for information presentation. Two mechanisms, tags and jumps, supported the interconnection of information chunks appearing in different windows. *Tags* provided one-way paths to the next information item while *jumps* allowed for bidirectional movement between chunks in different windows. Unfortunately, the interface did not provide spatial cues to relate information across different windows, making it difficult for readers to keep track of their locations within the information web.

Early systems such as the Electronic Document System (EDS) (Feiner, Nagy, & van Dam, 1982) and "ZOG" (Mantei, 1982; McCracken & Akscyn, 1984) examined navigation problems associated with multiple window displays, embedded menus and linked information spaces. EDS made substantial use of multiple tiled windows and regions within the windows, called *buttons*, which were sensitive to mouse selection actions. Composed of three subsystems (the Picture Layout System, the Document Layout System, and the Document Presentation System), EDS used a high resolution colour display to present documents, images, animations and maps. Two types of *map* were available to provide the user with context information. The *history map* was a timeline set of miniatures (icons) showing previously viewed windows. The *neighbors map* showed icons representing the set of windows the reader could have come from and could subsequently jump to next. Similar navigation features subsequently appeared in commercial personal computer hypermedia-like systems such as Apple's Hypercard (*Hypercard User's Guide*, 1988) and Asymetrix Corporation's Toolbook (*Using ToolBook: A Guide to Building and Working with Books*, 1989).

The ZOG system presented a more rigid user interface with the goal of providing a modeless, easy-to-learn environment supporting information exploration. Information appeared in either full- or half-screen frames (tiled windows), without support for scrolling. If a given chunk of information did not fit within a frame, it was split between two successive frames. Links to other frames were identified by a prefix character and a text description separate from the main text but within the frame. Navigation menu options were located in a third window across the bottom of the display screen. ZOG became a commercial product in 1983 running on Sun workstations under the name KMS (Knowledge Management System). ZOG and KMS used static window configurations and very fast screen update in an attempt to reduce the chance of a user becoming disoriented in the information space.

Various metaphors have been associated with systems to provide users with a familiar mental model for information management tasks. A hypermedia system from Xerox PARC called NoteCards presented users with a multiple-window environment resembling a collection of 3" by 5" notecards (Halaz, Moran, & Trigg, 1987). Each notecard was implemented as a standard editable Xerox Lisp window which could overlap other windows on the screen. A card could include an arbitrary amount of text, a structured drawing or a bitmap image and was differentiated by its title and the data type of its contents. Links, represented by link icons, could appear anywhere within the source card, with a link destination always being another complete card. Special "browser cards" presented an interconnected set of notecards to introduce spatial context and provide users with a quick way to jump to related information. NoteCards was designed for an individual or small group to author and manipulate a small information network, with the browser cards and a limited search capability providing users basic orientation and navigation tools.

Experience has shown that serious navigation problems occur in large-scale distributed information systems (Nielsen, 1990). Now on its way to becoming a very large system, The World Wide Web (WWW), developed by researchers at the CERN particle accelerator in Switzerland, has grown into a widely distributed hypermedia system on the Internet (Krol, 1992). The WWW provides users with access to text, compressed video, images, graphics and sound through embedded links. WWW browsers are currently available for many window systems including Microsoft Windows, Apple Macintosh, NeXTStep and X Windows. The Web is organized around *pages* of information which appear within a window or screen display. A Home Page provides the user with a starting point for entry into the Web. The Home Page may be stored on any WWW server connected to the Internet. As users traverse links, a History Page is maintained which provides a means of immediately linking back to any previously viewed page during a WWW session. The WWW is a rapidly growing hypermedia system with hundreds (and potentially thousands) of authors. Given the ad hoc, constantly changing nature of Web interconnections and information sources, the system presents interesting challenges and possibilities for testing various navigation and user interface strategies.

Drawing upon the experience gained with window-based computing environments, hypertext and hypermedia researchers offer a number of window characteristics to enhance user access to information. In the next section, I examine the components of a window interface and how these components provide support for user tasks. The final section then examines our current understanding of window use by people to accomplish a task, how window manager software can affect task context and how cognitive and perceptual factors can be used to support navigation through an information space.

Window Widgets - The Expression of the Interface

Graphical user interfaces are now the interaction method of choice by the majority of workstation users. However, while multiple-window GUIs are visually appealing and widely used, the advantages and disadvantages of specific design options for these interfaces are still poorly understood (Shneiderman, 1992). An examination of window interface attributes including window appearance, window arrangement and window control mechanisms will introduce the design choices available for these systems.

Modern window systems include a number of components that define the structure and behaviour of a window and provide us with cues to support task context. These parts include:

Borders. Window borders provide a visual boundary for the information and controls associated with a given window (Figure 2-2, page 10). Everything appearing within the border represents the local context for the application which owns the window. Some window systems use the border to indicate the listening state of the window (i.e., whether the window is currently waiting for input from the user). For example, in the Xerox Star and the Sapphire systems, the window border changes colour to indicate it is in the listening state. The change in appearance of the border may provide a visual cue or reminder to elicit the correct action from the user.

Buttons. Window displays can include identifiable areas which are sensitive to selection actions with the mouse or other pointing device (Figure 2-2, page 10). These components of the window and border provide mechanisms for the user to resize the window, reduce the window to an icon or invoke a pop-up menu. For example, the Microsoft Windows 3.1 environment places buttons for minimizing and maximizing a window in the upper right corner of the window. Selection of the "minimize" button reduces the window to an iconic representation of the application which owns the window. Selection of the "maximize" button causes the window to expand, filling the entire display screen.

The ability of the border to respond to resizing operations can be indicated by changes in the shape of the mouse cursor as it passes over a specific portion of a window boundary. This technique, used by Microsoft Windows, allows for resizing at any of the four corners or sides of a window. In contrast, the Macintosh environment uses a button located in the lower right corner of the window for all resizing actions. The Macintosh window is treated as if it were pinned at the upper left corner while the user moves the lower right corner to its new location.

Text or Icon Menus. A considerable amount of empirical work exists examining depth (levels of submenus) versus breadth (number of items per menu) and other issues in menu construction (Norman, 1991). Menus provide users with access to operations on a window or its contents by enumerating available actions, thereby reducing user memory load. Menus can appear in many forms. For example, icons distributed on a display screen are a form of explicit menu; each icon represents one option out of the set of options visible on the display. The Apple Macintosh and Microsoft Windows control panels are examples of explicit menus. However, a more common approach is to organize explicit menus into a hierarchy. In this case, the selection of a menu item could result in a new list of menu choices. Studies show that the time taken to make a selection in a hierarchy is significantly reduced for broad (one to three levels) as opposed to deep designs using four or more menu levels (Norman, 1991).

Buttons in a hypertext/hypermedia environment provide visual indicators of links to other items in the information space. This produces an environment in which the data functions as the menu, potentially resulting in a complex cyclic network of links and nodes. The link items represent an embedded menu; the available menu options are highlighted in some manner but represent an integral part of the object or document currently visible on the display screen.

A major area of research on hypertext and hypermedia systems is directed at understanding why users become easily disoriented in these systems (Mantei, 1982; Nielsen, 1990; Wright, 1991). A study of embedded versus explicit menus found that users work faster with embedded menus (Koved & Shneiderman, 1986). The within-subject study examined performance on a question answering task. Access to the required information was provided through hierarchically structured (explicit) menus and embedded menus. Performance, measured in terms of the number of correctly answered questions, was better in the embedded menu condition. As well, subjects required fewer menu selections to arrive at an answer. Koved and Shneiderman (1986) believe the explicit menu condition results in a separation of menu item selection from the information of interest, leading to a possible loss of task context.

Scroll Bars. In many situations the user wishes to view information which cannot be entirely displayed within the boundaries of the current window. Scroll bars have been widely adopted as a solution to this problem. Generally placed along the right and bottom borders of a window (see Figure 2-2, page 10), scroll bars allow a user to move the window over its contents or move the contents within the window in four directions (up, down, to the left or to the right). Scroll bars provide the user with explicit control over the portion of information that will appear within the window. A square scroll box contained within each scroll bar provides feedback about the user's current horizontal and/or vertical position within the information space being viewed. At a glance, the user may obtain a visual indication of the quantity of information available above, below, to the left and to the right of the current window contents by visually estimating the position of the scroll box relative to each end of the scroll bar. Thus a scroll box appearing mid-way in a vertical scroll bar indicates that the information visible in the window occurs about half-way through the document. By including scroll boxes, scroll bars provide a mechanism to change the view of information in the window and context feedback about the relative location, in the information space, of the content being viewed.

Title. Most windows possess a title to identify the window contents. Titles may take the form of a coloured bar across the top, a tab attached to the top or side, or a label across the bottom of the window. For example, the Microsoft Windows environment uses the title bar to identify the window and to indicate its state. When the window becomes the listening (active) window, the colour of the title bar changes to a colour different from the title bars of all other windows on the display (*Microsoft Windows User's Guide for the Microsoft Windows Operating System*, 1992). In the Macintosh System 7 environment, the title bar shows horizontal stripes when a window is active. Titles provide recall cues to the user (the text string) which may facilitate identification of a window's function.

Colour and Texture. The careful use of colour can enhance the utility of window attributes already described. Colour can distinguish different visual components such as window borders, backgrounds, buttons and scroll bars. However, when colour is used to group menu items as well as to distinguish window components such as buttons, window borders and menus, incorrect associations among colour coded elements can result (Foley et al., 1990). Care must be taken to see that the colour coding scheme used for one application is not transferred, through the user, to another application which may possess a different coding scheme. In general, UI designers are cautioned to apply colour sparingly (Foley et al., 1990). Alternatively, button appearance and window regions can be highlighted through various shading techniques. For example, the Microsoft Windows interface gives buttons a three-dimensional appearance by applying a "drop shadow" (lighter shading to the top and left edge and a darker shading to the bottom and right edge). The proper application of colour and shading can enhance the appearance and usability of interfaces considerably by drawing upon lower-level human perceptual processing. Using these techniques, the identification of window components may become an automatic process.

Sound. The association of specific sounds--auditory icons--with various user-computer interactions is a relatively recent addition to the user interface (Gaver, 1989). In his description of the Apple Macintosh SonicFinder, Gaver (1989) considers ways that sound can augment the visual cues available in WIMP user interfaces. In particular, sound can highlight actions such as the opening and closing of windows by providing an auditory cue (a whooshing sound, for example) representing the progress of the activity. To effectively use auditory icons, the mapping of a sound to a specific action must be compatible, that is, in some sense the mapping must parallel an action/sound combination in the real world which is also familiar to the user. The potential benefits of sound as a source of context feedback are worth further investigation. As with colour, it seems likely that we should use sound sparingly to avoid overwhelming the user on yet another sensory dimension.

Animation. As an attribute of a window, animation can help redirect the attention of the user from one window to another. Probably the best example of window animation involves the opening and closing of windows in the Apple Macintosh (*Macintosh User's Guide*, 1992) and Hewlett-Packard NewWave (*NewWave User's Guide,* 1992) environments. Both of these GUIs use an animated expanding or shrinking border effect to show the transition from an icon view to the open window view and vice versa. Other animation effects such as flashing borders and title bars can focus the user's attention on a specific window. Animated icons, such as those described in Baecker, Small, and Mander (1991), also provide important state information to a user (e.g., progress indicators such as an hourglass with sand pouring through it or examples of an objects' behaviour). It is likely that animation will play a more important role in the future in helping the user integrate changing views of an information space displayed in a given window.

Tiled and Overlapped Windows. While computer systems have benefited from rapid improvements in processor speed, memory capacity and disc storage capacity, only marginal gains in display screen size have occurred. Large, high resolution display screens still command a fairly high price, relative to overall system cost. Two window presentation methods have evolved to maximize the use of available screen area and assist the user in regaining and maintaining task context. Tiled windows provide the user with access to several tasks in parallel by multiplexing the screen along one or two dimensions. In this way, all windows remain visible and are never obscured by another window. Overlapping windows provide access to several tasks by multiplexing the screen space over two-and-a-half dimensions (Card, Pavel, & Farrell, 1984). An overlapping window can partially or completely hide other windows from view. The user must initiate a specific action to bring an occluded window to the "top" so its entire contents can be viewed. Which window presentation method is most beneficial to the user? Do windows provide context information for the task at hand? The answer to both questions seems to be: it depends on the task.

As was mentioned earlier, the advantages and disadvantages of specific design options for these interfaces are still poorly understood. Clearly there are a number of window attributes at the disposal of the designer, who can combine them in many different ways to form a graphical user interface. To support the construction of a window-based UI, designers need some guiding principles.

In the next section I look at three different aspects of window system interaction. First, studies of user performance in different window environments provide a partial answer to the questions just posed. System designers do not often provide data supporting their choice of window presentation format, so it is useful to examine two studies that attempt to collect data on this aspect of system behaviour. From these analyses we can see if any important principles become apparent. Second, I examine some of the cognitive issues associated with screen display layouts and the temporal patterns of display presentation. Can we combine principles from window manipulation studies and cognitive issues in display presentation to produce new ways of navigating through an information space? In the third part of the next section I examine recently developed approaches to navigating through a large set of information, which brings the latest GUI facilities to bear on the problem of maintaining and regaining task context.

Window Systems in Action - The Language of the Interface

Only a small number of studies have examined the arrangement and management of multiple windows on display screens by the user of a computer system. Two of the better known investigations are discussed here. Bly and Rosenberg (1986) examined the performance of users carrying out tasks under tiled versus overlapping window organizations. The general belief, in the early 1980's, was that overlapping windows were preferable to tiled window arrangements. An analysis of window management issues suggested two classes of user requirements for window systems: (a) Windows which conform to their contents to maximize visibility (to display a complete form, for example), and (b) windows which relieve the user of size and location management (such as some types of pop-up menus). The hypothesis of interest was whether task performance was influenced by tiled or overlapping window arrangements. The researchers designed tasks which they believed would benefit from a specific window arrangement. The results indicated that the nature of the assigned task influenced which type of window arrangement was more effective, as expected. For tasks requiring few window manipulations, tiled windows resulted in quicker task completion. This task involved matching graphical objects arranged in a regular pattern in one window with text descriptions in another window display. In contrast, tasks requiring many window manipulations were handled more quickly with overlapping window configurations. This task involved matching graphical objects arranged in an irregular pattern in one window with text descriptions in another window display.

Subject experience with window systems also played an important role in these tasks. Subjects who regularly used window systems in their daily software development work performed considerably better on the tasks associated with the overlapped window environment, relative to inexperienced subjects. Bly and Rosenberg viewed their work as an initial foray into the topic of window organization, and suggested a need for further research on this issue. It is apparent that most of the window-based interface research of the day focused on general performance issues associated with the overall appearance and arrangement of information on the display, the number of commands used to perform a task and the frequency of specific command use.

A study by Gaylin (1986) sampled command use by experienced programmers with the goal of establishing a benchmark for window-based user interaction with a computer system. The benchmark tasks were intended to provide a basis for comparing user command activity across different window-based user interfaces. The system under study presented users with an overlapping window environment. Although the sample size was small (9 subjects) and the observation time per subject relatively short (approximately 22 minutes), window manipulation commands clearly dominated the user's interactions with the system. In particular, user activity seemed to focus on a small number of commands which were associated with moving between windows. Users would generally initiate some activity within a window--a program compilation, for example--then select a different window to perform another task until the compilation completed. Although overlapping window systems may have been the predominant window environment of the day, a more detailed investigation of tiled window management issues was also underway.

Cohen, Smith, and Iverson (1986) investigated the use of constraints to control the appearance and behaviour of windows in an environment called RTL/CRTL. The goal of this work was to enumerate and test a wide range of constraints placed on windows shown in a tiled arrangement on a display screen. A significant portion of the work concentrated on issues associated with interactions among constraints and how the system could reduce the need for user attention to window management activities. Implemented on top of the

Sapphire window system, RTL/CRTL applied five constraint categories to tiled window configurations including:

1) which windows must be present on the screen,
2) window size and location,
3) window adjacency and alignment requirements,
4) limits on how the system may automatically change the layout, and
5) particular organizations of windows on the screen.

The constraint category of primary interest here involves window layout. As Cohen, Smith and Iverson (1986) point out, the enlargement of one window could result in significant changes to the position and size of all other windows on the display. Repeated instances of this type of screen reorganization could quickly persuade users to abandon the system. To counter this problem, Cohen et al. (1986) implemented stability constraints which would prevent confusing reconfiguration of existing window size and position when window creation and destruction occurs. These stability constraints are further divided into four categories:

- global stability - the appearance of the display screen as a whole will not change,
- configuration stability - the adjacency relationships between windows will not change,
- appearance stability - the size and/or location of windows remain the same, and
- presence stability - that specifies which active windows will be closed to accommodate the opening new windows.

Cohen et al. (1986) did not provide any fundamental principles which might guide the assignment of constraint priorities to support a specific window manager policy. Nor did they discuss how such a policy might facilitate or hinder recovery of the user's task context. Assigning priorities to the different types of constraints would allow the window manager to implement a specific window presentation *policy,* which may be tuned to the users primary task. For example, one policy might emphasize maintaining window positions on the display while another policy might require window sizes to remain stable by allowing for the creation of overlapping windows. The goal would be to make the policy currently in force depend upon the user's task requirements. As noted earlier, Bly and Rosenberg (1986) discovered the linkage between task requirements and window presentation policy could be very simple. Having a mechanism for the implementation of a window presentation policy is useful, however, we require some guiding principles upon which we can construct a policy. To begin an evaluation of these issues, I now turn to a review of research examining the maintenance of task context when users must navigate among several screen displays or windows.

It is likely that stable screen displays play an important role in maintaining task context. In a survey paper discussing the cognitive issues associated with user navigation through large information spaces, Woods (1984) introduced the term *visual momentum* as " . . . a measure of the user's ability to extract and integrate information across displays, . . . ". In this context, Woods considers multiple-display systems to mean successive screen displays on a single physical display rather than simultaneous viewing of multiple physical display devices. With respect to hypertext and window environments, we might focus on visual momentum for the entire screen display or just within a given window on the display. Woods (1984) defines high visual momentum as continuity across successive screen displays supporting the rapid comprehension of data. In contrast, low visual momentum results in viewer confusion with each new screen display appearing independent of and discontinuous with the previous one. Four characteristics of a display result in a user interface supporting a high degree of visual momentum.

The long shot or overview display provides the user with a global map of the relationships among data chunks found in more detailed displays. The overview must explicitly incorporate relationships among display screens which are important to the user's task. Many systems now make use of overviews and browsers to serve as a source of context information.

Perceptual landmarks link successive views, helping the user integrate these images by providing easily discernible features across display screens. When a feature within a display screen is immediately recognized, it may direct subsequent looking behaviour (Woods, 1984), perhaps resulting in a similar eye movement pattern to that used on the previous display. Display overlap--used in the cartographic sense--means to physically overlap sections of previous display screens with peripheral areas of the current display screen. Fish-eye views provide a user with an image of this type, as would an animated transition from one display to the next. These features, taken together, provide support for a spatial representation of displays by establishing explicit relationships between successive views which may be interpreted by the user as itineraries or paths through the information space.

Finally, Woods (1984) discusses the concept of spatial cognition as the construction of analogical representations or maps of the user's task domain. Thus, overviews, perceptual landmarks, display overlap and a spatial representation of the displays are combined through the process of spatial cognition into maps of the task domain. The hypothesis is that by creating this cognitive map, the user is able to reduce mental workload and increase the transparency of the interface. Billingsley (1982) investigated the use of paper-based maps of a menu structure in a menu navigation task. She wished to test the hypothesis that maps would have a positive influence on menu navigation performance. Her results showed that users can locate target information on a series of displays faster and more efficiently using a map-based versus menu-based searching technique. When confronted with a search task in an unfamiliar part of the system, subjects showed stable map-based selection performance while menu-based performance dropped to novice levels.

Norman, Weldon, and Shneiderman (1986) also discussed window layouts and coordinating multiple screens to assist the user with information acquisition. They defined *visual scope* as " . . . the degree to which the user is able to integrate information across a display of multiple windows or screens and to grasp the whole of whatever is being displayed. . . . ". While multiple windows may increase the availability of information, if the user fails to see important relationships among the windows or is distracted by irrelevant or changing display detail, visual scope suffers.

Norman, Weldon and Shneiderman (1986) proposed three methods which invoke features of human visual perception to facilitate information transfer to the user. First, *spatial grouping*--the physical proximity, similarity of shape, size , colour, texture, or orientation--can be used to associate related information items. When attempting to group information across multiple windows or screen displays a compelling but careful use of these attributes is necessary. In particular, the inconsistent use of colour across different window displays would likely produce confusion. Second, the *temporal association* of changes to information in multiple windows can produce a powerful grouping effect. Linear information relationships across windows should be reflected in the order of screen updates to these windows. For example, a page previewer for a word processor which is displaying two adjacent pages should redraw the text pages in a left to right sequence. Third, *animated changes* to the form and content of windows or entire screen displays provide powerful cues to the perceptual system. Changes to perspective views of objects through smooth rotation can reduce user disorientation because the path from one view to the next is made explicit on the display.

Woods (1984) and Norman et al. (1986) could see the importance of tapping subtle aspects of cognitive psychology to improve user performance at the human-computer interface. We now see a growing interest in the application of more sophisticated metaphors and psychological principles to HCI design. In the following studies a novel twist is added to the desktop metaphor. This new metaphor is then kneaded and reformulated to produce yet another environment which draws upon the users' built-in perceptual processing systems thereby reducing their mental workload.

Rooms with a View

As users became more familiar with GUIs, the standard desktop metaphor seemed to place significant limitations on the number and complexity of window-based applications available and accessible to them at one time. Henderson and Card (1986) make the small screen dilemma abundantly clear by providing a few interesting comparisons: a standard office desk has the area of 22 original IBM PC screens or 46 original Macintosh screens; a dining table is equivalent to 57 original PC screens or 119 original Macintosh screens. Finally, a 19" Sun Workstation display is only one tenth the size of an office desktop. Clearly there is a spatial mismatch between the "desktop metaphor" and a real desktop. Henderson and Card looked at four possible techniques for dealing with the problem:

1) alternating screen usage - switching the entire contents of the screen,
2) distorted views - icons represent the extreme form of this approach,
3) large virtual workspaces - zooming and panning of a large virtual screen,
4) multiple virtual workspaces - the hierarchical aggregation of workspaces.

Henderson and Card settled on investigating the last of these; the design of a multiple virtual workspace environment.

The Rooms system supports the management of multiple windows associated with a task by assigning a set of windows to a virtual "room". Each room retains the window configuration as it was when the user was last "in" the room. Overview diagrams of available rooms provide reduced pictures of the each room's contents. Within a room, reduced rectangles represent the placement of windows and other objects. Henderson and Card examined four measures for defining a locality set of windows referenced by the user:

- simple observation - watching how users worked with groups of overlapping windows,
- inter-reference interval - the number of window references between two references to the same window,
- Denning's working set - the number of windows referenced in the last T references, and

- Madison's bounded locality interval - this measure groups window references over time to show not only the number of windows referenced but the temporal pattern of references.

These metrics are used to group together frequently referenced windows in a given room. By distributing related window sets to different rooms (virtual workspaces), screen space associated with a task is conserved and used to display windows likely to be referenced again in that context. Henderson and Card (1986) found that users of Rooms would leave more windows open and make them larger when these windows were clustered in different workspaces.

The Rooms system supports a number of additional design features including: overview sets of pictograms, doors, back doors and wiring diagrams to provide context information and facilitate user navigation among tasks. Through overview collections of *pictograms*--miniature representations of a given room--the user can see the current arrangement of windows in a number of rooms. Within a room, windows are part of an abstraction called a *placement*. Each placement represents a window, the position of the window and a set of window attributes (size and colour, for example). The same window can appear in more than one room, and in each case the window may have a different placement. *Doors* provide users with one-way access to other rooms. A *back door* in the current room gives users immediate access to the rooms from which they came. Thus, each pictogram can show a number of placements. *Wiring diagrams* illustrate specific connections between multiple rooms. The goal of the Rooms study was to examine key constraints which affect human performance in the use of computers. The issue of screen space being a scarce resource and the problem of the "messy electronic desktop", both components of the larger navigation problem, were addressed through the Rooms metaphor.

More recently, Card, Robertson, and Mackinlay (1991) and Robertson, Card, and Mackinlay (1993) enhanced the Rooms metaphor by using 3-D animation techniques in an attempt to lower the cognitive "cost" associated with finding information. These studies investigated four design strategies in pursuit of this goal: (a) making the user's immediate workspace larger, (b) providing user interaction with multiple agents, (c) applying low-cost, high performance graphics hardware and software to produce the visual displays, and, (d) using visual abstractions to shift information assimilation and retrieval to the human perceptual system. 3-D Rooms allow the user to "walk around" the virtual workspace and increase the density of accessible information through the use of animated 3-D information structures. Two of these animated structures, cone trees and the perspective wall, aid user's information interpretation by employing perceptual cues associated with the information display format.

Cone trees use a rotating 3-D structure to show relationships among large numbers of information items. Information is placed on individual "cards" which are connected in a hierarchy. All cards at a given level of the hierarchy are displayed in a 3-dimensional Rolodex-like arrangement. A particular path through the hierarchy is shown by rotating the appropriate card, at each level, to the front of the 3-D view. Thus, at each level, the user has a visual sense of the entire information structure. This technique has proven effective even when the displayed hierarchy contains thousands of items.

The perspective wall highlights the information of interest in a normal 2-D view while accommodating extreme distance relationships between information items to maintain global context. Previous techniques used overview and detail windows to provide context for a specific chunk of information. The perspective wall displays the relevant information mapped onto a flat surface in the foreground and attaches linearly related information onto perspective views of surfaces along each side of the foreground view. Using 3-dimensional animation, the user can move along the wall to examine information, with the area of interest always moving into full, 2-D view. Robertson, Card, and Mackinlay (1993) state that this method of information visualization increases screen space utilization by a factor of three and allows for smooth transitions of views, however, they did not provide any data on how these techniques affected user performance on information search tasks.

A notable component of recent user interface research is the growing role specific aspects of cognitive psychology, such as low-level human perception, play in HCI and information assimilation. In the sample of studies described here we observe progressively more sophisticated approaches to the analysis of window use, information presentation and user capabilities. In particular, we note that early work used *explicit* features of the interface (windows, menus, scroll bars) as tools to support users with task context and information space navigation. The newest HCI studies are shifting the emphasis from explicit features towards *implicit* features of the interface--perceptual cues and animation that do not require high-level cognitive activity--to enhance information location and transfer to the user via smooth transitions in task context. In the next chapter I examine recent studies of human memory which may prove useful to our understanding of human-computer communication, context and navigation in window-based computing environments.

CHAPTER THREE

Cognitive Psychology and Memory Research

In this chapter, relevant studies from the psychology literature on human memory that have influenced or may influence human-computer interface design are reviewed. Associated with these studies, several distinct types of memory or memory processes have been proposed by researchers. Some types such as short-term and long-term memory may be familiar to the reader. These two forms of memory, along with others which may be less familiar, are discussed in the sections that follow. In particular, recent studies of the differences between *explicit* and *implicit* memory are examined and a possible role for the findings in user interface design considered. Cognitive psychologists refer to memory which involves the conscious recollection of a previous learning episode as *explicit memory*. Alternatively, memory which is revealed through a change in task performance due to previous exposure to a set of information is referred to as *implicit memory*. A wide range of observations consistent with theories of distinct types of memory or memory processes support many researchers' arguments that this evidence reflects fundamental differences in how the brain operates.

In the first section, several of the distinctions in memory are presented with some emphasis on studies of amnesic patients that highlight the differences. In the second section, the differences between explicit and implicit memory are considered against the background of studies on normal adults. Tests of word fragment completion, picture presentation, and skill learning provide important data emphasizing the differences between explicit and implicit memory. The existence of these differences suggests aspects of both explicit and implicit memory that may also influence task performance in computer-based environments. The last section summarizes the characteristics of explicit and implicit memory and connects the results with user interface research issues.

The Many Faces of Memory

Short-term and Long-term Memory. In the context of computing, we think of memory tasks as ones which require a person to recall or recognize a specific item or event previously encountered--such as a list of menu-items or a specific command to request some action from the system. User interface designers quickly discovered that requiring users to remember many commands, or that commands with poorly constructed names would result in unsatisfactory task performance. It was apparent that some aspects of the psychology memory literature had something to offer human-computer interface design. For example, MacLean, Barnard and Hammond (1985) discussed the utility of recall tests as a measure of user performance in interactive systems. Many psychology studies have used different types of tasks to measure characteristics of memory. In one kind of experiment, subjects might be shown a list of words for a fixed period of time and then be asked to recall the words, in any order, without the aid of the list. This is known as a *free recall task.* In another type of experiment, subjects might be presented with a list of words to learn, grouped under different categories. Subjects are subsequently shown a category name and are asked to recall the words in that category. This is known as a *cued recall task.* As a final example, subjects might be shown a number of pictures. After a short interval of time some of the previously viewed pictures would be presented along with new pictures. The task would be to state ("yes" or "no") whether a given picture had been seen earlier. This would represent a *recognition task.*

Data from experiments such as those mentioned above provided the basis for the distinction between short-term or working memory and long-term memory, one of the most commonly held ideas associated with human memory today. Murdock (1974) presents a comprehensive review of the research examining this distinction. For example, using different types of free recall task, researchers identified memory performance effects such as *primacy* and *recency.* Primacy represents the greater ability to recall items which appear sequentially early in a list while recency represents the greater ability to recall the last several items sequentially present on a list. A plot of the serial position of items in a list by the proportion of items correctly recalled reveals a somewhat broad and flattened U-shaped curve. The existence of immediate free recall serial position functions, as they are known in the literature, provide researchers with a barometer of memory performance. Using clinical studies of patients suffering from severe memory deficits researchers have argued for separate types of memory (Hirst, 1982). For example, patients with Korsakov's syndrome--a disorder which leaves working memory and memories from the distant past intact--have great difficulty learning and retaining new information. These patients show normal recall test performance on the recency portion of the curve while producing virtually flat results in the primacy region for new material.

As a result of these and other studies on normal subjects, the concepts of working memory and long-term memory led researchers to ask questions about possible limits on the capacity of memory (Miller, 1956). A discussion of issues surrounding the research on memory capacity is beyond the scope of this thesis, however, the subject has had a significant influence on user interface design principles.[1] As experimental techniques improved and new data began appearing in the literature, new proposals for the organization of memory evolved.

Semantic and Episodic Memory. By the early 1970's other distinctions in the organization of memory appeared in the literature which were highlighted by studies of amnesic patients. Tulving (1983) provides an extensive review of this work from which some key results will be presented here. *Semantic memory*, according to Tulving (1983), represents memory for organized knowledge of the world. Another term which might describe this type of memory is categorical memory. Semantic memory would include, memory for facts, concepts, rules and associations. For example, the relationship defining an eagle as a type of bird would comprise

[1]Research on the characteristics of working memory had a significant influence on the development of guidelines for user interface design (Shneiderman, 1992, p. 73). The influential study by Miller (1956) examined data on working memory capacity using communication theory developed by Shannon and Weaver (1949). Miller found individuals could process five to nine "chunks" of information where a chunk may represent a digit, letter, word, or possibly a combination of these items recoded in some manner. A portion of the title of this study--"The Magical Number Seven, Plus or Minus Two"--has become a "rule of thumb" in user interface design.

an object in semantic memory. *Episodic memory* refers to events which have some relationship to time and the individual. Episodic memory would involve retention of information about events which occurred together or in some sequential order and were experienced by the individual. For example, the events which you recollect that occurred during your travel to work today would be drawn from episodic memory.

This distinction in type of memory or memory processes may help user interface designers understand the cognitive issues concerned with navigating through a multi-window information system. The high-level models created by designers to represent the structure of an information system would be the domain of semantic memory, while each step taken in the course of navigating through the information space would mark a time stamped event (a bookmark, if you will) in the episodic memory of the individual using the interface.

Evidence for the existence of semantic/episodic memory can be seen in studies of amnesic patients. The details of the following study will provide the reader with a sense of how the data are gathered. Warrington and Weiskrantz (1974) had severely amnesic patients and normal controls perform two different tasks using studied lists of 16 common words. Tests were carried out 10 minutes after the list was studied. One test was a standard yes/no recognition test which mixed previously studied words with new words. The second test, called a "cued recall" test by the authors, is now known as a word stem completion task. The task involved the presentation of the first three letters of each studied word with the request that subjects complete the word stem with a word from the previously studied list. Word stems were selected such that each matched at least ten common words in the dictionary. Since the amnesic patients generally did not remember having studied any list, they were asked to complete the word stem with any meaningful English word.

The results showed amnesic patients performed as well as normal controls on the semantic memory task (word stem completion--a test of knowledge) while amnesic's performance on the episodic task (word recognition--a test of recollection of a previous event) was well below that of controls. This research provided some support for the semantic/episodic distinction and was followed-up with another carefully controlled study which confirmed and extended these findings (Warrington & Weiskrantz, 1982). The interesting feature of the Warrington and Weiskrantz (1974) study was the normal performance of amnesics on the word stem completion test. Although these subjects did not recognize studied words, there appeared to be some type of transfer of information acquired during list study which benefited semantic memory. The nature of this effect has become an active area of research.

Explicit and Implicit Memory. The results presented above, along with subsequent studies (see Schacter, 1987 for a review), suggested to investigators a further distinction in memory might be appropriate. Graf and Schacter (1985) used the terms *explicit* and *implicit* memory to distinguish the psychological states associated with the two forms of memory. Specifically, people demonstrate explicit memory through the act of conscious recollection of information in well-known tests of free recall, cued recall, and recognition. Tests of explicit memory make clear reference to, and necessitate conscious recall of, a specific past learning event. Implicit memory is revealed through a change in task performance associated with exposure to some task-relevant information during a prior study event. Tests of implicit memory such as word stem completion, word fragment identification, and picture fragment identification make no reference to previously presented information (see Roediger, 1990 for a summary of implicit test characteristics). Given these definitions, it would seem that implicit and explicit memory represent two aspects of episodic memory as it was defined earlier.

If episodic memory retains a series of previously experienced events, the definitions above suggest that explicit memory may provide the memory system or processes to consciously access the events while implicit memory may make the events available through improved task performance. We can use this distinction to analyze hypermedia information systems: our explicit memories (short-term and long-term) may represent the history list of information pages or nodes visited in the course of navigating through the information space, while our implicit memories may be expressed by improved task performance the next time we choose to navigate the same or similar path through the information space. If differences between implicit and explicit memory represent fundamental properties of brain organization and/or function, it is likely that human-computer interface designers of hypermedia information systems will want to incorporate features compatible with these properties into their systems. Presenting a broad base of data which supports these distinctions in memory may encourage further investigations in the context of user interface design.

Research probing the extent of memory impairment in patients with amnesic syndrome is one piece of compelling evidence for the differences between implicit and explicit memory. Anterograde amnesic patients demonstrate relatively intact perceptual abilities, language and memories of their more distant past but appear incapable of retaining new information. While these patients perform normally on tasks which use working memory, they are extremely poor at tasks which require long-term retention of new material. Hirst (1982) catalogues the detailed characteristics of amnesic syndrome.

Work by Warrington and Weiskrantz (1970) found amnesic patients did retain new information based on tests in which presented words were degraded in appearance. Although they did not realize at the time that their results would lend support to theories of implicit and explicit memory, the experiment and results provides a clear example of the distinction. Amnesic patients were provided with lists of words to study and then carried out four different memory tests. The free recall test required the subjects to write down as many of the recently presented words as possible, in any order. The recognition test presented studied words intermixed with new words and subjects were asked to identify the previously studied words. Thus each test involved explicit recall of previously studied material. The third test involved the presentation of word fragments where subjects had to identify words which were severely degraded in appearance (portions of the letters within a word would be missing). The fourth test involved word stem completion where the first three letters of a word would be presented and subjects were requested to complete the word. The words used for the last two tests were a mixture of new words and words previously studied. The performance measure was retention score expressed as the percentage of correct responses.

The results showed that controls were significantly better than amnesics on the recall and recognition tests. However, there was no significant difference in performance on the tests involving degraded words and word stems. The significant finding of this study was that an aspect of the amnesic patients' memory remained intact since they were able to remember new material if the testing method provided some form of partial information different from that available in recognition tests.

The study just described provided some of the early evidence for what would later be viewed as a *priming effect*. The work of Warrington and Weiskrantz and others stimulated the development of new techniques and experiments to examine the priming effects, as we will see below.

The term *priming* refers to the improvement in performance measured when processing a test item, word, or picture by subjects that have been previously exposed to the test material (Cofer, 1967). In our discussion, priming effects refer to the observed performance enhancement while implicit memory refers to the type of memory or memory processes which may be tapped to produce the enhancement (Graf & Schacter, 1985). A number of studies (e.g., Graf, Squire, & Mandler, 1984; Graf & Schacter, 1985) have confirmed the intact performance of amnesic patients on implicit tests of word stem completion relative to the deficits observed in explicit tests of free recall, cued recall, and recognition.

Graf, Shimamura, and Squire (1985) examined priming effects in amnesics between visual and auditory study and test presentations. They presented amnesic and normal control subjects with an explicit recall test and an implicit word stem completion test. The results showed significant and comparable priming between amnesics and controls in the same modality (e.g., visual presentation-visual test) and different modality (e.g., auditory presentation-visual test) conditions for the implicit word stem completion test. The results also showed that the same modality condition produced a greater degree of priming for both amnesic and control subjects than did the different modality condition. In contrast, explicit recall tests showed no difference across study-test modalities but confirmed that normal controls perform significantly better than amnesic patients. These findings highlight the specificity of priming effects and suggest that implicit memory influences performance to a greater degree when study and test conditions are the same.

As a final piece of evidence for the fundamental nature of the differences between implicit and explicit memory, we turn to a study involving the acquisition of a skill over time. This study is of interest because a derivative of the procedure forms the basis for the experiment carried out as a part of this thesis. Nissen and Bullemer (1987) investigated the attentional requirements of learning through performance measures on a button pressing task. Subjects were asked to carry out a serial reaction time task where a light appeared at one of four locations on a video monitor. Subjects had to press one key, out of a set of four keys, that was directly below the position of the light. A specific 10-trial sequence of light positions was repeated 10 times to form a block of 100 trials. For each trial, the subject's reaction time and any errors in button selection were recorded. In their study of amnesic patients, Nissen and Bullemer wished to examine suggestions in the literature that amnesia disrupted memory for attended information. If conscious processing of the light-button pressing pattern is required for learning, then amnesic patients should perform poorly on the task relative to controls. The results indicate this is not the case. Amnesic patients learned the serial light pattern at the same rate as normal controls although amnesics showed slower reaction times. Thus over the course of 400 trials (four blocks), amnesic patients could learn a specific pattern of button presses as indicated by a decline in reaction time of approximately 150 ms. The same subjects showed little change in button pressing performance across four blocks when presented with a random light pattern. This result illustrated how these subjects could learn a skill with no conscious recollection of the pattern being present during training. Nissen and Bullemer's study extended the distinction between implicit and explicit memory systems to include the acquisition of a skill through the use of performance measures (reaction time and button selection errors).

Skill acquisition is an important aspect of human-computer interfaces involving a number of user interaction issues such as the nature of input device behaviour and use (Boritz, Booth, & Cowan, 1991; MacKenzie, Sellen, & Buxton, 1991). If skill-based aspects of the user interface are learned without user awareness, it will be important for user interface designers to understand the positive and negative consequences that different patterns of device and display interaction may ultimately have on task performance.

To summarize the studies described above, the following key issues emerge. First, by definition, amnesic patients show severe deficits on explicit tests of recall, cued recall and recognition relative to normal subjects. Second, implicit tests of memory produce the same levels of performance (priming) in both amnesic and normal subjects while explicit tests of memory produce poor performance in amnesic subjects relative to normal controls. These results provide strong support for the existence of underlying differences between explicit and implicit memory or memory processes. In the following section, studies of implicit memory in normal adults are presented to strengthen the assertion that this form of memory or memory process represents a fundamental characteristic of human behaviour.

Implicit Memory in Normal Adults. While evidence from amnesic patients accentuates the differences between implicit and explicit memory characteristics, we find that a variety of studies involving normal adults have also produced data which support this distinction.

- Measures of priming in normal subjects suggest that an implicit form of memory or memory process facilitates task performance even after long periods of time have elapsed.

- Manipulation of the level of cognitive processing[2] required to accomplish a task shows differences in performance between implicit and explicit tests.

---

[2]The levels of processing refer to a conceptual framework proposed by Craik and Lockhart (1972). This theory proposes that in conjunction with stages of perceptual processing where physical and sensory features of a stimulus are extracted, later stages may match these aggregated patterns with previously learned patterns. As this analysis proceeds to greater depths, more semantic and cognitive processing occurs leading to recognition which may trigger other associations based on past experience with the item (a word, picture, or sound for example).

- Priming effects are specific to the format of the material presented at study and test.

The following studies illustrate the items outlined above.

With his important investigation of reading skills, Kolers (1976) initiated a line of work that examined the perceptual aspects of memory when long periods of time elapsed between study and test. After subjects had learned to read, without error, pages of text printed in an inverted typography (upside-down and backwards) 13 to 15 months earlier, they were subsequently tested on reading speed and accuracy. Kolers found the subjects showed significant priming effects, yet they generally failed to recognize the content of the material at second reading. Jacoby and Dallas (1981) examined word identification performance of subjects who were briefly (for 1 sec) exposed to individual words at study. At test, subjects were required to report each word on the screen after it was flashed for a period of 30 to 35 ms. Subjects identified more words correctly when these words had been studied than if they were new. Performance on the identification task did not decline by a significant amount even when subjects were tested up to 24 hours later while performance on a recognition test for previously presented words showed a significant drop over the same time interval.

Craik and Lockhart (1972) proposed a framework for the operation of memory which argued that retention was subject to the amount of processing applied to acquired information. They suggested a number of levels of processing with the early levels involving perceptual aspects of the to be remembered item and successive levels dealing with greater degrees of semantic and cognitive structure and associations. Under this model, explicit recall and recognition test performance would be the result of deeper processing (semantic/cognitive levels) and implicit test identification performance would involve shallow processing (perceptual levels). Jacoby and Dallas (1981) examined these issues in an experiment which required subjects to answer two types of yes/no question about presented words during the study phase. One type of question was oriented to a semantic aspect of the word while the other type of question focused on the identification of a specific letter within the word. After study, subjects were given an implicit word identification test, identical to that described earlier, with the words being flashed on the screen for 30 to 35 ms. The subjects also were given an explicit word recognition test. The results showed significantly better explicit recognition performance in the semantic question condition than for the letter search question condition. In contrast, there was no difference in performance between semantic and letter question conditions on the implicit word identification test. As well, subjects identified significantly more words correctly when the words had been previously presented regardless of their response to the yes/no question.

Human-computer interface research has often focused on the semantic or cognitive processing a system may require of its users. Studies such as the ones just described indicate that we, as interface designers, are targeting just one aspect, albeit an important one, of memory or memory processing through the application of semantic and cognitive analyses. As will be seen below, the specificity of priming effects observed suggests we may need to pay attention to subtle differences in information presentation during the course of a session with an interactive system.

When trying to distinguish between two types of memory or memory processes, psychologists look for patterns in results from different tests called cross-over interactions. A cross-over interaction is evident when the results associated with different experimental conditions on an explicit test (recall or recognition) show the opposite pattern of results under the same set of conditions on an implicit test (word identification, word-fragment completion, or word stem completion). Weldon and Roediger (1987) provided subjects with a list of pictures and words for study. The pictures were line drawings of easily identified objects and the words referred to specific real objects. One subject group was given a free recall test where they attempted to recall, in any order, the study words and names of the studied drawings. The second group was given a word-fragment completion test. Word fragments such as, **`e_ep_an_`** would be presented and subjects were required to identify the word, **`elephant`**. Some of the fragments matched presented words, some matched the names of presented pictures and others corresponded to non-studied words or pictures. Subjects were asked to guess the word from the fragment. The pattern of results showed a cross-over interaction between the explicit recall test and the implicit word-fragment completion test. Free recall performance was better for pictures than words while prior study of words produced greater priming on a word fragment completion test than did prior study of pictures. Other studies such as those by Graf and Ryan (1990) have also demonstrated specificity of priming effects using words studied and tested in the same and different fonts.

The fact that subjects show better performance on tests of implicit memory when study and test conditions match suggests a possible basis for the concept of visual momentum proposed by Woods (1984) and discussed in chapter 2. According to Woods, visual momentum exists when there is a high degree of similarity in the visual appearance of successive display screens associated with a user interface. This similarity may correspond to providing information in the same format at study and test, thus we might consider the previous display screen as the "studied" item and the current display screen the "test" item. This similarity suggests that visual momentum could be measured using performance tests of implicit memory such as priming.

This section concludes with a more detailed description of a study discussed earlier when the amnesic data was reviewed. The description is important because it outlines a skill learning experiment upon which part of the research carried out for this thesis is based. Familiarity with this work will facilitate later discussion.

Nissen and Bullemer (1987) used variants of the serial reaction time task to investigate skill acquisition under single- and dual-task conditions. Subjects were asked to carry out a button selection task when a light appeared at one of four locations on a video monitor. For each trial, the subject's reaction time and any errors in button selection were recorded. The first experiment included two groups to characterize the learning of a light sequence under single task repeating and random conditions. The first group received 8 blocks of trials using a repeating 10-trial sequence of light positions. The second group received 8 blocks of trials using a random 10-trial sequence of light positions. The results showed an early and significant divergence in task performance between the two groups. The reaction times (RTs) for the repeating condition declined quickly over the first four blocks while RTs for the random condition remained fairly constant. Subjects in the repeating condition appeared to rapidly learn the pattern of light presentations with a high level of accuracy when their attention was focused on the task.

A second experiment examined the ability of subjects to learn a particular light pattern under conditions of divided attention. The companion task involved counting audible tones during each block. Using the same trial and block organization for a total of four blocks, three subject groups were tested. The first group was assigned to the single repeating light sequence condition identical to the first experiment. The second group was assigned to the dual-task repeating light sequence condition and the third group received the dual-task random light sequence condition. Subjects in the dual-task condition were required to perform the button pressing task and simultaneously count the number of low-pitched tones which occurred during a block. Upon completing four blocks under this training procedure, subjects were then transferred to the "generate" procedure for two more blocks. The generate procedure required subjects in each of the three groups to alter the button pressing task as follows. Instead of pressing the button under the displayed light, subjects were told to press the button at the position where they expected the next light to appear.

Nissen and Bullemer found that although subjects in the dual-random and dual-repeating conditions did not differ significantly in reaction time over the course of the first four blocks, the mean RT of the dual-repeating group tended to be *less* than the mean RT for the dual-random group during this training phase. As well, in the generate procedure the accuracy of button selection (measured as percent correct) during the first 10 trials of block 5 was 59% for the single-repeating condition, 32% for the dual-random condition and 47% for the dual-repeating condition. Accuracy measures of 33% represented chance performance. These data showed that learning under the dual-repeating condition resulted in one item above chance being correctly chosen. The interesting feature of these results was that upon questioning, subjects in the dual-repeating group could not recall noticing a repeating pattern of lights. While not as persuasive as the amnesic patient results, this experiment provided some evidence for the acquisition of a skill without awareness in normal adults.

A summary of the studies presented in this chapter highlights the following critical features of implicit memory in normal adults. There are measurable improvements in task performance when subjects have previously studied the test material even though they appear to have no conscious recollection of the material. These performance improvements can be long-term and are independent of the level of cognitive processing required for the task. The benefits to task performance appear to be very specific, with maximum priming occurring when conditions at study match conditions at test. Finally, there is an indication that some learning of a skill can occur under conditions of divided attention, suggesting an additional role for implicit memory.

Implicit Memory and the User Interface

This chapter presented several studies from the literature on implicit memory which may be relevant to user interface design. It is apparent that human performance on many tasks, including window-based computing tasks, must be subject to the characteristics of explicit memory, implicit memory and interactions between these memory systems or processes. Several distinctions in memory organization have provided human-computer interface researchers with a theoretical framework on which to base their designs. The concepts of working, long-term, episodic, and semantic memory have enabled designers to use memory characteristics such as chunking, recall, recognition, and the incorporation of conceptual knowledge to improve the quality of user interfaces.

Today we see the quantity and complexity of information accessible to users rapidly increasing along with the sophistication of information presentation techniques. The visual form of information presented to a user on a computer display can remain very stable over a long period of time or change dramatically, minute by minute. Evidence provided above suggests subtle attributes associated with the presentation of information items can influence subsequent performance on tasks involving those items. Consistency of visual information format and visual or auditory presentation may also have an important effect on user performance independent of the high-level cognitive models used to represent the structure of the information.

The purpose of this thesis is to suggest the first steps towards the application of basic research on implicit memory to user interface design and to window-based user interfaces, in particular. The goal of the experiment described in the subsequent chapters is to examine performance measures of memory on a question-answering task involving a simple multiple-window user interface.

CHAPTER FOUR

The Experiment - Method

Overview

This experiment involved the interactive presentation of text and graphical objects, combined in various ways, on a computer screen. Discussion of the objects and their relationship to each other in the experiment requires a clear and concise set of terms and definitions. Table 4-1 and the following overview of the experiment provide the necessary base upon which a description of the materials and procedures can be constructed.

This experiment consisted of 32 subject *sessions*. Each session included a *mouse practice phase*, a *target training phase* and a *recall test phase*, in that order. The mouse practice phase familiarized subjects with using the mouse and involved a variable number of *mouse menu-item selections* carried out by the subject until a criterion level of performance was reached. Mouse menu-item selections represented the clicking of the mouse button with the cursor positioned over a sensitive region of the screen, a *menu-item*. A menu-item was represented as a small bordered region of the screen (2-cm by 1-cm or 3-cm by 1-cm) surrounded by a thin black border, which may or may not contain a text label, one or more of which were located within a *window*. A window was represented as a larger region of the screen (11.5-cm by 7.5-cm or 10.5-cm by 7.0-cm) surrounded by a thin black border containing text and/or menu-items.

The target training phase was designed to expose subjects to different screen display patterns each requiring a specific sequence or a random sequence of mouse menu-item selections across 48 *target training blocks*. The goal was for a subject to unconsciously learn a specific pattern of menu-item mapping in association with a specific display geometry by the end of the phase. Each target training block of 5 *target training trials* was assigned a specific *display geometry* and a correct answer *menu-item mapping*. Each of four possible display geometries represented a combination of *window arrangement* and *text justification* and was

Table 4-1. Terms which describe the elements of the experiment.

Term	Definition	Components	
Question	A question derived from the board game *Trivial Pursuit™*.		
Menu-item	Six regions on the display which are sensitive to mouse button clicks and may appear labeled with one or two words of text.		
Window	A black rectangular border on the display which bounds a set of Menu-items and one Question.		
Trial	The display of a specific Window, with six Menu-items located at specific positions, and a specific Question. A trial includes the selection, by the subject, of one Menu-item labeled OK followed by the selection of one Menu-item from a choice of five within the same Window.	An OK Menu-item, a Question, and 5 answer Menu-items.	
Window Arrangement (WA)	The successive positioning of 5 Windows as they are drawn starting at the upper left corner and moving down to the right or starting at the upper right corner and moving down to the left of the computer screen.	Left Diagonal • • •	Right Diagonal • • •
Text Justification	The layout of the Question text resulting in a smooth border along either the left or right margin of the text.	Left Justified	Right Justified
Display Geometry (DG)	A specific pairing of Window Arrangement and Text Justification.	Left-Left, Left-Right, Right-Left, Right-Right	
Menu-item Mapping (MM)	The vertical position of the Menu-item, in the set of five Menu-items, which is labeled with the correct answer to the Question.	Fixed Mapping (FM)	Variable Mapping (VM)
Fixed Mapping (FM)	The correct answer was assigned to the same Menu-item position for a specific Trial within a Block.		
Variable Mapping (VM)	The correct answer was assigned to a randomly selected Menu-item position for a given Trial.		
Block	A set of 5 Trials using a specific Display Geometry and Menu-item Mapping	Display Geometry-FM	Display Geometry-VM

associated with a block as was one of two possible menu-item mapping patterns, *fixed mapping* or *variable mapping*. A window arrangement defined the visual pattern of windows as they were presented, starting at the upper left (left diagonal) or upper right (right diagonal) corner of the computer screen. A target training trial included a *question*, *labeled menu-items* and the selection of the correct labeled menu-item. Text justification specified the visual layout of the text question within a window and was either left justified or right justified. A fixed mapping pattern always placed the correct answer at the same menu-item position, out of five possible, and was uniquely specified for each of the 5 trials in a block. A variable mapping pattern placed the correct answer at a random menu-item position, out of five possible, for each of the 5 trials in a block. The measure of performance improvement across blocks was the time required to select the correct answer within each trial.

The recall test phase was designed to test a subject's ability to accurately recall the position of the correct answer chosen on a previously viewed target training trial and included 16 *recall test blocks*. Each recall test block of 5 *recall test trials* was assigned a specific display geometry and correct answer menu-item mapping. One of four possible display geometries representing combinations of window arrangement and text justification were associated with a block and matched the group assignment made in the previous target training phase (see Table 4-2). The same group matching applied to one of two possible menu-item mapping patterns, fixed mapping or variable mapping. A recall test trial included a question, blank menu-items and the selection of one blank menu-item. A fixed mapping pattern always placed the correct blank menu-item at the same menu-item position used during the target training phase, out of five possible, for a designated trial. A variable mapping pattern placed the correct blank menu-item at a random menu-item position, out of five possible, for a designated trial. The measure of recall performance was the proportion of menu-item selections which matched the correct menu-item selections from the corresponding target training trial. A control group of eight subjects received only the recall test phase of the

Table 4-2. Experimental design. The design had three within-subject factors: window arrangement (left-diagonal vs. right-diagonal), text justification (left vs. right), and menu-item mapping (fixed vs. variable). The levels of window arrangement and text justification were combined to specify the four display geometries shown across the top of the table. For each subject, one display geometry was administered under fixed- and the others under variable-mapping conditions.

	Display Geometry (DG)			
	Window Arrangement-Left Diagonal (L)		Window Arrangement-Right Diagonal (R)	
Menu-item Mapping (MM) Group	Text Justification-Left (L)	Text Justification-Right (R)	Text Justification-Left (L)	Text Justification-Right (R)
1	**FM***	VM	VM	VM
2	VM	**FM**	VM	VM
3	VM	VM	**FM**	VM
4	VM	VM	VM	**FM**

Target Training Phase	12 Blocks	12 Blocks	12 Blocks	12 Blocks
Recall Test Phase	4 Blocks	4 Blocks	4 Blocks	4 Blocks

*FM = Fixed Mapping and VM = Variable Mapping experiment to provide baseline measure of correct answer selection without the benefit of a prior training phase.

The previous section introduced a number of terms and definitions which together represent critical attributes of the experiment. The following sections expand upon specific items mentioned in the overview and provide a detailed description of the subjects, design, materials, experimental session, mouse practice phase, target training phase and recall test phase for this experiment.

Subjects

The experiment included 32 subjects; one was a volunteer from the community and 31 were University of British Columbia students from the Department of Psychology who participated for course credit. Subjects ranged from 18 to 41 years of age ($M = 22 \pm 5.2$ SD) and consisted of 10 men and 22 women. A separate control group consisted of eight volunteers who received $5.00 each for their participation in the recall test phase. Control subjects ranged from 18 to 24 years of age ($M = 20 \pm 2.1$ SD) and consisted of 5 men and 3 women.

Design

The design had three within-subject factors (see Table 4-2): window arrangement (left-diagonal vs. right-diagonal), text justification (left vs. right), and menu-item mapping (fixed vs. variable). The levels of window arrangement and text justification were combined to specify the four different display geometries shown across the top of Table 4-2. For each subject, one display geometry was administered under fixed- and the others under variable-mapping conditions, thereby ensuring that variable-mapping occurred 3 times as often as fixed-mapping. Across subjects, counterbalancing guaranteed that fixed-mapping occurred equally often with each display geometry.

Equipment

Subjects were trained and tested using a Hewlett-Packard Vectra 486DX microcomputer with 8 Mbytes of RAM and operating at a processor clock speed of 25 MHz. The Microsoft Windows version 3.1 graphical environment provided the underlying application programming interface support for the custom experiment control software written in C++. The control software presented the trial items on a Hewlett-Packard 17-inch colour monitor (model D1193A) connected to a Hewlett-Packard UltraVGA video adapter operating at a resolution of 1024 horizontal pixels by 768 vertical pixels (non-interlaced). The subjects sat approximately 50 to 60 cm from the colour monitor. Subjects interacted with the computer system with a Microsoft two-button bus mouse using their preferred hand. The control software only responded to clicks of the left mouse button during the experiment. The multimedia timer services (*Microsoft Windows 3.1 Multimedia Programmer's Guide*, 1992) included with the Microsoft Windows environment provided suitable timer event precision and accuracy for the subject's menu selection responses (see Appendix A).

Procedure

The subjects were tested individually in a session lasting about 45 minutes. A session had a mouse practice phase, a target training phase and a recall test phase. To begin the experiment, subjects provided information about gender, age and handedness, as well as about previous experience with microcomputers (yes or no) and mouse pointing devices (yes or no). The experiment was described as investigating whether window arrangement and text justification (i.e., display geometry) influence speed on a menu selection task. At the end of the experiment, subjects completed a four item questionnaire (see Appendix B) that probed their facility with the mouse and asked them to indicate any differences in task performance across the different display geometries.

Mouse Practice Phase. The purpose of the mouse practice phase was to familiarize and train subjects on a task similar to that used in the critical parts of the experiment. The task consisted of a series of trials. To begin each trial, the computer screen showed a *mouse practice window* (see Figure 4-1) -- a rectangular region 11.0-cm wide by 7.5-cm high, surrounded by a solid black border and containing 6 menu-items. This window was centred on the screen. One menu-item (2.5-cm wide by 1.0-cm high) was positioned 0.3-cm from the left border of the window and centred between the top and bottom borders. This item was labeled with the word **OK**. The five remaining menu-items were aligned in an evenly spaced vertical column along the right border of the window with 0.3-cm between the menu-items' right border and the window border. Each of these items was 3.0-cm wide by 1.0-cm high and all were initially blank. In addition, a 1.5-cm by 0.5-cm box centred and positioned 2.0-cm below the window was used to show reaction time results. The Asymetrix Toolbook system, running in the Microsoft Windows environment, was used to present and control the mouse practice phase.

The subject's task in this phase was to use the mouse to select the **OK** item and then select a target menu-item as quickly as possible. Each trial included the following events:

1. The subjects positioned the cursor over the **OK** menu-item on the left side of the mouse practice window and then clicked and released the left mouse button. This action changed the window; it caused the word **Target** to appear on one of the five menu-items (randomly selected) at the right of the window.

2. The task was to move the cursor as quickly as possible from the **OK** item to the target item (i.e., the menu-item with the word **Target**) and to click on that item.

3. The screen then reverted to its initial state as shown in Figure 4-1, to begin a new trial.

Subjects repeated this procedure for sets of 5 practice trials. The time between clicking **OK** and the target item was recorded (in ms) for each trial. At the end of each set of five trials the average time for these trials appeared in the box below the window, providing performance

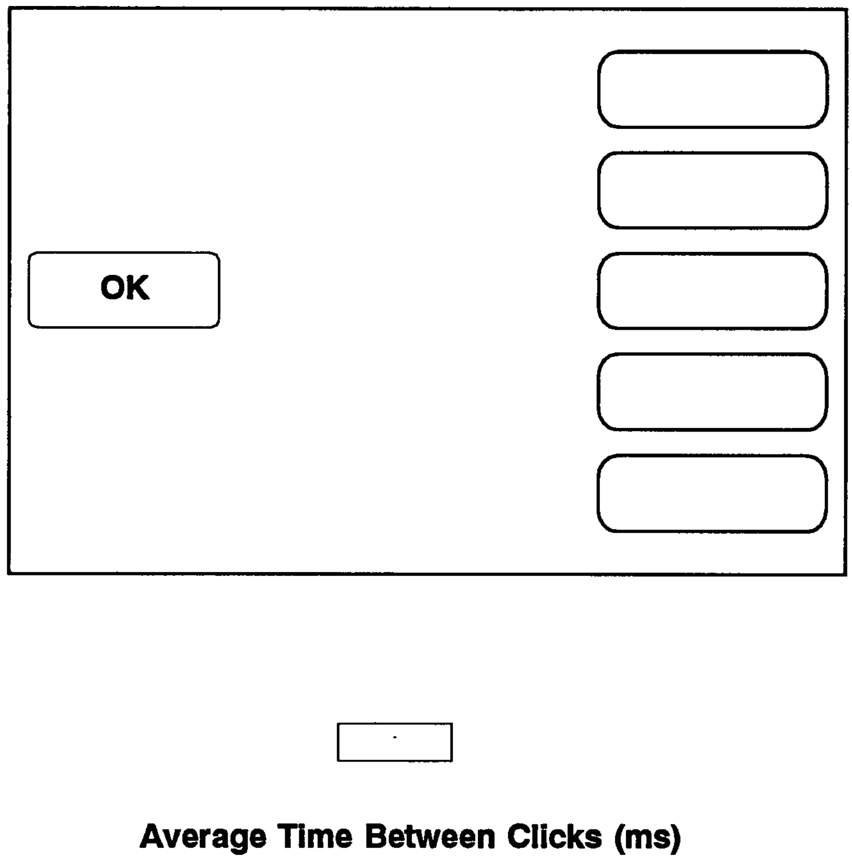

Figure 4-1. Mouse practice window at the start of the mouse practice phase of the experiment.

feedback. Subjects practiced the task until they reached one of two criteria (whichever came first): two consecutive five-practice trials averaging less than 700 ms or 10 minutes total practice time. These performance criteria were selected as a result of pilot testing. Eighty-one percent of the subjects reached the first criteria.

Target Training Phase. The purpose of this phase was to examine whether subjects' performance on a question answering task was influenced by the critical independent variables (menu-item mapping and display geometry). The phase had 48 blocks with 5 trials per block. Figure 4-2 shows the events associated with a single trial, and Figure 4-3 shows how trials were arranged to form different kinds of blocks. To begin each trial, the computer screen was blank, except for a 3.5-cm by 1.0-cm black rectangular border positioned in the middle, 2.0-cm from the bottom of the screen (see Figure 4-2). The rectangle was used to provide feedback about the subject's progress through the phase; at the end of each target training block, a red region inside the rectangle was updated by making its size, from left to right, proportional to the number of target training blocks completed.

A click of the left mouse button caused the appearance of the first window -- a rectangular region 10.5-cm wide by 7.0-cm high -- surrounded by a solid black border and containing 6 menu-items. The screen location of this first window differed depending on window arrangement: for left-diagonal displays, the first window appeared in the upper left corner of the screen (0.5-cm from the top and left screen boundary), while it appeared in the upper right corner (0.5-cm from the top and right screen boundary) for right-diagonal displays. Inside the window (see Figure 4-4), one menu-item (2.0-cm wide by 1.0-cm high) was positioned 0.3-cm from its left border, centred between the top and bottom borders. This item was labeled with the word **OK**. The five remaining menu-items were aligned in an evenly spaced vertical column along the right border of the window with 0.3-cm between the menu-items' right border and the window border. Each of these items was 3.0-cm wide by 1.0-cm high and all were initially blank. The text of a Trivial Pursuit question appeared in an unmarked region, 4.5-cm wide by 3.0-cm high, positioned between the **OK** menu-item and the 5 vertical menu-

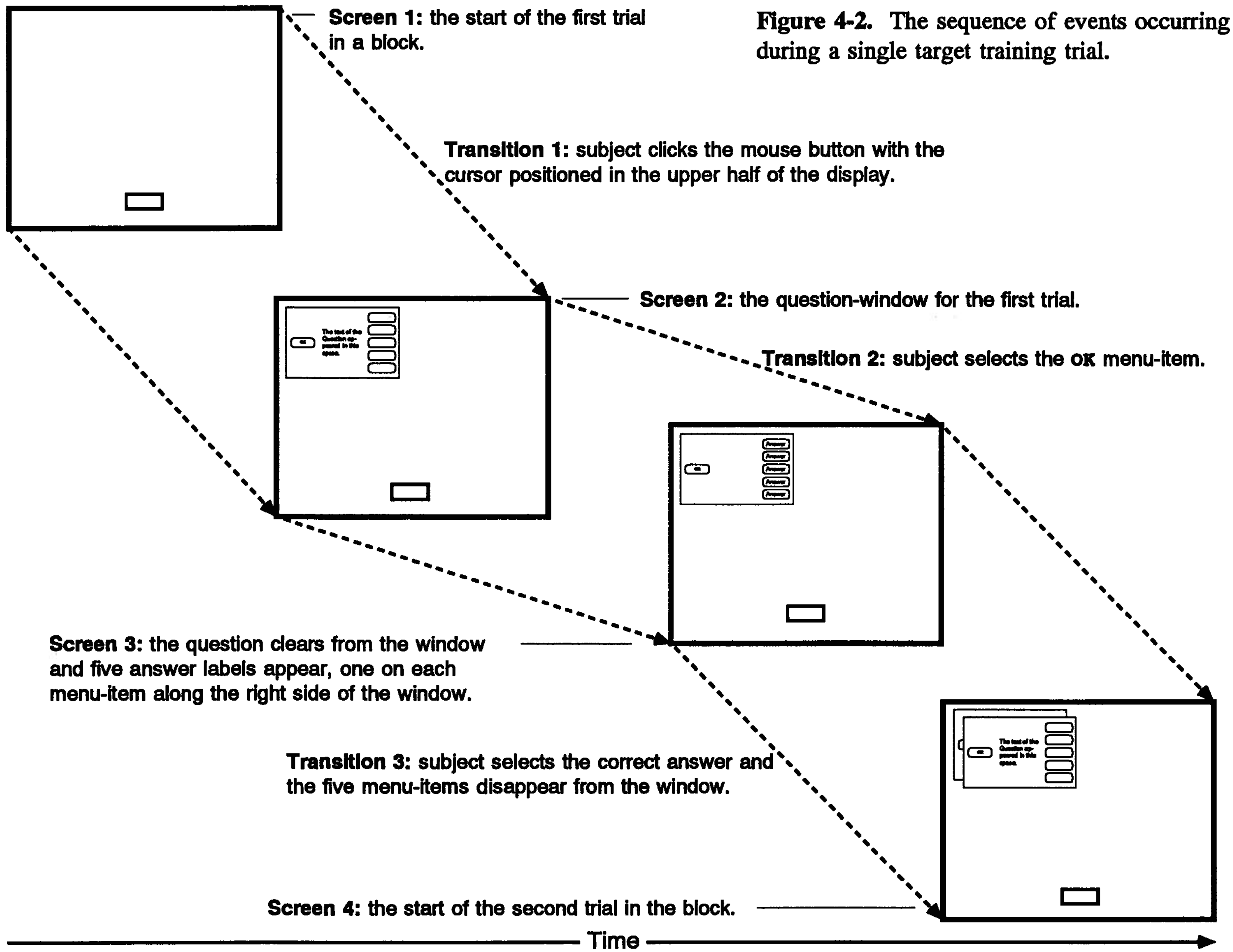

Figure 4-2. The sequence of events occurring during a single target training trial.

items, centred between the top and bottom window borders. The question text was either left- or right-justified as shown in Figure 4-3. A click on the OK menu-item changed the window as shown in Figure 4-2: the Trivial Pursuit question was removed and 5 answers appeared, one on each of the blank menu-items. The five answers were chosen to make the one correct answer obvious. For example, if a question involved the naming of a major city in the world, the correct city name would appear on one menu-item (e.g., Montreal) while the four remaining menu-items might be labeled with furniture objects such as desk, chair, lamp and stool. All text (Trivial Pursuit questions and answers) was written in black, in the Microsoft 12-point proportional System font. The window, text region, menu-items and the screen background were all the same light-grey colour with an RGB value of (0.75, 0.75, 0.75).

Figure 4-2 also summarizes the events of each target training block. The following steps are critical:

1. To begin the first trial of a block, the subject clicks the mouse button with the cursor positioned over the light-grey screen background, causing the display of a window with the 5 vertically arranged blank menu-items and a Trivial Pursuit question.

2. The subject reads the question. When ready to answer, the subject selects the **OK** menu-item, thereby causing the question to be removed and answers to appear within each of the blank menu-items.

3. The next step is to select the correct answer. Clicking on an incorrect answer has no effect; the screen remains the same until the correct answer menu-item is chosen. A click on the correct answer menu-item ends the trial and causes the appearance of a new window with a new question to be overlayed on the current window. At the end of the fifth trial, clicking the correct answer is followed by a

1 second pause (screen remains the same) and updating of the progress bar (bottom of the screen). Then the screen clears and reverts to the status described in Step 1.

Figure 4-3 illustrates the arrangement of the 5 target training trials constituting a block for each treatment condition or display geometry (left-left, left-right, right-left, right-right). The five trials comprising each block were cascaded diagonally either from the upper left or from the upper right corner of the screen. Each new trial window was offset from the previous window by 0.5-cm (see Figure 4-3). At the end of a block the display geometry remained visible for one second, after which it was removed from the screen. (Note: in order to reduce visual clutter within a display geometry and maintain the symmetrical appearance of left and right diagonal window arrangements, it was also necessary to remove the 5 menu-items and answer text of the current window prior to opening each new trial window.)

The subjects proceeded at their own pace through the target training phase. When ready, they clicked the left mouse button to initiate the first trial of a new block. Subjects were instructed to read the Trivial Pursuit question for this trial and think about a possible answer. When ready to answer, they selected the menu-item labeled **OK**, and then answered the question by moving the mouse cursor "as quickly as possible" to the correct answer which appeared on one of the five menu-items in the window. The subjects were told that the correct answer would be obvious and easy to find, that "one answer was obviously correct while the remaing four were obviously incorrect". Immediately after clicking the correct answer menu-item, the next trial began with a different Trivial Pursuit question, and the subject proceeded as in Trial 1.

The target training phase had 240 trials, with a different Trivial Pursuit question for each trial. The trials were arranged into 48 equal-sized blocks, with 12 blocks from each display geometry (left-left, left-right, right-left, right-right). Figure 4-5a shows how the 12 blocks from one display geometry were presented in the fixed-mapping condition, whereas Figure 4-5b shows how the remaining 36 blocks were presented in the variable-mapping condition for

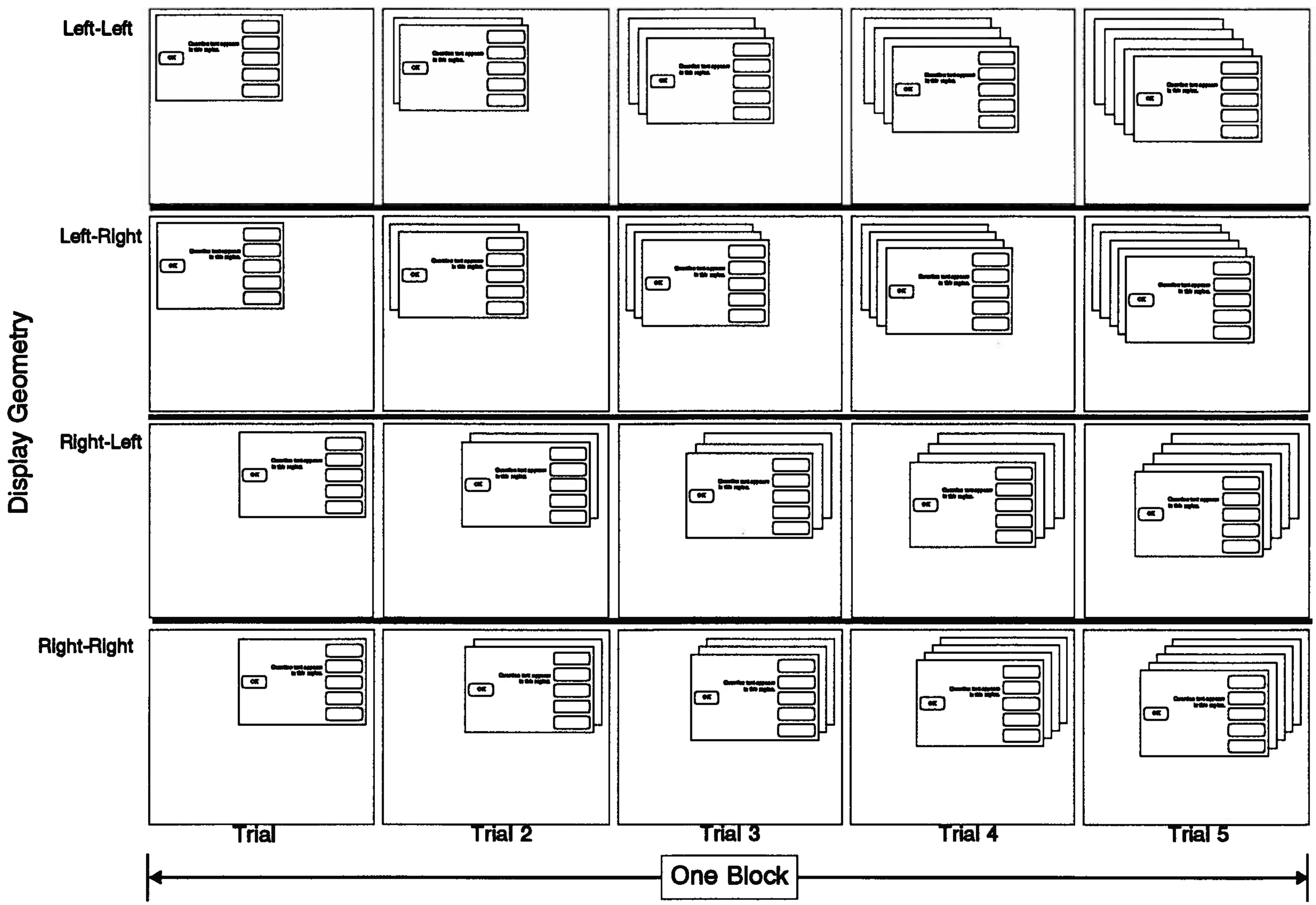

Figure 4-3. Five successive monitor screen images depicting each of the trials comprising a *block* for each of the display geometries. Left-left, left-right, right-left and right-right refer to a *window arrangement* (left diagonal/right diagonal)-*text justification* (left justified/right justified) pair, the attributes that determine *display geometry*.

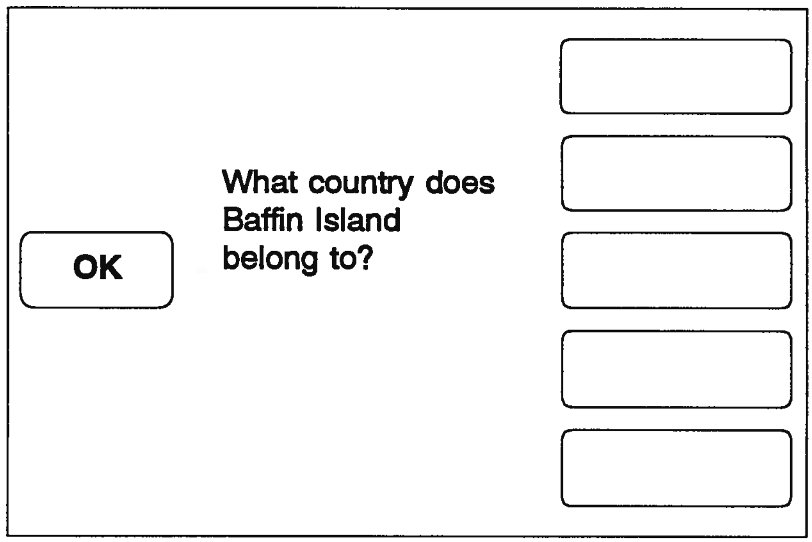

Figure 4-4. An example of a target training trial window prior to the selection of the **OK** menu-item.

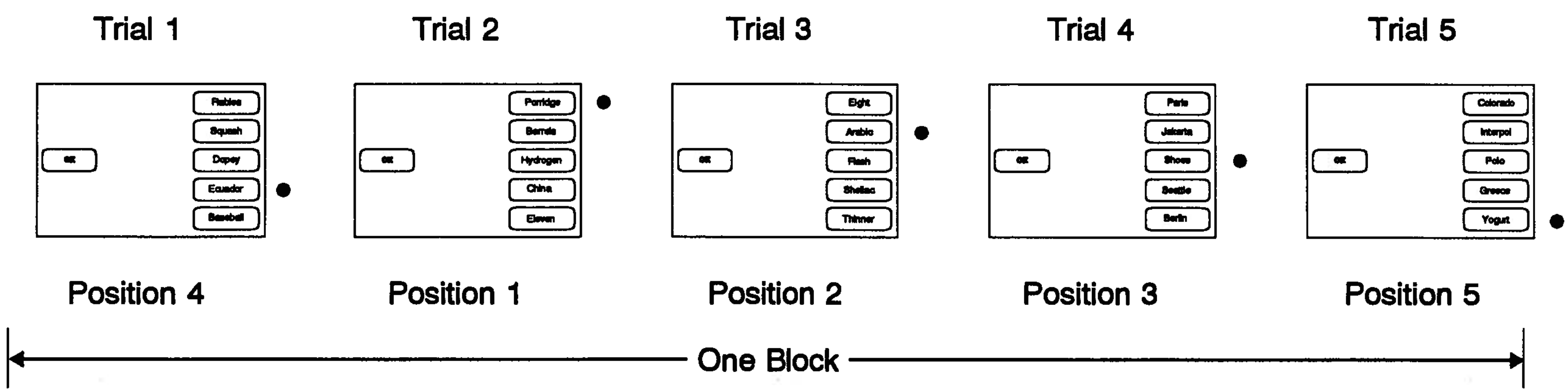

Figure 4-5a. An example of a fixed-mapping (FM) pattern of correct question answers to menu-item position within a trial block. The dot indicates the position of the correct menu-item (e.g., 4-1-2-3-5).

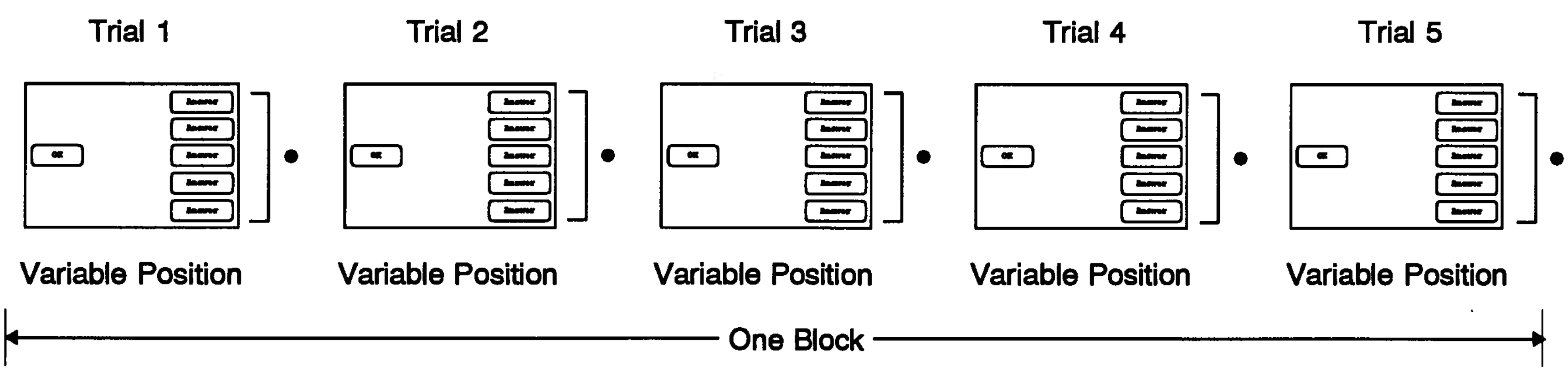

Figure 4-5b. Variable-mapping (VM) of the correct question answers to menu-item position within a trial block involved the random selection of the menu-item position, out of five possible positions, for each trial.

each subject. The specific blocks assigned to each mapping condition and how they were counterbalanced across subjects are shown in Table 4-2. The measure of performance improvement across blocks was the time required to select the correct answer after having selected the menu-item labeled **OK** within each trial.

In the fixed-mapping condition, the position of the menu-item (1, 2, 3..., from top to bottom) with the correct answer for each trial was the same for all blocks. That is, if the correct answers were on menu-items 4 - 1 - 2 - 3 - 5 for the first, second, third, fourth and fifth trial, respectively; these answer positions were the same (i.e., they were fixed) for all 12 blocks in this condition. Four different correct menu-item sequences were used in the fixed mapping condition: (sequence 1: 4 - 1 - 2 - 3 - 5; sequence 2: 3 - 4 - 5 - 2 - 1; sequence 3: 2 - 5 - 4 - 1 - 3; sequence 4: 5 - 3 - 1 - 4 - 2). Table 4-2 shows how sequences were assigned to the different display geometries in the experiment. An equal number of subjects was randomly assigned to each sequence condition.

In the variable-mapping condition, the position of the menu-items with the correct answers was randomly chosen on each trial, and thus, the correct answer position was unlikely to be the same (i.e. it was variable) across blocks. The critical question posed by the experiment is whether question answering would be facilitated by the associating a specific display geometry with a particular (i.e. fixed rather than variable) menu-item answer sequence.

Figure 4-6 shows how the fixed-mapping and variable-mapping blocks were distributed across the target training phase. The 48 blocks (12 fixed, 36 variable) were randomly selected for presentation, with one important exception: the 3rd, 7th, 11th, and every 4th succeeding block was randomly chosen from the blocks assigned to the fixed mapping condition. Thus, the fixed mapping blocks always occurred in the same position among the 48 blocks.

Recall Test Phase. The purpose of this phase was to examine whether subjects' performance on a recall task was influenced by the critical independent variables (menu-item mapping and display geometry). The phase had 16 blocks with 5 trials per block. Figure 4-6 shows the events associated with a single trial. Trials were arranged to form different kinds of

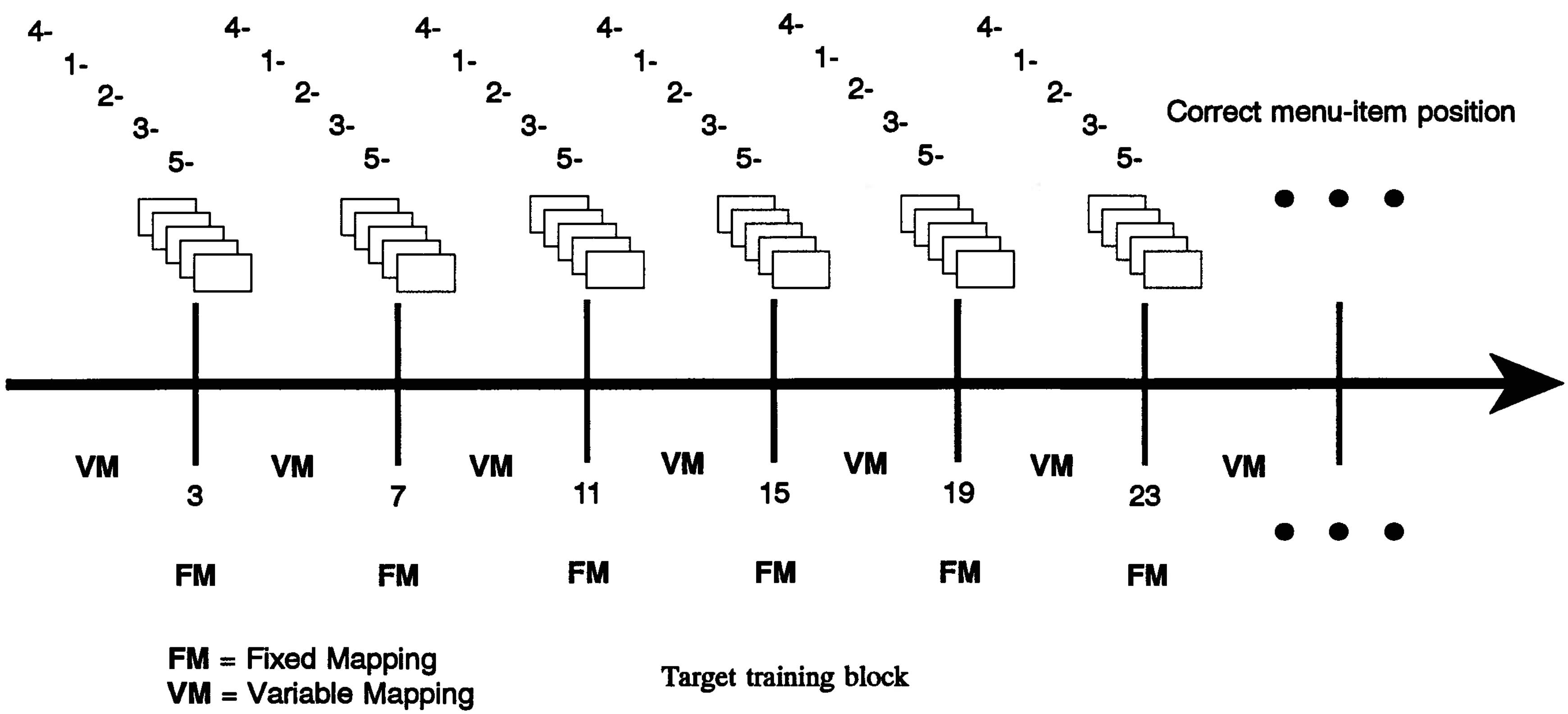

Figure 4-6. Presentation sequence of fixed-mapping and variable-mapping blocks during the target training phase. Fixed-mapping blocks occur at block 3, 7, 11, 15, etc. while variable-mapping blocks occur at blocks 1, 2, 4, 5, 6, 8, etc.

blocks using the same procedure followed in the target training phase (see Figure 4-3). To begin each trial, the computer screen was blank, except for a 3.5-cm by 1.0-cm black rectangular border positioned in the middle, 2.0-cm from the bottom of the screen (see Figure 4-7). The rectangle was used to provide feedback about the subject's progress through the phase; at the end of each target training block, a red region inside the rectangle was extended by making its size, from left to right, proportional to the number of recall test blocks completed.

A click of the left mouse button caused the appearance of the first trial (window) -- a rectangular region 10.5-cm wide by 7.0-cm high -- surrounded by a solid black border and containing 6 menu-items. The screen location of this first window differed depending on window arrangement: for left-diagonal displays, the first window appeared in the upper left corner of the screen (0.5-cm from the top and left screen boundary), while it appeared in the upper right corner (0.5-cm from the top and right screen boundary) for right-diagonal displays. Inside the window (see Figure 4-4), one menu-item (2.0-cm wide by 1.0-cm high) was positioned 0.3-cm from its left border, centred between the top and bottom borders. This item was labeled with the word **OK**. The five remaining menu-items were aligned in an evenly spaced vertical column along the right border of the window with 0.3-cm between the menu-items' right border and the window border. Each of these items was 3.0-cm wide by 1.0-cm high and all were blank. The text of a Trivial Pursuit question appeared in an unmarked region, 4.5-cm wide by 3.0-cm high, positioned between the **OK** menu-item and the 5 vertical menu-items, centred between the top and bottom window borders. The question text was either left- or right-justified as shown in Figure 4-3. Up to this point, the recall test trials were identical in appearance and procedure to the target training trials. A click on the OK menu-item changed the window as shown in Figure 4-7: the Trivial Pursuit question was removed and each of the 5 menu-items on the right remained blank. All text (Trivial Pursuit questions) was written in black, in the Microsoft 12-point proportional System font. The window, text

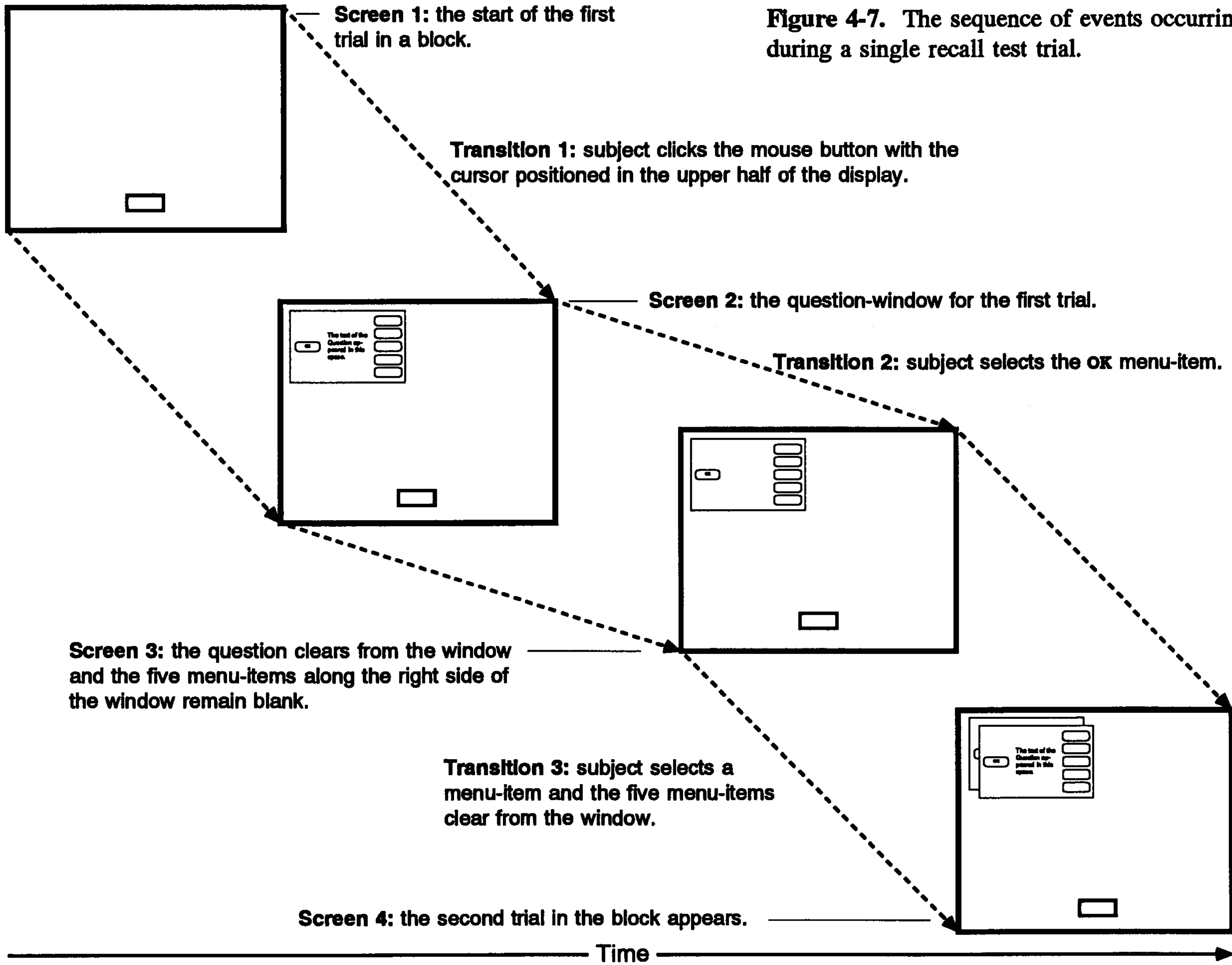

Figure 4-7. The sequence of events occurring during a single recall test trial.

region, menu-items and the screen background were all the same light-grey colour with an RGB value of (0.75, 0.75, 0.75).

Figure 4-7 also summarizes the events of each recall test block. The following steps are critical:

1. To begin the first trial of a block, the subject clicks the mouse button with the cursor positioned over the light-grey screen background, causing the display of a window with the 5 vertically arranged blank menu-items and a Trivial Pursuit question.

2. The subject reads the question. When ready to answer, the subject selects the **OK** menu-item, thereby causing the question to be removed and leaving each of the five menu-items on the right, blank. This is the first difference in procedure from that used in the target training phase.

3. The next step is to recall and then select the blank menu-item which contained the correct answer when the trial was viewed during the previous target training phase. Clicking on any one of the 5 menu-items ends this trial and causes the appearance of a trial (window) with a new question. This is the second difference in procedure from that used in the target training phase. At the end of the fifth trial, clicking any one of the 5 menu-items is followed by a 1 second pause (screen remains the same) and updating of the progress bar (bottom of the screen). Then the screen clears and reverts to the status described in Step 1.

Figure 4-3 illustrates the arrangement of the 5 target training trials constituting a block for each treatment condition or display geometry (left-left, left-right, right-left, right-right). The five trials comprising each block were cascaded diagonally either from the upper left or from the upper right corner of the screen. Each new trial window was offset from the previous window by 0.5-cm (see Figure 4-3). At the end of the block the display geometry remained visible for one second, after which it was removed from the screen. (Note: in the same manner as was done for the target training phase it was necessary to remove the 5 menu-items and answer text of the current window prior to opening each new trial window, to maintain the symmetrical appearance of left and right diagonal window arrangements within a display geometry.)

The subjects proceeded at their own pace through the recall test phase. When ready, they clicked the left mouse button to initiate the first trial of a new block. Subjects were instructed to read the Trivial Pursuit question for this trial and try to think back to when they saw this trial during the target training phase. When ready, they selected the menu-item labeled **OK**, and then selected the recalled menu-item out of the five blank menu-items in the window. Immediately after clicking a menu-item, the next trial began with a different Trivial Pursuit question, and the subject proceeded as in Trial 1.

The recall test phase had 80 trials, with a different Trivial Pursuit question for each trial. The trials were arranged into 16 equal-sized blocks, with 4 blocks from each display geometry (left-left, left-right, right-left, right-right). Four blocks were randomly drawn from the 12 fixed-mapping blocks previously viewed during the target training phase. Twelve blocks were randomly drawn from the 36 variable-mapping blocks previously viewed during the target training phase. Thus, subjects saw a subset of the previously viewed Trivial Pursuit questions, each time in the same context with respect to window arrangement, text justification and sequence of trials within a block.

In the recall test phase, the 16 blocks (4 fixed, 12 variable) were randomly selected for presentation, with one important exception: the 3rd, 7th, 11th, and 15th blocks were randomly chosen from the blocks assigned to the fixed mapping condition. Thus, the fixed mapping blocks always occurred in the same positions among the 16 blocks.

Recall Test Phase - Control Group. The purpose of the recall test phase control group was to provide a baseline measure of recall performance when subjects were not exposed to the target training phase. With the exception of omitting the target training phase, all other aspects of a session were identical to the description of the experiment provided above. That is, subjects received the mouse practice phase and the recall test phase. For the recall test phase, subjects were asked to read the question and select a blank menu-item as if they had previously viewed the trial and were attempting to recall the correct menu-item location. The measure of correct menu-item selection performance was the proportion of menu-item selections made by the subject which matched a randomly chosen menu-item for a given trial. The probability of selecting the correct menu-item was 1 out of 5 or 20% on each trial.

CHAPTER FIVE

The Experiment - Results

This thesis examined the influence of window and text display characteristics on subjects' menu-item selections through a formal experiment. In a window-based user interface, correct menu-items were linked in either a fixed or variable manner (menu-item mapping) to different display geometries (window arrangement and text justification). Each subject session consisted of three phases: a mouse practice phase, a target training phase and a recall test phase. The target training phase and the recall test phase contained of a sequence of blocks and each block included five trials. For each trial, the speed and accuracy of subject menu selections were recorded. The experiment was designed to investigate the following questions:

1) During the target training phase, would display geometry and/or menu-item mapping influence subjects' speed or accuracy on a question answering task?

   If display geometry affects performance, which of its component(s) are important: window arrangement, text justification or a combination of the two?

2) During the recall test phase, would the display geometry and/or the manner of mapping menu-items influence subject selection speed or accuracy?

   Would the number of trials (windows) visible on the display influence recall performance under different mapping conditions?

If display geometry affects recall performance, which of its component(s) are important: window arrangement, text justification or a combination of the two?

The results for the target training phase and the recall test phase are presented below. Results for the target training phase include menu-item selection times (ST) and menu-item selection accuracy. As well, an analysis of the influence of menu-item mapping and display geometry on STs and selection accuracy is reported. The analysis of menu-item selection speed and accuracy for the recall test phase focused on three independant variables: menu-item mapping (fixed or variable), display geometry (window arrangement and text justification), and trial number within a block. An examination of menu-item Accuracy associated with window arrangement and text justification conditions completed the analysis.

Target Training Phase

The target training phase was designed to expose subjects to different screen display patterns each requiring a sequence of mouse menu-item selections. The goal was for a subject to learn a specific pattern of mapping responses to specific menu-items that were consistently associated with a specific display geometry. Each training block of 5 trials had a specific display geometry, which was defined by a combination of window arrangement and text justification. For all subjects, the sequence of menu-items with the correct answer was pre-selected (it was fixed) for all blocks with a particular display geometry. In contrast, the sequence of menu-items with the correct answer was randomly selected and not associated with a particular geometry (it was variable) for the remaining blocks.

Menu-item Selection Times (ST). The critical dependent variable for this analysis was how fast subjects could select the correct menu-item, out of five possible items. The independent variables were the manner of mapping the correct answers to menu-items (fixed or variable), display geometry (a combination of window arrangement with text justification) and trial block number (1 to 48).

The first step in this analysis was to verify that the ST data conformed to commonly observed patterns associated with learning and memory tasks. A plot of the mean time to select a correct item for each block combining all conditions and subjects is shown in Figure 5-1. For this summary of findings, ST values greater than two standard deviations above the mean were excluded as outliers; less than 2.5% of the 7,680 ST values had to be excluded. The data shown in Figure 5-1 are well fit by a power function, $R^2 = 0.67$, $F(1, 46) = 94.64$, $p < .001$, which is common to other learning and memory tasks (Anderson & Schooler, 1991).

The following analyses were carried out to examine the influence of menu-item mapping conditions (fixed or variable) on the speed with which subjects made menu-item selections. If fixed-mapping made the menu-item selection task easier as the experiment progressed, mean STs for the fixed-mapping blocks were expected to decline. If subjects are able to acquire the pattern of menu-item selections associated with the fixed-mapping condition, and its linked geometry, we should see faster selection time on the fixed blocks relative to the variable-mapping blocks as the training phase progressed.

Figure 5-2 shows the mean STs for the 48 blocks during the target training phase. All STs reported here refer selection times for the correct menu-item choice. To investigate the overall influence of mapping condition on STs across blocks, a Repeated Measures Analysis of Variance (ANOVA) was carried out. Fixed-mapping block 15 was selected as the starting block for this analysis to exclude the initial stages of rapid learning which were not relevant to the effect we were interested in. As well, each set of three variable-mapping block STs occurring between fixed-mapping block STs were averaged for the analysis. A within-subjects comparison of STs by block indicated a significant main effect of menu-item mapping with variable-mapping STs greater than fixed-mapping STs [$F(1,31) = 9.52$, $p < .01$]. However, examination of the repeated measures ANOVA revealed the significant result was due to a single comparison between fixed-mapping block 23 and the average of variable-

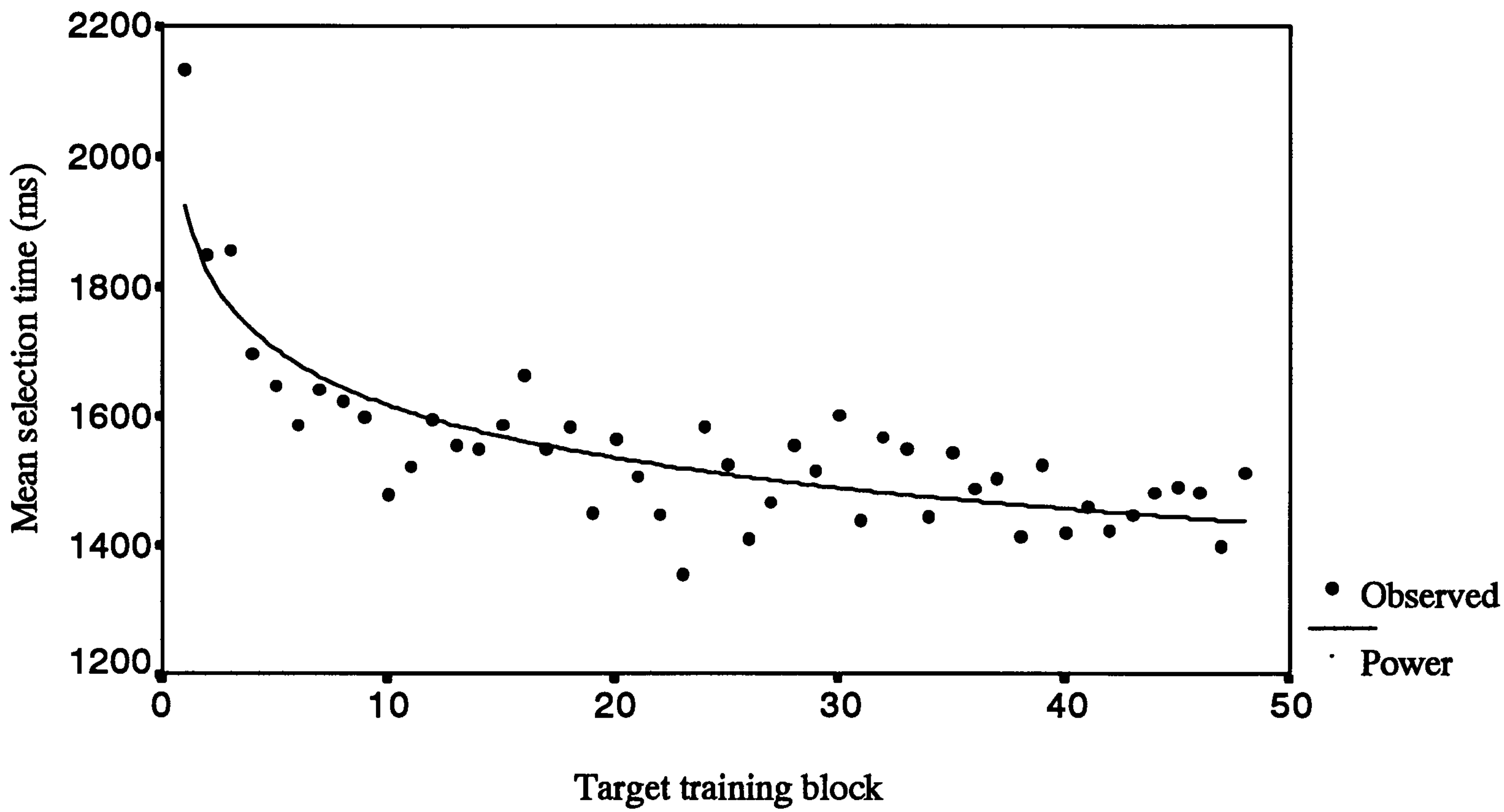

Figure 5-1. Power function fit to mean item selection time for all conditions combined across the target training phase (N = 32).

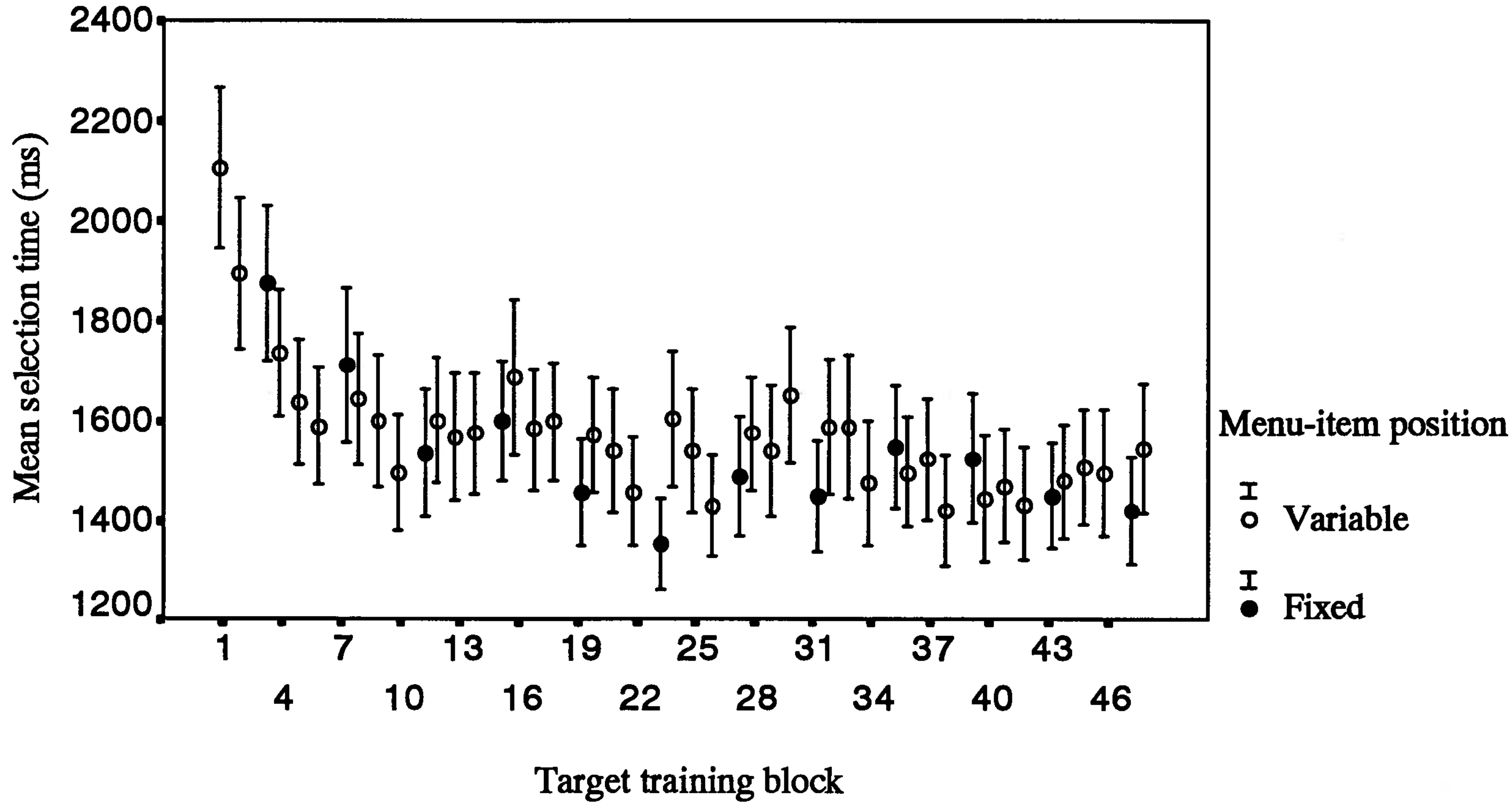

Figure 5-2. Mean selection time (ms) and 95% confidence limits during the target training phase.

mapping blocks 16, 17, and 18. While the fixed-mapping STs were expected to be lower than variable-mapping STs, the variation in STs, as shown in Figure 5-2, under both conditions suggests a mapping effect, if any, would require more than 12 fixed-mapping blocks to stabilize.

Selection Time and Display Geometry Conditions. Consistent with the lack of any overall influence of mapping conditions and the speed of menu-item selection, an analysis of the STs associated with each display geometry showed no systematic difference among the four conditions. In particular, the reduction in sample size associated with the analysis of display geometry appeared to increase the sample variance.

Menu-item Selection Accuracy. The critical dependent variable for this analysis was the number of correct menu-items selected out of five possible within a block of trials. The independent variables were menu-item mapping (fixed vs. variable) and display geometry (left-left, left-right, right-left and right-right). If fixed-mapping made the menu-item selection task easier as the experiment progressed, errors in selecting the correct menu-item for the fixed-mapping blocks should decline relative to variable-mapping blocks.

Menu-item Mapping Conditions. In general, few errors occurred in the selection of the correct menu-item during the target training phase in either the fixed- or variable-mapping conditions. Table 5-1 and Table 5-2 show that performance under the fixed-mapping condition was relatively stable and improved slightly as the target training phase progressed. For the fixed-mapping condition the mean number of correct menu-item selections across all blocks was 4.77 ± 0.66 *SD* (obs = 384), and very close to the mean number of correct selections under the variable-mapping condition [4.69 ± 0.80 *SD* (obs = 1152)] out of a maximum of 5 possible within a block. The high level of performance accuracy was expected since the task was designed to make the correct answers obvious to the subjects during this phase of the experiment. A Friedman Two-way Analysis of Variance (Siegel, 1956) showed the number of correct menu-item selections did not change significantly over the course of the target training phase in either the fixed- or variable-mapping conditions.

Table 5-1. The mean number of correct menu-item selections (max. 5) for each fixed-mapping block during the target training phase (N = 32).

Block	Mean *(SD)*
3	4.7 (0.7)
7	4.6 (1.0)
11	4.8 (0.7)
15	4.6 (0.9)
19	4.8 (0.7)
23	4.9 (0.4)
27	4.8 (0.5)
31	4.9 (0.5)
35	4.8 (0.5)
39	4.8 (0.9)
43	4.9 (0.4)
47	4.8 (0.6)

Table 5-2. Mean number and standard deviation of correct menu-item selections for each variable-mapping block during the target training phase (N = 32). Bold entries identify blocks immediately following fixed-mapping blocks.

Block	Mean (*SD*)
1	4.3 (1.2)
2	4.3 (1.3)
4	**4.6 (1.0)**
5	4.6 (0.8)
6	4.9 (0.3)
8	**4.7 (0.6)**
9	4.7 (0.6)
10	4.7 (0.9)
12	**4.8 (0.7)**
13	4.6 (0.9)
14	4.7 (0.9)
16	**4.5 (1.2)**
17	4.7 (0.8)
18	4.8 (0.8)
20	**4.8 (0.4)**
21	4.8 (0.8)
22	4.7 (0.8)
24	**4.8 (0.5)**
25	4.6 (0.8)
26	4.8 (0.6)
28	**4.7 (0.8)**
29	4.6 (0.9)
30	4.6 (1.0)
32	**4.7 (0.7)**
33	4.6 (1.0)
34	4.9 (0.6)
36	**4.7 (0.9)**
37	4.7 (0.8)
38	4.8 (0.5)
40	**4.7 (0.8)**
41	4.9 (0.3)
42	4.9 (0.4)
44	**4.9 (0.4)**
45	4.8 (0.7)
46	4.7 (0.5)
48	**4.5 (1.1)**

Selection Accuracy and Display Geometry Conditions. An examination of correct menu-item selections did not reveal a significant change in performance across blocks for display geometry (window arrangement or text justification). Again, ceiling effects precluded the need for further statistical analysis.

Recall Test Phase

For the recall test phase, subjects were presented with a series of trials and blocks that used the same materials as in the target training phase. For each test trial, subjects were asked to think back to the target training phase trial and to recall and select the correct menu-item out of the five blank menu-items provided. Subjects were told they could take as long as they wished to make each selection. The time actually used by subjects to make their selections during this phase provides useful information for the interpretation of the results described in the remaining sections. Figure 5-3 shows the mean selection times and 95% confidence limits across the 16 recall test blocks. Overall, subjects completed the identification and selection of a menu-item for a given trial in just under one second across the 16 recall test phase blocks. Choice time values greater than two standard deviations above the mean were excluded as outliers from Figure 5-3; they represented less than 2% of the 2560 ST values. This adjustment resulted in a mean ST of 931 ms and standard deviation of 246 ms (*obs*= 2509 trials).

Menu-item Selection Accuracy. The critical dependent variable for this analysis was the proportion of correct menu-items recalled within a block of five trials. The independent variables were menu-item mapping conditions (fixed vs. variable), display geometry (left-left, left-right, right-left and right-right), and the number of trials visible on the display. If training under fixed-mapping conditions aids correct menu-item selection regardless of display geometry, subjects should show higher performance across all fixed-mapping recall test blocks. If a particular display geometry provides additional cues for menu-item recall, subjects should correctly select more menu-items when this geometry is presented. If the

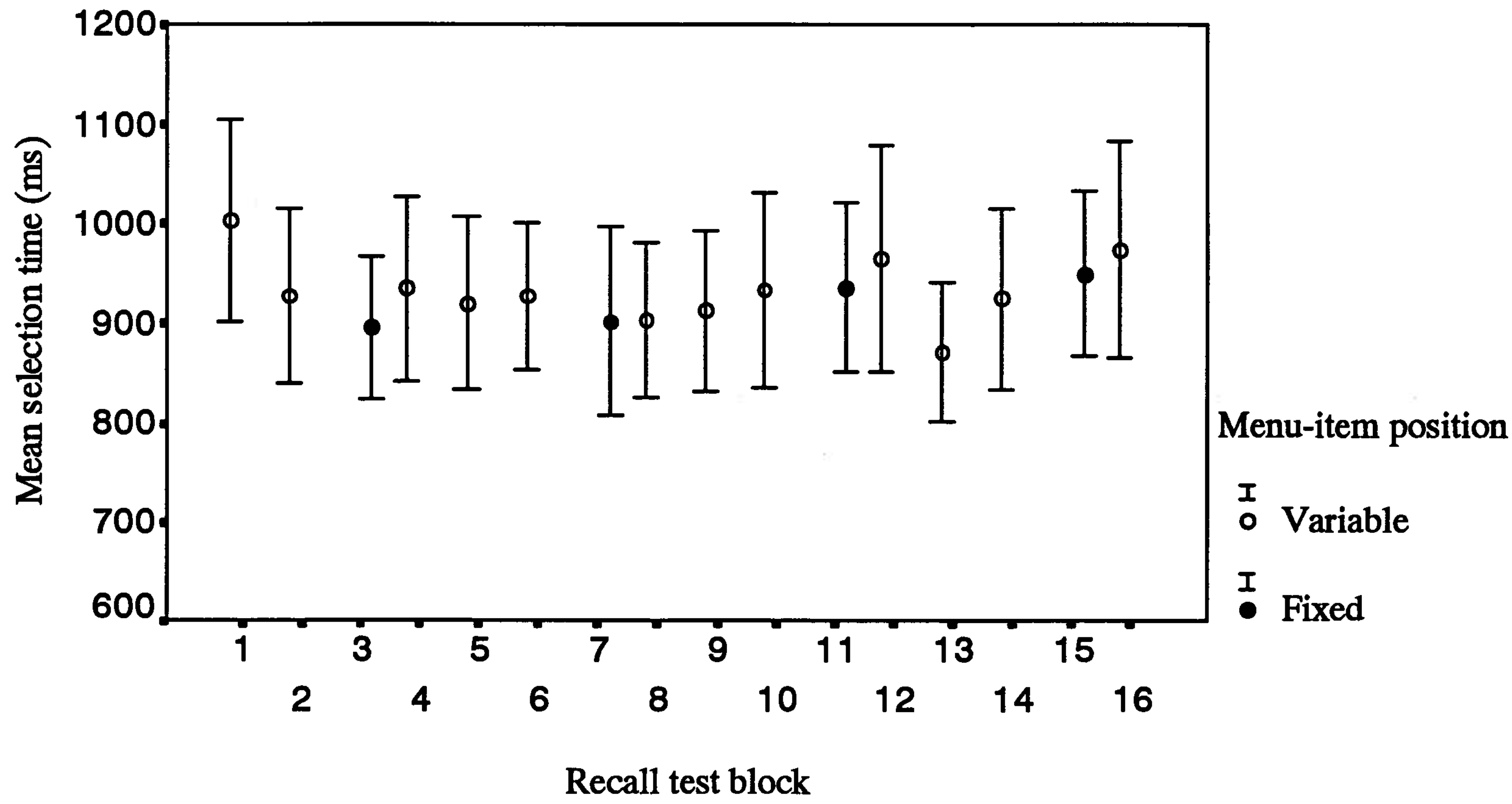

Figure 5-3. Mean selection time (ms) and 95% confidence limits during the recall test phase (N = 32).

amount of information visible on the screen influences correct menu-item selection, then as the number of trials displayed during a block increases, errors in menu-item selection should decline.

For the following analyses, the five answer menu-items are each identified by a number, starting with one (1) for the top-most item on a trial, through to five (5) for the bottom-most item on a trial. These numbers will be referred to as *menu-item numbers*. A cross-tabulation of correct menu-item numbers and selected menu-item numbers for each trial was examined for all blocks combined.

Menu-item Mapping and Selection Accuracy. Subjects showed no difference in the accuracy of selecting menu-items under fixed- and variable-mapping conditions. The overall mean proportion of correct menu-items selected was 0.41 with a standard deviation of 0.08 (N = 32). Figure 5-4 shows a plot of the mean proportion of correct menu-item selections across the 16 test blocks. The proportion of correct responses was relatively stable across blocks with a level of performance for fixed-mapping blocks 7 and 11 which was higher than any of the variable-mapping blocks. A slightly higher proportion of items were correctly selected in the fixed-mapping condition [fixed: $M = 0.43 \pm 0.18$ *SD* ($N = 32$), than in the variable: $M =$ 0.41, $\pm$ 0.08 *SD* ($N = 32$)] but the difference was not significant by the Mann-Whitney U test [$U = 23110$, $z = -1.05$, $p = .30$].

To establish baseline performance for the recall test, the menu-item selection scores for the 32 experimental subjects were compared with the scores for 8 control subjects who were not exposed to the training phase. Figure 5-5 shows the performance of experimental and control subjects for the 16 recall test blocks combined. The experimentals performed significantly better than the controls [unequal variance t-test: $t = 7.18$, $df = 26$, $p < .001$] with experimentals showing a mean of 0.41 $\pm$ 0.11 *SD* proportion correct (N = 32) or 41% while controls produced a mean of 0.21 $\pm$ 0.05 *SD* proportion correct (N = 8) or 21%. Thus, controls were at chance level of performance (20%) while experimentals were at twice chance level.

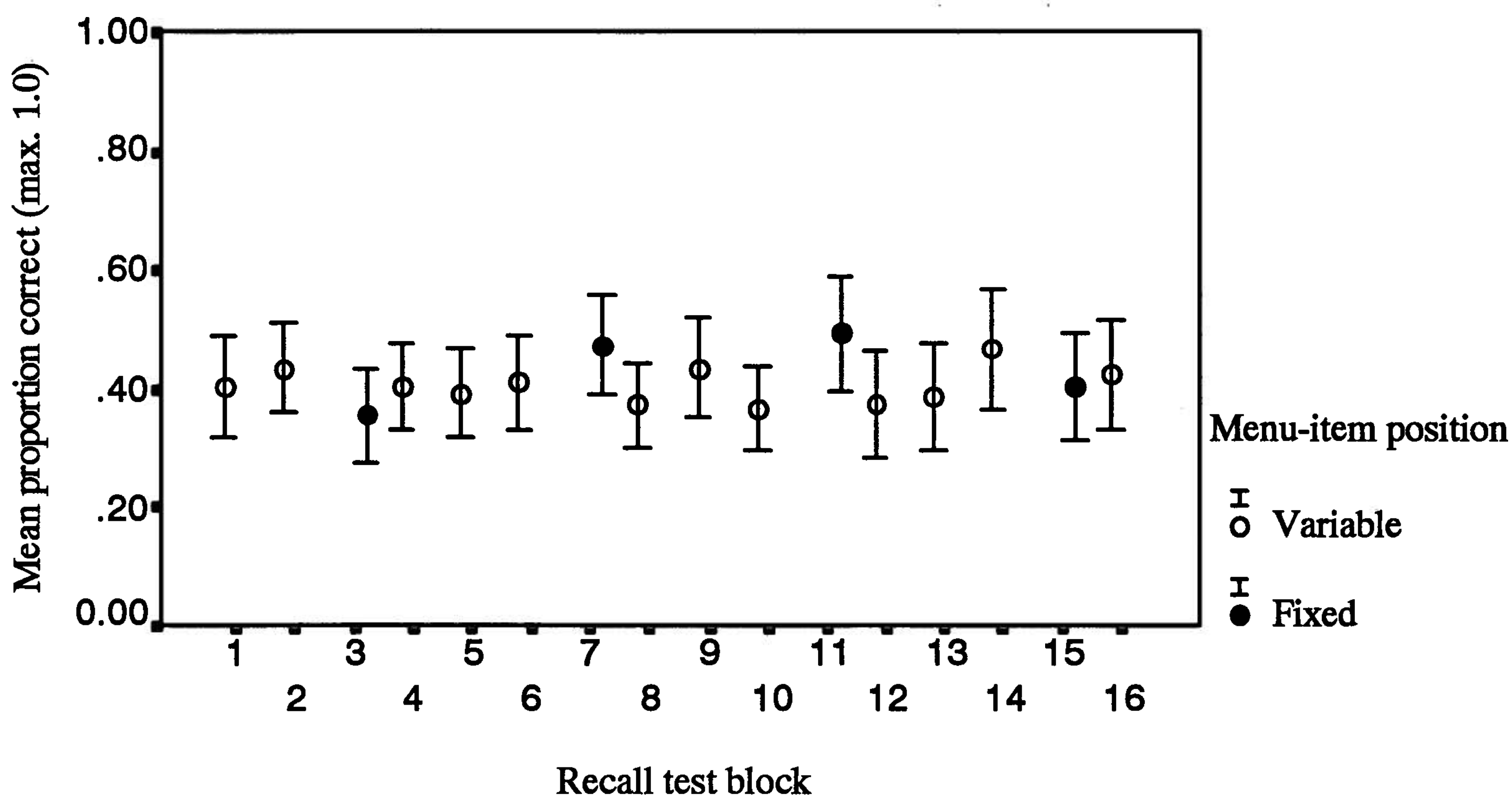

Figure 5-4. Mean proportion of correct menu-item selections and 95% confidence limits during the recall test phase (N = 32).

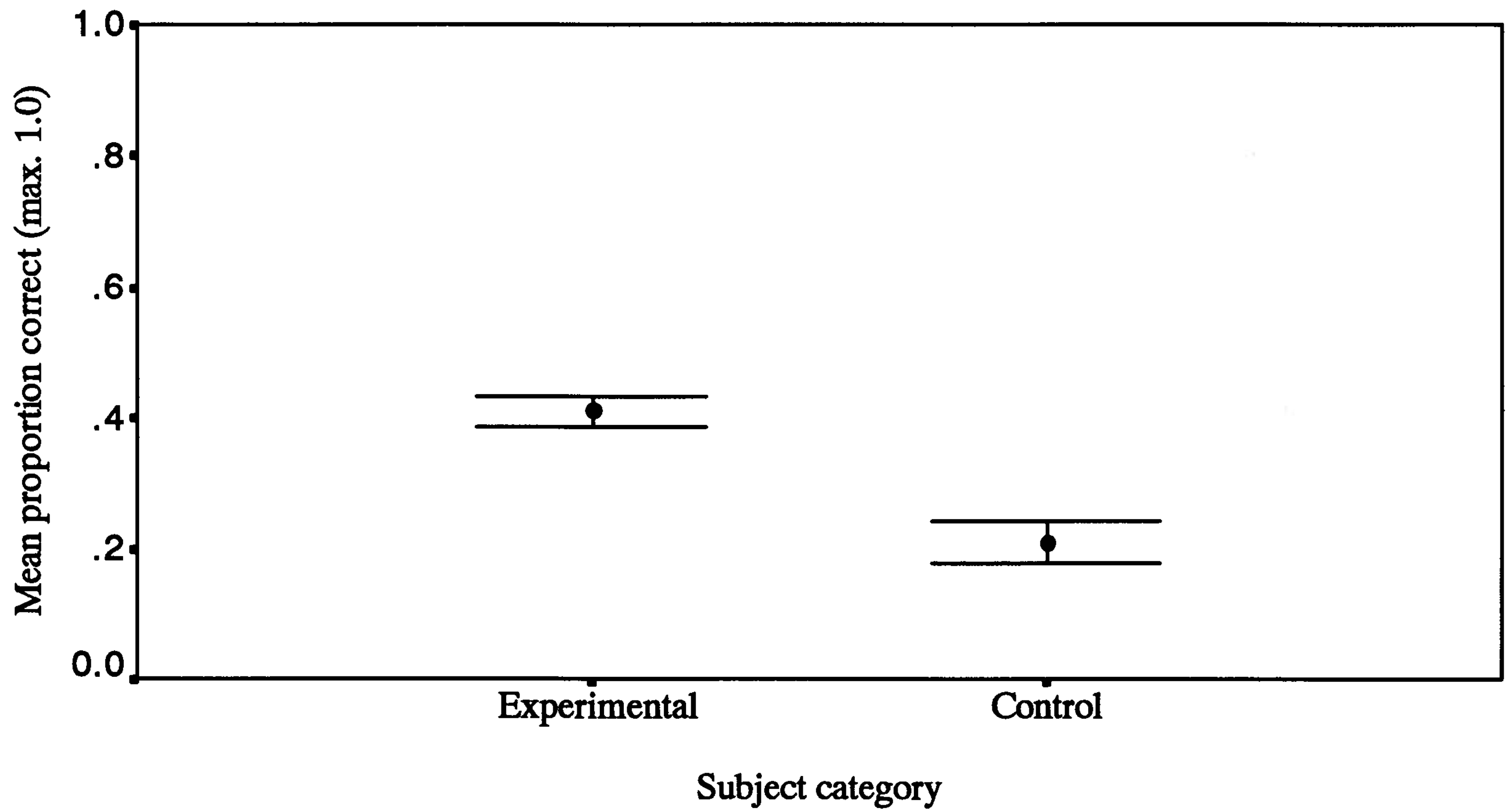

Figure 5-5. Experimental and control menu-item selection accuracy and 95% confidence limits; experimental (N = 32), control (N = 8).

If adding more information to the display aids menu-item selection, we should observe an increase in the proportion of correctly selected menu-items as trials progress. Figure 5-6 shows the performance of experimental and control subjects for each trial within blocks. No difference in the proportion of correct menu-item selections was found across blocks for the fixed-mapping or variable-mapping conditions. As well, there was no change in menu-item selection accuracy across trials for control subjects.

Distribution of Menu-Item Selection Errors. The previous analysis shows that the proportion of correct menu-item selections did not change across trials. However, it is possible that the distribution of selection errors could change within each block with no change in the proportion of correct selections. If additional display information influences the pattern of selection errors, earlier trials might result in less focused menu-item selection. As additional trials are presented, selection patterns could change causing item selection to be drawn nearer the correct item. On the assumption that menu-item recall is constrained, in part, by information on the display screen, there is clearly more information available with each succeeding trial, thereby predicting an improvement in performance across trials.

To examine the distribution of subject responses across recall trials, the menu-item numbers of correct menu-items were cross-tabulated with the menu-item numbers of selected items for each trial, for all blocks combined. The resulting 5 by 5 table contained the cell frequencies of all possible correct and selected menu-item combinations. These tables were constructed for each of the five trials presented at recall across all blocks. For each trial and each correct menu-item number, the cells of that row were examined and the menu-item numbers for the two highest cell frequencies recorded. In the case of ties, the menu-item numbers for all tied cell frequencies meeting the previous criteria were included in the analysis. For all experimental subjects, the sum of the two highest cell frequencies represented, on average, over 68% of the 512 menu-item selections associated with each trial across all blocks.

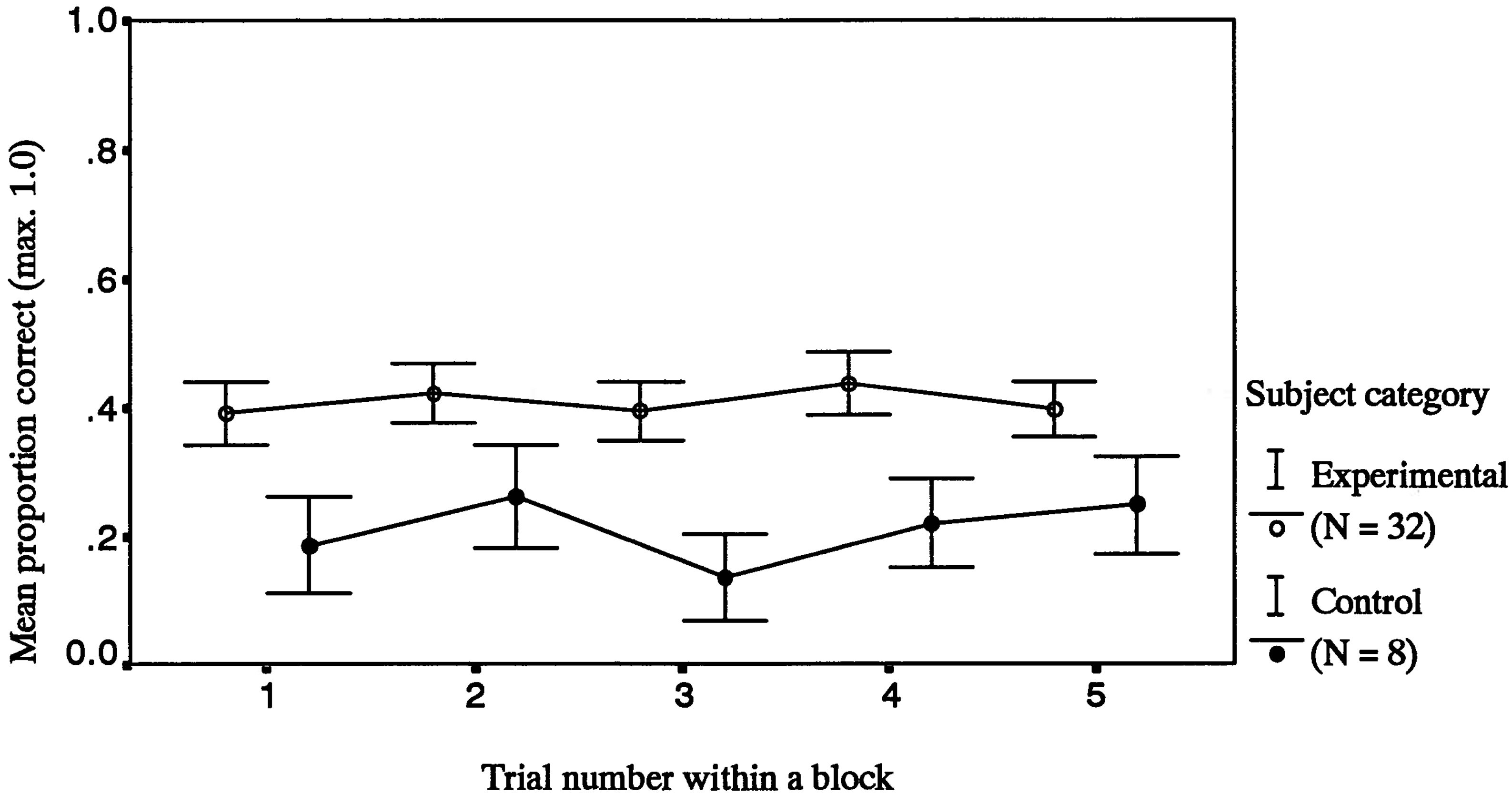

Figure 5-6. Experimental and control subjects. Correct menu-item selections, by trial number, and 95% confidence limits.

Figures 5-7, 5-8, and 5-9 show a trend in the association between correct menu-item number and the two most frequently selected menu-item numbers (regression with 95% confidence bands) across all 16 recall test blocks for trials 1, 3, and 5 carried out by the experimental subjects. Figures 5-10, 5-11, and 5-12 show similar scatter plots and regressions for control subjects. The label on each point is the cell frequency for that correct/selected menu-item pair. The correct selections would place all menu-item selections along the positive diagonal (1,1 to 5,5). Regression analysis was used to examine the reduction in error associated with menu-item selection across trials. The reduction in error associated with regression is evident from the increase in the values of R^2 as trials progress (see Table 5-3) and the corresponding narrowing of the 95% confidence interval bands shown in Figures 5-7 through 5-9. Thus, we see in the early experimental trials a more disperse set of menu-item selections which move closer to the diagonal on successive trials as the number of trials (windows) visible on the screen increases.

Selection Accuracy and Display Geometry Conditions. For these analyses, the data were classified by window arrangement (left diagonal or right diagonal) and text justification (left-justified or right-justified). No difference in correct menu-item selection accuracy was found across the four display geometries (left-left, left-right, right-left, right-right). For the experimental subjects, no difference in menu-item selection accuracy was found between the two window arrangement conditions (left-diagonal, right-diagonal). However, differences in selection accuracy associated with the two text justification conditions (left-justified, right-justified) was significant as shown by a Kruskal-Wallis One-way ANOVA [$\chi^2 = 5.74$, $df = 1$, $p < .01$]. The direction of this difference revealed left-justified text produced more accuracte menu-item selections than right-justified text by a slim margin [left: $M = 0.44 \pm 0.50$ *SD* proportion correct; right: $M = 0.38 \pm 0.47$ *SD* proportion correct]. Control subjects showed no difference in menu-item selection accuracy for the window arrangement or text justification conditions.

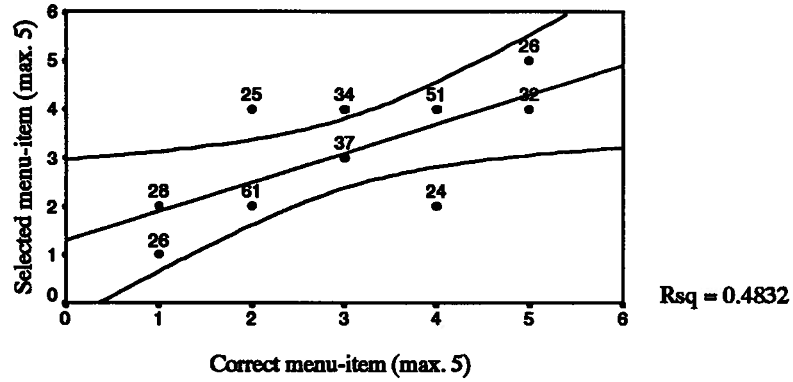

Figure 5-7. Relationship between correct menu-item and two most frequently selected menu-items on Trial 1, across all 16 recall test blocks. Experimentals: regression with 95% confidence bands.

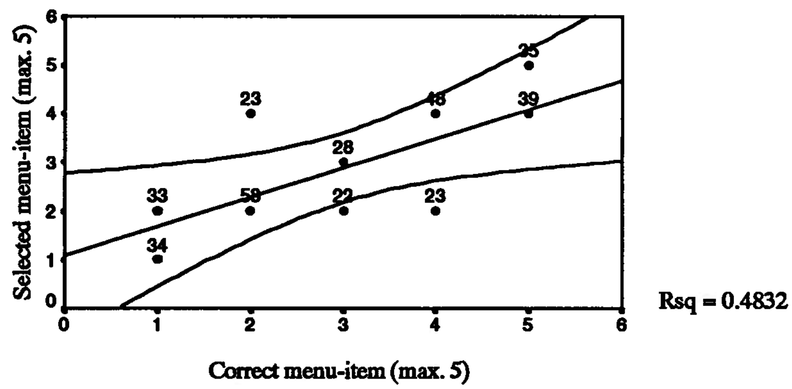

Figure 5-8. Relationship between correct menu-item and two most frequently selected menu-items on Trial 3, across all 16 recall test blocks. Experimentals: regression with 95% confidence bands.

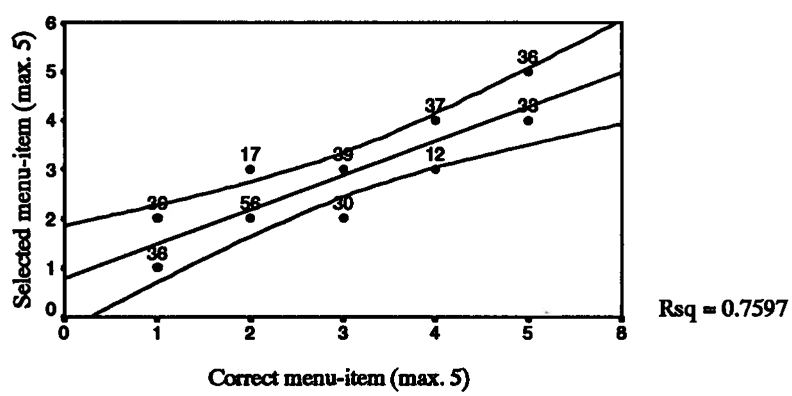

Figure 5-9. Relationship between correct menu-item and two most frequently selected menu-items on Trial 5, across all 16 recall test blocks. Experimentals: regression with 95% confidence bands.

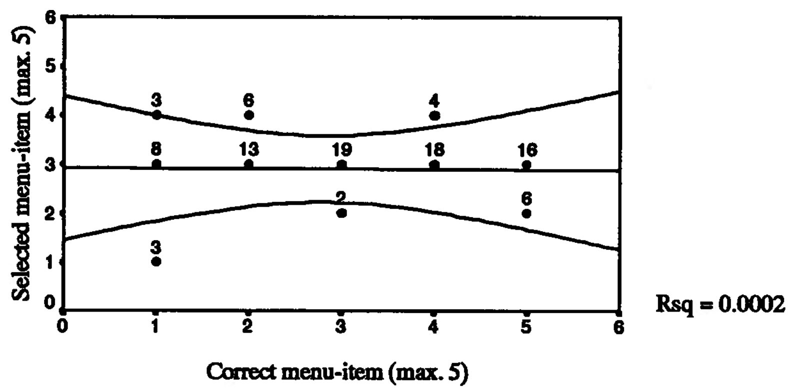

Figure 5-10. Relationship between correct menu-item and two most frequently selected menu-items on Trial 1, across all 16 recall test blocks. Controls: regression with 95% confidence bands.

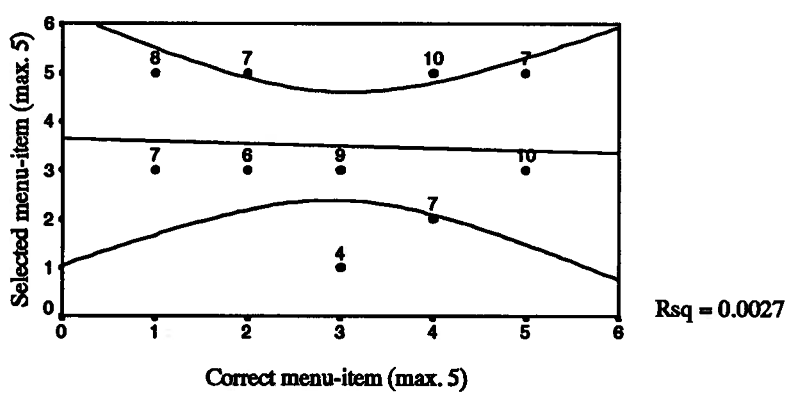

Figure 5-11. Relationship between correct menu-item and two most frequently selected menu-items on Trial 3, across all 16 recall test blocks. Controls: regression with 95% confidence bands.

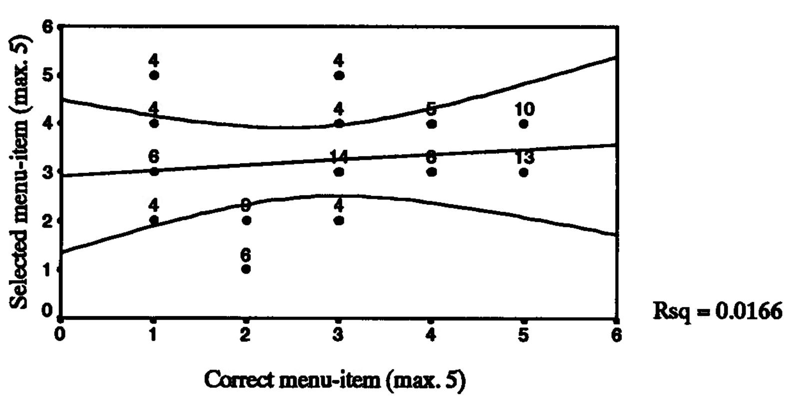

Figure 5-12. Relationship between correct menu-item and two most frequently selected menu-items on Trial 5, across all 16 recall test blocks. Controls: regression with 95% confidence bands.

Table 5-3. Values of R^2 for the relationship between correct menu-item values and subject menu-item selections across the five trials within all recall blocks combined.

Trial Number (number of windows visible on the screen)	R^2 Controls	R^2 Experimentals
1	0.000	0.483
2	0.036	0.483
3	0.003	0.483
4	0.098	0.760
5	0.017	0.760

Subjective Reports

When questioned at the end of the recall test phase, subject reports varied on which window arrangement made the question answering task easier. Table 5-4 shows the preferences expressed by the 13 subjects who responded to a short questionaire. These results show a slight tendency for subjects to find it easier to carry out the selection task when the window arrangement being presented was also associated with the fixed-mapping condition. It is interesting to note that only one subject, out of 32, reported recognizing the fixed-mapping of answers to menu-items, however, the mapping pattern this subject described bore no relationship to the pattern they were actually assigned for the experiment.

Table 5-4. Relationship between the fixed-mapping window arrangement assigned to a subject and their subjective reports of task difficulty.

Number of Subjects Reporting	Assigned Fixed-mapping Window Arrangement	Reported Window Arrangement and Task Difficulty
5	Right-diagonal	Right-diagonal -- Easy
3	Left-diagonal	Left-diagonal -- Easy
3	Left-diagonal	Right-diagonal -- Easy
1	Left-diagonal	Left-diagonal -- Hard
1	Right-diagonal	Right-diagonal -- Hard

CHAPTER SIX

Discussion

This chapter examines the major results of the study in relation to the hypotheses being investigated, and draws some tentative conclusions regarding the presentation of information in a multiple-window user interface. The purpose of the research reported in this thesis was to investigate the influence of visual context on window navigation in a human-computer interface. Specifically, the research examined the effects of different window and text displays on different measures of memory, both implicit and explicit, that focused on menu-item selection. Two hypotheses were tested by the study:

A:

H1: The arrangement of windows and justification of text on a computer display influenced performance on implicit and/or explicit memory tests.

H0: The arrangement of windows and justification of text had no influence on memory performance.

B:

H1: If window arrangement and text justification influenced performance on implicit and/or explicit memory tests, this influence was mediated by specific aspects of the display geometry.

H0: There was no difference in memory performance across display geometry.

The following sections consider these hypotheses in light of the findings.

Major Findings

Four major results from the experiment address the hypotheses examined in this thesis. First, no significant difference in menu-item selection speed and accuracy was observed between the fixed- and variable-mapping conditions during training. Second, no significant difference in menu-item selection accuracy was observed between the fixed- and variable-mapping conditions on a recall test, although the fixed-mapping condition did produce slightly higher selection accuracy relative to the variable-mapping condition. Subjects in the experimental group showed significantly higher selection accuracy than controls on a recall test. Third, all experimental subjects showed a pattern of menu-item selections which became increasingly more focused around the correct menu-item as trials progressed on a recall test. Fourth, no difference in menu-item selection performance at recall was evident between the two tested window arrangements, however, selection accuracy at recall was significantly higher when text appeared in the left-justified format than in a right-justified format.

The first finding, that menu-item selection speed and accuracy were not affected by mapping conditions during the course of training, stands in conflict with both hypotheses. No difference between mapping conditions was found in the speed of item selection or menu-item selection accuracy over the course of 48 training trials. If users were able, over time, to implicitly associate the fixed pattern of correct item selections with specific components of the screen display, we expect the screen components to act as cues and influence selection performance. This did not appear to be the case.

The mouse actions associated with menu-item selection were reasonably simple. The average time required to move the mouse from the **OK** menu-item to one of the five targets was 546 ms based on the MacKenzie and Buxton (1992) formulation of Fitts's law. This leaves the approximately 750 ms to 1250 ms above movement time to account for item decision time. Given the variation in STs across blocks, the influence of decision time may have masked any effects associated with mapping condition. The absence of a difference in overall ST was also true for each of the four display geometry conditions tested. If performance across conditions does vary, selection time in this instance is not sensitive to these differences.

One of the standard guidelines for UI design involves the placement of items users will commonly access at fixed locations on the screen. Users who regularly interact with a specific graphical user interface generally become familiar with the items in the menus. As well, when making a menu selection the desired item (Open, Delete or Exit, for example) will, if item naming has been done correctly, be obvious among the other items in the list. As noted in Chapter Two, this technique is used in most window systems today. Telling users about the specific placement of menu-items and other interface attributes seems to help them learn the system. We can ask: will these users learn the location of items when they are not explicitly told to look for a pattern, but instead are provided with visual cues associated with the pattern? Will this learning be reflected in selection accuracy?

In this thesis the users' attention was focused on the question answering task with no mention of a subsequent need to recall menu-item positions. A selection accuracy of 95% for the fixed-mapping condition and 93% for the variable-mapping condition shows this aspect of task performance was near ceiling. The question answering task was designed to make the correct answer obvious within the set of five possible answers. The goal was to reduce the effects of decision time by making the search among the possible answers for the correct one relatively fast, and yet provide a task which approximated the type of menu-item selection activity users normally encounter in a window-based computing environment. The slightly better performance under the fixed-mapping condition indicates some aspect of the display may influence selection behaviour if ceiling effects are reduced or eliminated.

Given the unique nature of this experiment, there are few other studies with which a direct comparison can be made. However, the work by Nissen and Bullemer (1987) (discussed in Chapter Three) involved a simple reaction time task to investigate similar memory performance issues. Their experiment involved pressing a key at one of four locations when a symbol appeared on a computer display. The random condition involved presentation of the symbol at one of four randomly selected locations. The repeating condition required presentation of the symbol in a specific pattern of locations. Dual-random and dual-repeating conditions required that subjects count tones played into headphones during the previously described symbol presentation trials. Nissen and Bullemer found no difference in performance between the dual-random condition and the dual-repeating condition. However, reaction time decreased across all blocks for both conditions with accuracy never falling below 94%. Their dual-random and dual-repeating conditions are comparable to the variable-mapping and fixed-mapping conditions of the present study. The primary task in the current study involved reading a question and thinking about an answer while the secondary task involved locating the answer in a list of menu-items. Given the similarity of tasks, the findings for this aspect of the thesis are consistent with Nissen and Bullemer's results.

The second finding is that no significant difference in menu-item selection accuracy appeared between the fixed- and variable-mapping conditions on a recall test. Again, this result fails to support the hypotheses. Although no statistical measure showed a difference, fixed-mapping selection accuracy appeared to be slightly higher than variable-mapping accuracy. For the recall test, users were asked to think back to the previous training trial and try recalling the position of the correct menu-item selected at that time. If training under the fixed-mapping condition facilitates menu-item selection accuracy, we should see better performance on the recall test in this condition. These results suggest that prior exposure to 12 fixed-mapping blocks was insufficient to produce a significant training effect for menu-item selection. It is worth noting that two of the four fixed-mapping blocks (7 and 11) did produce higher mean selection accuracy scores than any of the variable-mapping blocks on recall. Thus, while statistically significant results were elusive, trends which support a mapping effect are common throughout the data.

While the mapping data appeared inconclusive for the experimental users, a pooling of selection accuracy data from both mapping conditions showed a highly significant difference between experimental users and controls. Since control users never viewed the training phase they could expect to perform no better than chance on a recall test, selecting one in five menu-items correctly on average. This chance level of performance was also expected from the variable-mapping experimental users since the random shifting of correct menu-item position should have reduced user performance. The actual results were quite different.

Across all conditions, experimental users performed at levels two times better than chance when recalling previously selected items, choosing two out of five menu-items correctly on average. This result is surprising when the behaviour of users and the nature of the task is closely examined. Users viewed a total of 240 different questions in four different display geometries with a total of 1200 different answers presented across five menu-item locations. These users selected an answer for a given question in less than 2 seconds, on average, before continuing with the next trial. As well, users did not know they would be subsequently asked to recall the location of correct answers, so there was no reason for them to explicitly try to remember the question or answer location. Since users made their recalled menu-item selections in less than one second, on average, the data suggest some aspect of the display must be providing strong cues to guide their choice without lengthy periods of reflection. If the visual context of the display plays an important role in guiding selection behaviour, these results suggest that a brief prior exposure to a specific display organization can significantly facilitate subsequent task performance.

This finding may enhance our understanding of Koved and Shneiderman's (1986) study of context and embedded menus used in hypertext systems. Embedded menus are specific items within a set of information, such as particular words in a body of text, that provide links to other chunks of information. Explicit menus are generally a list of enumerated items contained in a separate window from the information of interest. Koved and Shneiderman (1986) found users answered significantly more questions correctly when using embedded menus to search for information as opposed to explicit menus. It is possible that users perform better with embedded menus because, having read the text material once, the task allowed them to search an already familiar display organization rather than switch to what would be a new window presentation (and context) representing each explicit menu.

The third finding highlights the fact that all experimental users showed showed a pattern of menu-item selections which became increasingly more focused around the correct menu-item as trials progressed within all blocks. This result does provide support for H1 in that the number of windows (trials) present on the display influenced recall performance. While the number of correct menu-item selections did not change, the substantial improvement in R^2

values moving from trial 1 to trial 5 is a very interesting result. This result indicates that even though users do make an incorrect selection, they are more likely to select a menu-item directly adjacent to the correct item after they have completed three trials. Alternatively, on the first three trials users are more likely to incorrectly select a menu-item which is not directly adjacent to the correct item. If users were only drawing upon previously acquired semantic information (the question text) to assist them during the recall test, we would expect no change in selection errors across the five trials, because each replayed question in the context of its own window would provide sufficient information to support a correct or nearly correct choice. This is clearly not the case. This result suggests that as constraints on the display environment increase (more windows appear), the users' menu-selection behaviour is more tightly focused around the correct menu-items. Whether the observed effect is a result of implicit memory or explicit memory cannot be determined from these data. A characteristic of implicit memory, as measured in studies of priming, is that the highest levels of priming occur when conditions at study match conditions at test. Since the incremental build-up of the screen display with successive windows during the recall test phase reinstates the conditions during the training phase (study), it seems very likely the results observed represent, in part, a priming effect.

There is another aspect of memory research from psychology that may provide some insight into the nature of these results. The cue overload principle (Watkins & Watkins, 1975) states that the efficiency of a retrieval cue in eliciting recall of an item declines as the number of items associated with that cue increases. If we examine the cues provided to users over the course of five trials we see an increase in the specificity of information available. The first trial provides location information in that the trial window appears on either the left or right side of the display. In addition, the first trial provides text justification information. Menu-item selection may provide some mouse movement (i.e. motor action) cues. The second trial adds to the display geometry possibly enhancing the left or right diagonal effect. By the third trial, display geometry is becoming more prominent with the growing number of window borders filling the display. When the fourth trial appears, the display geometry is approaching its final configuration. As well, the pattern of menu-item selections over the previous three trials, associated with the prior exposure at training, may begin constraining selection behaviour and draw users closer to the correct items even though incorrect selections are made. The fifth and final trial provides the last piece of context information which may enhance the specificity the display as an overall retrieval cue. Extensions to this work considered in a subsequent section will examine possible tests of the cue-overload principle in window-based user interfaces.

The fourth finding indicates that no difference in menu-item selection performance at recall was evident between the two tested window arrangements. In contrast, selection accuracy at recall was significantly better, by a small margin, when text appeared in the left-justified layout. The absence of display geometry effects, and in particular window arrangement effects, was quite surprising. As discussed in Chapter Two, there has been considerable research interest in window layout and window management schemes. The debate over which management style (tiled or overlapped) may provide a more productive work environment is still unresolved and appears to be coupled with the nature of the user's task. The data from this thesis suggest that no particular display geometry is preferred. Rather it is the uniqueness of the display contents and possibly the temporal pattern of display construction, taken as a whole, which may have the greatest influence on user behaviour. This result is consistent with Woods (1984) concept of visual momentum across successive screen displays. Priming effects may provide a basis for the continuity of information presentation across successive displays which Woods (1984) proposes. The importance of this global context suggests that in a window-based environment a brief prior exposure to the content, form and arrangement of every window on the display may influence subsequent task performance within a specific window.

Limitations of the Study

This thesis represents a first step at investigating the often subtle effects of different types of memory on user performance in a window-based human-computer interface. As such, the goal was to examine performance on a simple task which would maintain the interest of users and mimic the fundamental aspects of a menu-driven overlapped window system display. To enhance the appeal of the experiment, a question answering task was devised which made use of 240 different Trivial Pursuit questions. This task was very popular with subjects and, combined with the use of a progress bar visible in the lower portion of the display, maintained subject interest throughout the 45 minute experimental session. Based on this evidence, we recommend this task as a good candidate for other studies where maintaining user interest is important.

In any study, null findings can lead the investigator to think that an increase in sample size or adding more blocks might have produced a larger difference among test conditions. Nissen and Bullemer (1987) obtained differences in reaction time between repeating and random pattern conditions on a button pressing task after approximately 20 repetitions of a 10 trial repeating light pattern. This study presented subjects with a total of 12 blocks, each containing a five trial repeating menu-item selection pattern embedded in 36 blocks where no pattern of menu-item selection was present. In addition, these subjects were asked to perform this task in a visually more complex environment. Increasing the number of blocks or increasing the frequency of fixed-mapping blocks during training may have produced a more noticable effect on selection time without arousing subject awareness of the pattern.

This study focused on two elementary aspects of window-based information display, specifically window arrangement and text layout. By identifying the influence of one or more of these display attributes on measures of memory performance, we can establish a performance baseline for comparison with other, as yet untested, attributes. The design of this experiment purposely did not assess the effects of colour, sound or animation on performance measures of memory in a window environment. A primary reason for limiting the number of factors in this experiment was the increase in sample size required to provide statistically meaningful results. The addition of one factor would have raised the number of subjects needed from 32 to 64, a substantial increase in time and cost. Because this work was probing a new line of user interface research, a conservative approach to the number of factors under study seemed appropriate.

Extensions to this Work

Successful and productive human-computer interaction will depend on a solid understanding of how human cognitive capabilities operate in evolving graphical user interfaces. The purpose of this research was to seek new information about the relationships among task context, window system display organization and performance measures of human memory. Although no single feature of display geometry appeared to influence either implicit or explicit memory test results, a subtle but common pattern in the data emerges. Trends which favour fixed-mapping over variable-mapping conditions on tasks during training and at recall suggest that information on the display is guiding user behaviour. There are several lines of investigation which might be followed to resolve the ambiguities.

First, the fact that experimental subjects did so well on recall, even when the correct menu-item location varied randomly, needs further examination. In the present experiment, the same five questions appeared within a block in the same sequence while the order of block presentation varied between training and recall. It would be interesting to see if a random shuffling of the presentation order of questions within a block at recall would alter selection accuracy. If subjects tended to repeat the pattern of menu-item selections used at training this would suggest the visual aspects of the display have a greater influence on performance. However, if their choices shifted appropriately with the new trial positions of the question text, the content of the question would appear to guide behaviour. If we retained the within-block position of questions, how would a change in text layout affect performance at recall?

A drop in performance as a result of this manipulation would provide evidence that highly specific presentations of information are important cues for recall.

Second, we have seen that users become more focused at selecting correct or near-correct items as trials progress. To pursue this result, using cue-overload (Watkins & Watkins, 1975) as a guiding principle, we could expand the current experiment in two ways. An extension would involve using more questions and windows or more than five menu-items per trial in the same experimental design. If cue-overload plays a role, a point beyond five trials will be reached where adding additional windows (trials) to a block results in declining item selection accuracy and greater variation in the specific items incorrectly selected (measured by a drop in R^2). A slightly different approach would use the current design with five trials, but add other objects to the display which are not relevant to the question answering task. We could then examine the effect on recall performance of increasing the number and/or complexity of objects during training across blocks. This experiment might provide some insight into how complex the contents of a window display can grow before user performance degrades.

Finally, we could examine other aspects of a window-based display such as colour and animation using performance measures of memory. Motion provides strong cues to the visual system about the future location and state of objects. Psychological studies of memory for previously viewed motion are beginning to appear (Nilsson, Olofsson & Nyberg, 1992), with interesting results. Mackinlay, Robertson, and Card (1991) designed the Perspective Wall for the presentation of large linear sets of information (e.g., a disc file storage system organized by date). They place the information on the "wall" and use animation to rotatate the approprate data into view. Mackinlay, Robertson, and Card (1991) state that object constancy (slow changes to the display--visual momentum) is important when performing these animations to provide visual context to the user. Priming studies of the Perspective Wall interface may reveal that implicit memory plays a role in supporting object constancy in this environment.

The findings of this thesis suggest that we need to consider more than the attributes and behaviour of individual windows on a computer display when designing user interfaces. It may be important for application windows and their window managers cooperate such that the complexity of the user interface as a whole does not overwhelm the user. This is likely to be of particular importance with window interfaces supporting the complex information spaces of very large hypertext and hypermedia systems. It might be useful to look upon the human-computer interface as a kind of information ecosystem--an interconnected environment susceptible to small changes in the form of information that can potentially enhance or degrade user performance.

References

Anderson, J. R. & Schooler, L. J. (1991). Reflections of the environment in memory. *Psychological Science, 2,* 396-408.

Baecker, R., Small, I., & Mander, R. (1991). Bringing icons to life. In, *Proceedings of the 1986 ACM Conference on Human Factors in Computer Systems* (pp. 1-6). New York: ACM Press.

Billingsley, P. (1982). Navigation through hierarchical menu structures: does it help to have a map? In, *Proceedings of the Human Factors Society, 26th Annual Meeting*: *Vol. 26,* (pp. 103-107). Santa Monica, CA: Human Factors Society.

Bly, S. & Rosenberg, J. (1986). A comparison of tiled and overlapping windows. In, *Proceedings of the 1986 ACM Conference on Human Factors in Computer Systems* (pp. 101-106). New York: ACM Press.

Boritz, J., Booth, K. S., & Cowan, W. B. (1991). Fitts' Law studies of directional mouse movement. In, *Proceedings of Graphics Interface '91* (pp. 216-223). Toronto: CIPS.

Bush, V. ´945, July). As We May Think. *The Atlantic Monthly, 176,* 101-108.

Card, S. K., Pavel, M., & Farrell, J. E. (1984). Window-based computer dialogues. In B. Shackel (Ed.)., *Human-Computer Interaction-INTERACT '84* (pp. 239-243). London, UK: IFIP.

Card, S. K., Robertson, G. G., & Mackinlay, J. D. (1991). The Information Visualizer, an information workspace. In, *Proceedings of the 1991 ACM Conference on Human Factors in Computer Systems* (pp. 181-188). New York: ACM Press.

Cofer, N. C. (1967). Conditions for the use of verbal associates. *Psychological Bulletin, 68,* 1-12.

Cohen, E. S., Smith, E. T., & Iverson, L. A. (1986). Constraint-based tiled windows. *IEEE Computer Graphics and Applications, 6*(5), 35-45.

Conklin, J. (1987). Hypertext: an introduction and survey. *IEEE Computer, 20*(9), 17-41.

Craik, F. I. & Lockhart, R. S. (1972). Levels of processing: a framework for memory research. *Journal of Verbal Learning and Verbal Behavior, 11,* 671-684.

Engelbart, D. (1963). A conceptual framework for the augmentation of man's intellect. In P. Howerton (Ed.)., *Vistas in Information Handling, Vol. 1* (pp. 1-29). Washington, DC: Spartan Books. In I. Greif (Ed.)., (1988). *Computer Supported Cooperative Work: A Book of Readings.* San Mateo, CA: Morgan Kaufmann.

Engelbart, D. C. & English, W. (1968). A research center for augmenting human intellect. In, *Proceedings of the Fall Joint Computer Conference: Vol. 33.* (pp. 395-410). In I. Greif (Ed.)., (1988). *Computer Supported Cooperative Work: A Book of Readings.* San Mateo, CA: Morgan Kaufmann.

Feiner, S., Nagy, S., & van Dam, A. (1982). An experimental system for creating and presenting interactive graphical documents. *ACM Transactions on Graphics, 1*(1), 59-77.

Foley, J. D., van Dam, A., Feiner, S. K., & Hughes, J. F. (1990). *Computer Graphics: Principles and Practice* (2nd ed.). Reading, MA: Addison-Wesley.

Gaver, W. W. (1989). The SonicFinder: an interface that uses auditory icons. *Human-Computer Interaction, 4,* 67-94.

Gaylin, K. B. (1986). How are windows used? Some notes on creating an empirically-based windowing benchmark task. In, *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems* (pp. 96-100). New York: ACM Press.

Graf, P. & Ryan, L. (1990). Transfer-appropriate processing for implicit and explicit memory. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 16,* 978-992.

Graf, P. & Schacter, D. L. (1985). Implicit and explicit memory for new associations in normal and amnesic subjects. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 11,* 501-518.

Graf, P., Shimamura, A., & Squire, L. (1985). Priming across modalities and priming across category levels: extending the domain of preserved function in amnesia. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 11,* 386-396.

Graf, P., Squire, L., & Mandler, G. (1984). The information that amnesic patients do not forget. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 10,* 164-178.

Halaz, F. G., Moran, T. P., & Trigg, R. H. (1987). NoteCards in a nutshell. In, *Proceedings of the 1987 ACM Conference on Human Factors in Computer Systems* (pp. 45-52). New York: ACM Press.

Henderson, D. A. & Card, S. K. (1986). Rooms: the use of multiple virtual workspaces to reduce space contention in a window-based graphical user interface. *ACM Transactions on Graphics, 5*(3), 211-243.

Hirst, W. (1982). The amnesic syndrome: descriptions and explanations. *Psychological Bulletin, 91*, 435-460.

Hypercard User's Guide. (1988). Cupertino, CA: Apple Computer, Inc.

Jacoby, L. L. & Dallas, M. (1981). On the relationship between autobiographical memory and perceptual learning. *Journal of Experimental Psychology: General, 110*, 306-340.

Johnson, J., Roberts, T. L., Verplank, W.,Smith, D. C., Irby, C. H., Beard, M., & Mackey, K. (1989). The Xerox Star: a retrospective. *IEEE Computer, 22*(9), 11-29.

Kolers, P. A. (1976). Reading a year later. *Journal of Experimental Psychology: Human Learning and Memory, 2*, 554-565.

Koved, L. & Shneiderman, B. (1986). Embedded menus: selecting items in context. *Communications of the ACM, 29*, 312-318.

Krol, E. (1992). *The Whole Internet: User's Guide & Catalog*. Sebastopol, CA: O'Reilly & Associates.

Lampson, B. (1988). Personal distributed computing: the Alto and Ethernet software. In, Goldberg, A. (Ed.). *A History of Personal Workstations*. New York: ACM Press.

Macintosh User's Guide. (1992). Cupertino, CA: Apple Computer, Inc.

MacKenzie, I. S., Sellen, A., & Buxton, W. (1991). A comparison of input devices in elemental pointing and dragging tasks. In, *Proceedings of the 1991 ACM Conference on Human Factors in Computer Systems* (pp. 161-166). New York: ACM Press.

MacKenzie, I. S. & Buxton, W. (1992). Extending Fitts' Law to two-dimensional tasks. In, *Proceedings of the 1992 ACM Conference on Human Factors in Computer Systems* (pp. 219-226). New York: ACM Press.

Mackinlay, J. D., Robertson, G. G., & Card, S. K. (1991). The Perspective Wall: detail and context smoothly integrated. In, *Proceedings of the 1991 ACM Conference on Human Factors in Computer Systems* (pp. 173-179). New York: ACM Press.

MacLean, A., Barnard, P., & Hammond, N. (1984). Recall as an indicant of performance in interactive systems. In B. Shackel (Ed.)., *Human-Computer Interaction-INTERACT '84* (pp. 311-315). London, UK: IFIP.

Mantei, M. (1982). *Disorientation Behaviour in Person-Computer Interaction.* Unpublished doctoral dissertation, University of Southern California.

McCracken, D. L. & Akscyn, R. M. (1984). Experience with the ZOG human-computer interface system. *International Journal of Man-Machine Studies, 21*, 293-310.

Microsoft Windows User's Guide, Version 3.0. (1990). Redmond, WA: Microsoft Corporation.

Microsoft Windows User's Guide for the Microsoft Windows Operating System, Version 3.1. (1992). Redmond, WA: Microsoft Corporation.

Microsoft Windows Version 3.1 Multimedia Programmer's Guide. (1992). Redmond, WA: Microsoft Corporation.

Miller, G. A. (1956). The magical number seven, plus or minus two: some limits on our capability for processing information. *Psychological Science, 63*, 81-97.

Murdock, B. B. (1974). *Human Memory: Theory and Data.* Potomac: Lawrence Earlbaum.

Myers, B. A. (1984). The user interface for Sapphire. *IEEE Computer Graphics and Applications, 4*(12), 13-23.

Myers, B. A. (1985). The importance of percent-done progress indicators for computer-human interfaces. In, *Proceedings of the 1985 ACM Conference on Human Factors in Computer Systems* (pp. 11-17). New York: ACM Press.

Myers, B. A. (1988). A taxonomy of window manager user interfaces. *IEEE Computer Graphics and Applications, 8*(5), 65-84.

Nelson, T. (1975). *Dream Machines.* South Bend: Theodore Nelson.

NewWave User's Guide. (1992). Sunnyvale, CA: Hewlett-Packard Co.

Nielsen, J. (1990). The art of navigating through hypertext. *Communications of the ACM, 33*, 296-310.

Nielsen, J. (1993). Noncommand user interfaces. *Communications of the ACM, 36*(4), 83-99.

Nilsson, L. G., Olofsson, U., & Nyberg, L. (1992). Implicit memory of dynamic information. *Bulletin of the Psychonomic Society, 30*, 265-267.

Nissen, M. J. & Bullemer, P. (1987). Attentional requirements of learning: evidence from performance measures. *Cognitive Psychology, 19*, 1-32.

Norman, K. L., Weldon, L. J., & Shneiderman, B. (1986). Cognitive layouts of windows and multiple screens for user interfaces. *International Journal of Man-Machine Studies, 25*, 229-248.

Norman, K. L. (1991). *The Psychology of Menu Selection: Designing Cognitive Control at the Human/Computer Interface.* Norwood, NJ: Ablex.

Robertson, G. G., Mackinlay, J. D., & Card, S. K. (1991). Cone Trees: animated 3D visualizations of hierarchical information. In, *Proceedings of the 1991 ACM Conference on Human Factors in Computer Systems* (pp. 189-194). New York: ACM Press.

Robertson, G. G., Card, S. K., & Mackinlay, J. D. (1993). Information visualization using 3D interactive animation. *Communications of the ACM, 36*(4), 57-71.

Roediger, H. L. (1990). Implicit memory: retention without remembering. *American Psychologist, 45*(9), 1043-1056.

Schacter, D. L. (1987). Implicit memory: history and current status. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 13*, 501-518.

Scheifler, R. W. & Gettys, J. (1986). The X Window System. *ACM Transactions on Graphics. 5*(2), 79-109.

Shannon, C. E. & Weaver, W. (1949). *The Mathematical Theory Of Communication.* Urbana, IL: University of Illinois Press.

Shneiderman, B. (1992). *Designing the User Interface* (2nd ed.). New York: Addison-Wesley.

Siegel, S. (1956). *Nonparametric Statistics for the Behavioral Sciences.* New York: McGraw-Hill.

Sun, F. K., Cowan, W. B., & Booth, K.S. (1990). Understanding visual effects in a windowed environment. In, *Proceedings of Graphics Interface '90* (pp. 100-105). Toronto: CIPS.

Tulving, E. (1983). *Elements of Episodic Memory.* Oxford, UK: Clarendon Press.

Using ToolBook: A Guide to Building and Working with Books. (1989). Bellevue, WA: Asymetrix Corporation.

Warrington, E. K. & Weiskrantz, L. (1970). Amnesic syndrome: consolidation or retrieval? *Nature, 228*, 629-630.

Warrington, E. K. & Weiskrantz, L. (1974). The effect of prior learning on subsequent retention in amnesic patients. *Neuropsychologia, 12*, 419-428.

Warrington, E. K. & Weiskrantz, L. (1982). Amnesia: a disconnection syndrome? *Neuropsychologica, 20*, 233-248.

Watkins, O. C. & Watkins, M. J. (1975). Buildup of proactive inhibition as a cue-overload effect. *Journal of Experimental Psychology: Human Learning and Memory, 104*, 442-452.

Weiser, M. (1993). Some computer science issues in ubiquitous computing. *Communications of the ACM, 36*(7), 74-84.

Weldon, M. S. & Roediger, H. L. (1987). Altering retrieval demands reverses the picture superiority effect. *Memory & Cognition, 15*, 269-280.

Williams, G. (1984). The Apple Macintosh Computer. *Byte, 9*(2), 30-54.

Woods, D. (1984). Visual momentum: a concept to improve the cognitive coupling of person and computer. *International Journal of Man-Machine Studies, 21*, 229-244.

Wright, P. (1991). Cognitive overheads and prostheses: some issues in evaluating hypertexts. In, *Proceedings of the Third ACM Conference on Hypertext* (pp. 1-12). New York: ACM Press.

Yankelovich, N., Meyrowitz, N., & van Dam, A. (1985). Reading and writing the electronic book. *IEEE Computer, 18*(10), 15-30.

APPENDIX A

Microsoft Windows Mouse Timing Calibration

One of the issues associated with recording of reaction time data using the mouse in the Microsoft Windows environment involves the delay in system response, due to operating system overhead, to the mouse button click hardware event. Since little information was available to us on the details of the mouse driver software, we performed a small calibration experiment to determine the extent of system overhead associated with left mouse button click events. The Engineering Technician of the Department of Psychology constructed a simple external hardware timing device to measure the elapsed time, in milliseconds, between successive left mouse button clicks on a Microsoft Bus Mouse.

Method

Materials. The external timer system consisted of an Intel 8253-5, 16-bit counter/timer device operating off of a one KHz square wave timebase supplied by a Statetek PX-600 crystal oscillator. A standard J-K flip-flop was used to start and stop the 16-bit counter employing a pulse derived from clicking the left mouse button. The output from the pin of the bus mouse connector carrying the pulse produced by a click of the left mouse button was tied to the toggle input (J-K) of the flip-flop. The experiment control software developed for the primary study was used in its normal operating mode to provide the measure of elapsed time between mouse clicks, based on Microsoft Windows Multimedia Timer Services system calls (*Microsoft Windows Version 3.1 Multimedia Programmer's Guide*, 1992). The tests were run on a Hewlett-Packard Vectra 486 computer running at a clock speed of 33 MHz.

Procedure. After initializing and arming the external hardware timer system, the experimental control software was loaded and a series of target training trials run. Hardware and software times were recorded for each pair of left button mouse clicks, a start time for the selection of the OK menu-item and a stop time for the selection of an answer menu-item. Times were recorded to the nearest millisecond.

Results

A regression analysis of software timing interval on hardware timing interval was highly significant with an R^2 value of 0.999, $F(1,18) = 56132$, $p < .001$. The slope of the regression line was 0.999 with a standard error of 0.004 and the Y-intercept was 7.13.

Discussion

There was very little difference between the times recorded via hardware measurement and those obtained using the Windows Multimedia Timer Services. The regression intercept indicates software timing may carry a minimum overhead of 7 ms. Based on pilot testing, subject variation in mouse selection time was on the order of 300 to 400 ms. Given these results, it was decided that overhead due to Microsoft Windows system software processing of mouse activity could be ignored when recording subject selection times fr the purposes of the primary study.

APPENDIX B

Subject Consent Form, Debriefing Form, and Post-experiment Questionnaire

The consent form was provided to subjects prior to the start of the experimental session. Subjects were asked to read the consent form and, if satisfied with the information provided, asked to date and sign the form. These records were kept in accordance with the rules governing the use of the human subject pool.

At the conclusion of an experimental session subjects were verbally debriefed about the nature of the experiment and asked if they had any questions or concerns about the procedure. Subjects were then each given a copy of the debriefing form.

At the end of the recall test phase subjects completed the four item subjective questionnaire.

Consent Form

Computer Display Organization

Experimenter: Gary MacIsaac; Peter Graf

Location: Room 3203A in the Kenny Building
Gary MacIsaac's office -- 2531 Kenny Building
Dr. Graf's office -- 3110 Kenny Building

Telephone: Laboratory -- 822-2140
Gary MacIsaac's office -- 822-6733
Dr. Graf's office -- 822-6635

A modern computer can display information in many different ways: information can appear in different colors, different type fonts, left- or right adjusted, inside a circle or a box -- a window -- on any part of the screen, and windows can be arranged in many different ways. The designers of computer programs that are intended for use by the general public think carefully about how information and instructions ought to be displayed; they want to display instructions/information in such a way as to make the programs look good and easy to use. But very little is known about which characteristics of a display -- such as size and arrangement on the screen -- actually have an influence on how easy a program is to use. The goal of the present experiment is to collect this kind of information, specifically, to find out whether the speed with which you can work with a program is influenced by the manner in which text is displayed (left- vs. right-adjusted) and by the manner in which various displays or windows are arranged on the screen.

In the experiment, we will ask you to answer a series of Trivial Pursuit™ [1] questions. There will be many sets of five trials, with a different question appearing on each trial. Each question will be shown inside a small window, a box on the screen. On each trial, you will be asked to read the question, then to press the OK key, and finally to select the correct answer from among 5 alternatives. The experiment has two critical *independent variables*: the manner in which the individual question-windows are arranged (stacked from the top left to bottom right, or stacked from the top right to bottom left), and the type of text justification (left or right). The critical *dependent variable* is the time required for you to answer each question. The results of the experiment will tell us whether the speed with which you can answer questions on a computer is influenced by the manner in which information is displayed.

Your results in this experiment will be treated in strict confidence. To this end, the test data and computer files will not contain information that would allow anyone to link the results with you. Your results will be used to write a scientific report about the effects of computer display organization on performance. Although you participate as an individual, your individual results will not be published in any form; they will be combined with those from a large group of other subjects.

This experiment will require approximately 45 minutes to complete. If, at any time during the experiment, you would like to stop, you may do so. Your decision will not be held against you and will not be reported to anyone. If you consent to participating in the experiment described above, please sign the form on the following page. Thank you.

[1] Trivial Pursuit is a trademark of Horn Abbot of Downsview, Ontario.

Consent Form for Computer Display Organization Experiment

Please <u>print</u> your name below	Please <u>sign</u> your name below	Please provide today's date

Debriefing Form

Computer Display Organization

Experimenter: Gary MacIsaac; Peter Graf

Location: Room 3203A in the Kenny Building
Gary MacIsaac's office -- 2531 Kenny Building
Dr. Graf's office -- 3110 Kenny Building

Telephone: Laboratory -- 822-2140
Gary MacIsaac's office -- 822-6733
Dr. Graf's office -- 822-6635

A modern computer can display information in many different ways: information can appear in different colors, different type fonts, left- or right adjusted, inside a circle or a box -- a window -- on any part of the screen, and windows can be arranged in many different ways. The designers of computer programs that are intended for use by the general public think carefully about how information and instructions ought to be displayed; they want to display instructions/information in such a way as to make the programs look good and easy to use. But very little is known about which characteristics of a display -- such as size and arrangement on the screen -- actually have an influence on how easy a program is to use. The goal of the present experiment is to collect this kind of information, specifically, to find out whether the speed with which you can work with a program is influenced by the manner in which text is displayed (left- vs. right-adjusted) and by the manner in which various displays or windows are arranged on the screen.

In the experiment, you had to answer a series of Trivial Pursuit™[2] questions. There were many sets of five trials, with a different question appearing on each trial. Each question was shown inside a small window, a box on the screen. On each trial, you were asked to read the question, then to press the o.k. key, and finally to select the correct answer from among 5 alternatives. The experiment has two critical ***independent variables***: the manner in which the individual question-windows were arranged (stacked from the top left to bottom right, or stacked from the top right to bottom left), and the type of text justification (left or right). The critical ***dependent variable*** was the time required for you to answer each question. The results of the experiment will tell us whether the speed with which you can answer questions on a computer is influenced by the manner in which information is displayed.

Your results in this experiment are treated in strict confidence. To this end, the test data and computer files will not contain information that would allow anyone to link the results with you. Your results will be used to write a scientific report about the effects of computer display organization on performance. Although you participated as an individual, your individual results will not be published in any form; they will be combined with those from a large group of other subjects.

If you have any further questions about this experiment and its implications, I will be happy to answer them. For questions that arise after the experiment, you can contact me by calling the above phone numbers. If you are interested in the general topic addressed by this experiment, you might read: Weiser, M. (1991) The Computer for the 21st Century. **Scientific American**. September, 1991, pages 94-104.

---

[2]Trivial Pursuit is a trademark of Horn Abbot of Downsview, Ontario.

Post-experiment Subjective Questions

Computer Display Organization Experiment

1. How did you find the mouse training task as a procedure for gaining practice with the mouse?

2. How did you find the question answering task as a procedure involving the use of the mouse?

3. Did you find one or more of the display presentations made the task easier or more difficult to carry out? If so, which display presentation(s) made the task easier or more difficult?

4. Do you have any other comments you would like to make about the experiment?

APPENDIX C

Questions Used as Trial Items

The following are a sample of the Trivial Pursuit and other questions used within target training trials and recall test trials. Trivial Pursuit is a registered trademark of Horn Abbot of Downsview, Ontario, Canada. The Trivial Pursuit Genus Edition questions used are copyright © 1981 Horn Abbot Ltd.

Itemid: 1 ;
Text: What country do the Galapagos Islands belong to? ;
A: Rabies ;
A: Squash ;
A: Dopey ;
A: Baseball ;
Correct: Ecuador ;

Itemid: 2 ;
Text: What food of the three bears did Goldilocks eat? ;
A: Barrels ;
A: Hydrogen ;
A: China ;
A: Eleven ;
Correct: Porridge ;

Itemid: 3 ;
Text: What's the official language of Egypt, Tunisia and Morocco? ;
A: Eight ;
A: Flash ;
A: Shellac ;
A: Thinner ;
Correct: Arabic ;

Itemid: 4 ;
Text: What do you get by adding Lactobacillus bulgaricus to milk? ;
A: Colorado ;
A: Interpol ;
A: Polo ;
A: Greece ;
Correct: Yogurt ;

Itemid: 5 ;
Text: What did Abebe Bikila go without in winning the 1960 Olympic marathon? ;
A: Paris ;
A: Jakarta ;
A: Seattle ;
A: Berlin ;
Correct: Shoes ;

Itemid: 6 ;
Text: What country's flag flies over the Canary Islands? ;
A: Bicycle ;
A: Skate ;
A: Row ;
A: Dive ;
Correct: Spain's ;

Itemid: 7 ;
Text: What's the international radio code word for the letter Z? ;
A: Tourism ;
A: England ;
A: $150 ;
A: Four ;
Correct: Zulu ;

Itemid: 8 ;
Text: How many times larger than life size is the Statue of Liberty? ;
A: Cuba ;
A: Brazil ;
A: Snake ;
A: Ant ;
Correct: 20 ;

Itemid: 9 ;
Text: How many dollars a day did Arthur Frommer say you could get by on in Europe in 1968? ;
A: SFU ;
A: UBC ;
A: Yale ;
A: Harvard ;
Correct: Five ;

Itemid: 10 ;
Text: How many drops make a dash in cooking? ;
A: Chicago ;
A: New York ;
A: Paris ;
A: Berlin ;
Correct: Six ;

Itemid: 11 ;
Text: What's the largest Scandinavian country? ;
A: Hammer ;
A: Nail ;
A: Saw ;
A: Drill ;
Correct: Sweden ;

Itemid: 12 ;
Text: What country lies directly south of Detroit? ;
A: Table ;
A: Chair ;
A: Bench ;
A: Stool ;
Correct: Canada ;