A COMPARISON OF HUMAN PERFORMANCE ON COMPUTER TEXT EDITING TASKS USING WINDOWED AND NON-WINDOWED STRATEGIES

by

Timothy J. O'Keefe

Dissertation submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of
DOCTOR OF PHILOSOPHY
in
Psychology

APPROVED:

A.M. Prestrude, Chairman

John G. Casali

Roseanne G. Foti

Robert C. Williges

Stephen J. Zaccaro

June, 1987

Blacksburg, Virginia

A COMPARISON OF HUMAN PERFORMANCE ON COMPUTER TEXT EDITING TASKS USING WINDOWED AND NON-WINDOWED STRATEGIES

by

Timothy J. O'Keefe

A.M. Prestrude, Chairman

Psychology

(ABSTRACT)

Software packages which use windows have become increasingly popular in the last two years. Their popularity derives from the belief that windows will improve productivity by decreasing task completion time. However, two studies (Silver, 1985 and Davies, Bury and Darnell, 1985) have found this not to be the case. In fact, one of the studies (Davies et al., 1985) found that task completion time was increased when using windows. It is thought that performance using window systems is a function of the number of responses required to be executed as well as the amount of information which must be found and used to complete a task. The purpose of the present study was to determine under what conditions, in terms of memory load and task complexity, performance using windows and non-window strategies differed.

Forty-eight subjects were placed in one of four environments and each performed six editing tasks which varied on complexity and memory load level. Human performance in one windowed environment was compared to three non-window strategies. These three strategies, note-taking, memorizing, or switching between files, were included to allow comparisons in terms of working memory and number and types of responses executed. The tasks required subjects to locate information from a supplementary file and type it into a main file. The three memory load levels which were used required subjects to find either 2, 4 or 8 pieces of information. The two complexity levels referred to the placement of needed information in the supplementary file; whether or not information was located in close proximity to other needed information.

Results indicated that it made little difference which system was used in the low memory load condition. However, as memory load increased, more subjects were found to make errors in the non-window conditions. More responses were executed in the Switch condition than in the Window or Memorization conditions in the high memory load condition. Mental workload was also found to be higher in the Memorization and Switch conditions than in the Window and Notes conditions as memory load increased. Nevertheless, there was no significant interaction for task completion times between Memory Load and Environment. This was thought to be due to a failure to adequately load working memory as well as a failure of a test of verbal and spatial ability to account for individual differences.

It was concluded that the benefits of windows are not apparent until one's working memory capacity is exceeded. As memory load increases beyond this point, it is thought that memorization will quickly become an inefficient strategy due to limitations of memory capacity. As memory load continues to increase, a switching strategy should become inefficient due to limitations of both memory and response capacities. A strategy of note taking should not become inefficient until a large memory load is placed upon the user. This is because note taking is a well

learned uncomplicated response. The benefits of windows include a reduction in the number of responses, errors, and mental workload due to their ability to reduce the amount of mental resources required by providing the user with a very efficient and accurate memory aid.

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to the members of my committee for their wisdom, guidance, and encouragement. First, special thanks go to my committee chairman, Dr. A. M. Prestrude who believed in me and gave me unmeasurable amounts of help and support through some difficult times over the last five years. Second, I wish to thank Dr. Roseanne Foti for her valuable assistance and guidance in analyzing my dissertation as well as being a friend. I would also like to thank Dr. Robert Williges for his expertise and insightful comments. Finally, I extend my appreciation to Dr. Zaccaro and Dr. Casali for their support and help both on my dissertation and in my graduate career.

The many good friendships which I developed in graduate school made the work seem a little easier and the times a lot funner. I would like to thank for all of their encouragement, stimulating conversations and good times. I would also like to thank for all of their help, their friendship and for making our class a great one. An extra special thanks goes to for their friendship, great fun and tremendous support. I am especially greatful to who was extremely helpful with his expert advise on many technical problems which I encountered on this research.

My family has been a source of great encouragement through out these past five years. I am indebted to my parents,

for always listening and believing in me. Finally, the love, support, patience, and understanding which my wife, has given me these past few years made all of this research possible.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

INTRODUCTION

A primary use of computers in business is to increase productivity by providing information to the user quickly and efficiently. With the development of windows for computers, it has become possible to have several distinct pieces of information displayed to the user at one time. The window can act as a memory aid and save the user valuable time. Thus, windows potentially offer a way of increasing productivity by speeding up task completion time and reducing cognitive load. Davies, Bury and Darnell (1985) discuss three benefits of window displays. First, windows can present information which supplements the user's primary task. For instance, a window of a spreadsheet can be considered supplemental to the primary task of editing a business brief. A second benefit is to be able to monitor changes in a secondary window. For example, data in the primary window, such as productivity rate, could be graphed in the secondary window to give the user a visual portrayal of the data. And finally, two separate locations in the same application can be shown at the same time to aid in moving or copying data from one location to another location.

The drop in price of RAM chips for personal computers brought a concomitant rise in the number of RAM resident programs (Jeffries, 1985), including window programs such as IBM's TopView and Microsoft's Windows. This proliferation of software has given a great number of people the potential to increase productivity. However, this potential is useless if the human cannot effectively use the information presented (Steinbrecher, 1984). The large number of computer programs which contain windows prompts the question: are windows really useful? By useful, it is meant 1) do windows increase the speed with which a task can be accomplished, and 2) do windows reduce the number of errors in completing a task? In asking and answering these questions, one has not gained an understanding about windows in general, but rather an understanding about the specific software one has used. Instead, one must first ask <u>why</u> windows might be useful. In asking this question, one can develop a theoretical framework which will enable the determination as to how window software, in general, should be designed and in what situations it should be used. In this manner, one can understand null or possibly contradictory results.

Many software manufacturers would like to say that programs containing windows will increase productivity by some great percentage for all tasks. Intuitively, one might think that a window would be useful to call up some help information, to access a file with the latest budget so that the information could be typed into a letter to a colleague, or that a phone number be easily accessed. However, it is also possible that the windows could hinder performance. Consider the case where a manager needs to find the phone number of a client so the information can be typed into a letter to his/her supervisor. The information resides in a computer file separate from the current one. The user can call up a window to access the phone number. If the windowing system which the user is working on is complicated, the user may expend more effort and resources to use the windowing system than to simply exit out of that application, find the phone number and get back into the application. Thus, a window is not a useful device in all instances.

It is thought, by the present author, that as any task begins to tax and overload cognitive capacities, performance will decline. Any system which can reduce the amount of strain on cognitive capacities should increase performance. Simple window systems potentially can provide one way in which the strain of computer tasks on certain cognitive capacities may be reduced. Given that there are several different types of cognitive capacities (Wickens, 1984), windows should increase performance over non-windows if they reduce the cognitive load on an operator's limited capacity working memory. Windows should also reduce the load on the limited cognitive resources required for the selection and execution of responses.

It is proposed that the usefulness of windows relies on their ability to reduce the amount of cognitive load placed on the operator and reduce the amount of selection and execution of responses. The reduction of cognitive load and keyboard responses should result in faster task completion times. Cognitive load, or workload, can be diminished by reducing the amount of information and the time that this information must be held in working memory. Selection and execution of responses include not only keyboard responses, but also responses required to write down information which may be to large to fit in working memory. Responses may be reduced by requiring fewer keystrokes to perform a task in a window system than in a non-window system and by allowing the user to view the information directly on the screen as in a window format.

In the case of working memory, windows can serve as a memory aid. By memory aid, it is meant that instead of having to commit help information, directory information or any other information to memory, one could have the information displayed in a window. The information would not have to be held in working memory for a long period of time, only long enough to be used in the main application. Also, less information would have to be held in memory because the user can quickly acquire more information by glancing down to the window.

Windows should also reduce the cognitive load on the selection and response stage of cognitive processing. When the user must remember or understand a great deal of information, the person may write this information down in a non-windowed environment. In a windowed environment, the user need only glance down. There is no need to take notes which take time and may result in transcription errors.

It is important here to discuss non-window strategies. Operators in a non-window environment may adopt different strategies in order to complete a task. For instance, one operator may memorize information while another operator may take notes on information. Or an operator may open and close files a number of times in order to find information. Each of these strategies places a different emphasis on working memory resources and response selection and execution resources. For instance, memorizing information contains a strong memory component while switching from one application to another contains both a memory component and a response component. Note taking requires more of a response component than a memory component.

It is also proposed that the usefulness of windows will be moderated by an interaction of the task which the user must perform, the attributes of the windows themselves, and finally the personal attributes of the user. Some of these are discussed below.

Some task dimensions which might influence the usefulness of windows include how busy the task requires the operator to be, if the user must search for text (as in correcting typos in a paper), if the user must search a list (as in a list of phone numbers) to place in some other text, if the task requires the user to get help information, or if the task requires cognitive understanding of several pieces of information from two different files. Some window attributes which may prove useful include the size and number of windows displayed (Bury, Davies & Darnell, 1985), the presence of a status line and the information displayed on it, the ease with which windows may be opened, closed, moved, or made larger or smaller (Bury et al., 1985), and how well the user can keep track of the information in each of the windows (Woods, 1984). Finally, some personal attributes which may determine how useful windows are to each individual include different cognitive styles such as field dependence or independence, holist or serialist strategies, convergent or divergent thinking and reflective or impulsive acting (Robertson, 1985), the operators' spatial memory and logical reasoning as well as age (Gomez, Egan & Bowers, 1986) and experience (Scapin, 1981).

This experiment was designed to answer two main questions with regard to performance using windows. The subjects performed simple editing tasks in which they were required to find information from a supplementary file and type it in to a main file. The first question was: Why would performance using a window environment be better than a non-window environment? This question was answered by comparing performance for a group of subjects using a window environment to performance for three groups of subjects using one of three non-window strategies. The three non-window strategies included 1) Memorization or memorizing the information, 2) Notes or taking notes on the information, and 3) Switch or switching between files. These three strategies required different amounts of the working memory and response selection resources. Therefore, by comparing window performance to performance using one of these three non-window strategies, it was thought possible to determine the degree that differences were due to working memory and response selection resources.

The second question was: When is performance using a window environment better than a non-window environment? This question was answered in relation to two task parameters: the amount of information (memory load) which each subject was required to find and the displayed complexity of this information (complexity). This question provided information as to what types of tasks are improved by the use of windows.

It was expected that the answers to these questions were not independent of each other. Rather, it was proposed that there would be an interaction between environment and memory load and between environment and complexity for task completion times. It was thought that performance between the environments should not differ when memory load and task complexity are low. However, as the user's resources are diminished by higher memory loads and more complexity, it was thought that differences between the environments would become apparent. These differences suggest reasons why window performance may or may not be useful. The literature review is presented below and discusses the resources involved in using windows as well as review two studies which compared window and non-window performance.

LITERATURE REVIEW

Overview

Six main topics are discussed in this section. The first topic is an introduction into window design. Following this is a review of multiple resource theory. Multiple resource theory allows us to hypothesize that both working memory and response selection are important in making windows useful. Then a review of the major concepts in working memory is presented. An examination of response selection with regard to typing performance follows. Mental workload, which examines how taxed a user is, is then discussed. Finally, two studies which compared window and non-window performance are critiqued.

WINDOW DESIGN

The term 'window' is defined as a viewport to a usually larger set of information. In the simplest case, the entire computer screen may be thought of as a window which allows the user to view only one application at a time. In fact, Bury, Boyle, Evey and Neal (1982) found that novice users perceived a cathode ray tube (CRT) screen as a window to view a larger document rather than simply a scrolling device for a document. Usually, however, one is interested in multiple windows. In this case, a second, third or fourth (ad infinitum) window may be placed on the computer screen. Each of these windows may contain separate and distinct applications such as documents, spreadsheets, graphics, etc.

The full application that each window represents is called a Presentation Space (abbreviated "P-Space"). The information from the P-Space that is presented to the user in the window is referred to as an object. The virtual desktop is a representation of all of the objects open to a user. In most instances, the physical display is as large as the virtual desktop. Thus, all of the objects which a user is manipulating are located in the space of the physical display. However, it is possible that the virtual desktop is larger than the physical display and the display itself becomes a window to a larger set of windows (Bury, et al., 1985).

Many window systems allow the user to modify the window in several different ways. This includes opening and closing the window, placing one window on top of the others, making a window larger or smaller and moving the window around the physical display. This allows the user to customize the window setup for particular needs. Some window systems, however, allow only minimal modification or none at all. Thus when the window is opened both its size and placement cannot be changed. These systems are used primarily by menu, message and help windows and free up the operator from making window placement, size, etc. decisions (Bury et al., 1985).

The window system which was used in this study allowed the user to open and close only one window at a time which was of a fixed placement and size. The user could move the cursor between the windows and had full editing capabilities in both windows. This system was chosen because the number of decisions the user was required to make was kept to a similar number as that of the non-window environments. By controlling the window system in this manner, differences may be accounted for among the window and non-window environments in terms of responses executed.

RESOURCE THEORIES

Human performance, including the use of computers, relies on the execution of many different mental operations. These operations include rehearsal, inputting, responding, transformation, encoding, etc. There have been a number of theories which have been proposed to explain and define the limitations of the human in processing different signals. These theories of mental resources are important because the usefulness of windows for computers lies in their ability to reduce the cognitive load placed on the user. This is significant because humans are thought to have mental resources which are limited in capacity. A review of resource theory is presented below. This review is useful because it demonstrates the capabilities of the human as far as performing two tasks at once. Thus, the multiple resource theory can help in understanding how performance can be improved or degraded by increasing the amount of cognitive or physical work that must be completed.

Single Resource Theories

The first theories of mental resources viewed the human as a single limited capacity channel (Broadbent, 1958; Moray, 1967). Any signal which was received by the operator was thought to have complete control of the channel from selection of the signal to completion of a response. If two signals were presented at the same time, only one could be selected and acted on. The other could be filtered out or held in store until the channel was clear. This theory predicted that the time to complete two tasks simultaneously equals or exceeds the sum of the time required by each task independently (serial processing). However, there are many occasions when this is not true (Kerr, 1973).

Kahneman (1973) viewed the human as possessing an undifferentiated limited capacity of resources which could be allocated to any number of tasks as long as the total capacity was not exceeded (parallel processing). All operations were thought to require some amount of this limited undifferentiated resource. Allport, Antonis and Reynolds (1972) developed a multichannel theory which suggested that there were several different processors each with a limited capacity. Tasks which used different resources could be processed in parallel. However, those tasks which did require the same limited resource were thought to be processed serially (Kerr, 1973).

Multiple Resource Theories

The theories discussed above assume that the human has an undifferentiated limited capacity single resource (Kahneman, 1973). They cannot, however, account for several research findings. First, the single resource theories suggest that a task of greater difficulty, performed at the same level as a different task of lesser difficulty, should require more resources. However, this does not always occur. Instead, there are many examples of interference not by task difficulty, but by the task structure (Wickens, 1984). For instance, Wickens (1976) found that maintaining constant pressure on a stick (a pure response task) interfered more with performance on a manual tracking task than an auditory signal detection task which was judged to be more difficult by subjects. Second, according to single resource theory, if the difficulty of the same task is changed, the performance of a second task should also change, while holding allocation constant. Nevertheless, Gopher, Brickner and Navon (1982), found when using a letter typing task and a two dimensional tracking task that two levels of the cognitive component of the typing task did not affect the tracking task. However, both the typing and tracking tasks shared a motor resource component. Third, it is also possible for some tasks to be perfectly time shared so that no decrement in performance will occur. Thus, Allport, Antonis and Reynolds (1972) found that subjects could sight-read music and perform a verbal shadowing task perfectly.

These results suggest that there may not be one single resource but rather there may be many types of resources. Wickens (1980) suggested a model with three dichotomous dimensions (or structures) on which tasks may differ. Within a structure, resources are limited. However, between structures, resources are considered unlimited because different resources are used. The more two tasks require separate resources on any of these dimensions, the less changes in the difficulty on one task will influence changes in the difficulty on the other task. The first of the three dimensions is the stages of mental processing. In this dimension, both encoding and central processing are thought to require the same resources, but different resources from responding. Encoding and central processing occur in working memory while responding is thought not to (Wickens, 1984). Therefore, two tasks which both contain heavy encoding and central processing components will interfere more with each other than with a task which contains a heavy responding component.

The second dimension is that of the modalities of input and output. The input modalities include visual and auditory encoding. Two tasks which require two visual inputs will be less efficiently timeshared than if one is an auditory input. The output modalities include vocal and manual responses. Two tasks which require vocal responses will be less efficiently timeshared than if one of the tasks requires manual responding. The final dimension is the code of central processing; whether information in memory is coded spatially or verbally. Wickens, Mountford and Schreiner (1981) cite previous research on brain lateralization asserting that verbal processing may be associated with the left cerebral hemisphere of right handed individuals while spatial processing is localized in the right hemisphere. Herdman and Friedman (1985) found evidence to suggest that the left and right hemispheres of the brain do indeed behave as though they represent separate resources.

Tests of Wickens' model for multiple resources have received much empirical support (Gopher et al., 1982; Kramer, Wickens & Donchin, 1985; and Wickens, 1980). In the present experiment, input was necessarily visual (CRT display) and output was necessarily manual (keyboard typing) and the code of central processing was verbal although a spatial component was also expected since the ability to use a window may relate to spatial ability (Gomez, et al., 1986). Of interest, however, were the stages of mental processing.

As was discussed earlier, it is thought that windows will be more useful than non-windows when the windows 1) reduce the number and type of responses which must be made and 2) reduce the amount of information which must be remembered. Given that resources within the encoding and central processing stages are thought to be limited and occur within working memory, then windows will aid performance for tasks which exceed the capacity of memory by acting as a memory aid and reducing the demand on resources. Instead of having to remember information, the user can view the actual information. Windows should also reduce the responding stage of performance because the user need not write information down in order to remember it, or use many keystrokes to get in and out of files. Therefore, when the task requires a great deal of responding, performance will be better in an environment which requires little responding as opposed to one which requires more responding. Also, as central processing and encoding are thought to require different resources than responding, there should be no interaction between the amount of responding (keystrokes) and the amount of central processing and encoding (the amount of information which must be remembered and understood) in a task.

Working memory is the place where central processing and encoding occur. The next section examines working memory in detail and discusses the manner in which it will affect performance in both the window and non-window environments.

HUMAN WORKING MEMORY AND WINDOWING

A person's working memory is a construct which refers to a part of memory that holds either new information (such as a telephone number) or old information retrieved from long term memory (such as your own telephone number) which can then be acted on. As working memory is limited in its capacity, performance will degrade (be slower with more errors) when a task requires more working memory than can be allocated. However, any method which can reduce the memory load on an individual has the ability to increase task performance. Windows offer such a solution. Understanding the nature of working memory is therefore critical to understanding when and how windows will be useful.

One of the main attributes of working memory is a very limited capacity. Miller (1956) found that immediate memory (referred to here as working memory) could hold 7 +/- 2 'chunks' of information. A 'chunk' is any highly familiar unit which may be very individualized and depends on what is stored in long term memory. Thus, what may be a chunk to one person, may not be one to another person (deGroot, 1966). Although this definition is very vague, it is also useful. It can be used to explain why the same person can apparently hold very different amounts of different types of information in working memory. Thus, a secretary may be able to easily and correctly type, in a letter, three names of people who work in her office. However, she may not be able to remember or correctly type three names of other people with whom she is not familiar.

Windows, however, may free a user from filling up working memory. A user may find information in a window, place it in working memory, and then type the information into a main application. If the information is greater than 7 +/- 2 chunks, the user can take another glance, fill up working memory and type the information into the main application. This will be done until the task is completed. In the switch and memorization non-windowed conditions, these actions may also be performed. However, the user must hold the information in working memory for a longer period of time which brings us to the second aspect of working memory.

The second aspect of working memory is that it is not permanent. Forgetting can occur in two ways. First, items in working memory simply decay over time. Thus, time itself, causes forgetting (Reitman, 1974). However, in most circumstances, this method is not as important for forgetting as is the second method known as interference. The interference theory of forgetting states that activity or events occur, over time, to interfere with memory. According to this theory, there are two types of interference. The first, retroactive interference, is defined as a decrement in recall produced by events occurring between learning and recall (Adams, 1980). Thus, in the switch and memorization non-windowed conditions, a user may have to find and remember a certain name. Once the user has found this name, the name is placed in working memory. Then the user must get back into the main application and type the name into some text. However, the activity of getting back into the main application may produce retroactive interference, and the name may be lost or altered in memory. This may be less likely to occur in a windowed condition.

The second type of interference, proactive interference, is defined as a decrement in recall produced by events which occurred before the learning of some information or activity (Adams, 1980). Thus, the events were never properly encoded into working memory. Under this type of interference, a phenomenon known as proactive inhibition (PI) has implications for computer window use. Keppel and Underwood (1962) found that subjects, who were required to remember a different nonsense syllable on each of many trials without rehearsal, had poor recall after 18 seconds.

However, they also found that recall for the first three trials was near perfect. It was only after subjects had four or more trials, that recall was reduced. Thus, early items interfered with the recall of later items. This phenomenon occurs because of one of the ways in which items are coded into working memory, namely semantic or meaning based representation. Items which are highly related semantically will tend to interfere with each other by remaining in working memory. Therefore, proactive inhibition will build up over several trials. A release from proactive inhibition can be obtained by switching to a different kind of stimulus, one which has a different semantic meaning (Wickens, 1984).

Therefore, if a user must remember a number of semantically related pieces of information, the user is likely to get a build up of proactive inhibition. If this occurs, the user may have to refer to the information several times. In the switch and memorization non-windowed strategies, this would be very difficult and time consuming. However, in the windowed environment, this would be much easier. It should be noted that this phenomenon should occur when the user is not able to rehearse the items. This may be the case if the user has many things to do at the same time, such as remembering how to get in and out of an application.

Information may be kept in working memory indefinitely as long as the number of chunks to be retained does not exceed the capacity of the system and the information is constantly attended to, usually by rehearsal (Howard, 1983). Peterson and Peterson (1959) found that recall was perfect, up to 18 seconds, for subjects who were allowed to rehearse items (a similar experiment to Keppel and Underwood's 1962 experiment). However, for subjects who performed a distractor test, which prevented rehearsal, the percent recall was perfect immediately after presentation of the words and then dropped dramatically over the next 18 seconds to zero percent recall. Along with this, Melton (1963) found that not only rehearsal, but the number of items, affect recall. The more items subjects were required to remember over a period of time (up to 5 items) the less they could remember over the retention interval. Thus, anything which prevents rehearsal, will reduce one's ability to hold information in working memory. A user who must use a complicated word processor may find it difficult to keep information in working memory. A system which does not act as a distractor task will allow for the best performance.

Finally, working memory can be attenuated when certain types of other tasks must be concurrently performed. Baddeley and Hitch (1974) had subjects perform a reasoning task while holding 1, 2, or 6 digits in memory. They found that the reasoning task was unaffected for both speed and accuracy with 1 or 2 digits. However, subjects were significantly slower in the reasoning task with 6 digits in memory. Also, the more difficult the reasoning task was, the greater the effect of the 6 item memory load. Thus, the reasoning task must call upon working memory to some degree in order to get a deficit from holding the six items in working memory. Baddeley et al. (1974) also tested language comprehension in a reading task, and found that comprehension dropped significantly when six digits had to be held in working memory as opposed to 1 or 3 digits. Shulman and Greenberg (1971) also found that performance on two types of perceptual tasks was inversely related to memory load. Therefore, the more information a user must hold in working memory, the fewer resources are available from working memory to be used to process reasoning, language comprehension and perceptual tasks. Accordingly, these processes must all draw from the same resource.

A window then, may be useful by providing the operator with a way to reduce the number of items which has to be held in working memory. The fewer items held in working memory, the more room there will be for other types of processes which draw from the same resource. A user need only hold a few items in memory at any one time. After the information has been used, the user may then refer back down to the window to get more information. In the switch and memorization non-windowed strategies, this may also be done, except that the very actions of getting in and out of applications may reduce working memory capacity.

As shown above, windows may aid the user by reducing the load on human working memory. It is also possible that the responding stage of processing will also affect the usefulness of windows. The next section examines this possibility.

RESPONSE SELECTION AND EXECUTION OF KEYSTROKES

Typing, as a transcription task, is a very complex series of motor movements which occur in a highly precise manner. Typically, secretaries can type between 60 and 80 words per minute (wpm), 100 wpm is not uncommon, and championship typing can reach close to 200 wpm (Rumelhart & Norman, 1982; Seibel, 1972). This translates into around five keypresses every second (Alden, Daniels & Kanarick, 1972).

Although the typing speeds just mentioned seem relatively fast, they are measurements taken from the continuous typing of text rather than the discrete typing which occurs in computer text editing tasks. The mere act of pressing a key on a keyboard takes time. Therefore, task performance in a computer system which requires a large number of keystrokes will be slower than one which requires fewer keystrokes, all else held constant. It is apparent, then, that we should examine the selection and execution of keystroke responses in order to make accurate predictions about window and non-window performance.

The psychology of typing has been thoroughly studied (Shaffer & Hardwick, 1970) and has as its basis the understanding of reaction time. In reaction time studies, if two responses occur very close together in time, the second response will be slower than if the same response had occurred alone. This is known as the psychological refractory period (PRP) (Davis, 1965). The PRP is best explained in terms of Broadbent's (1958) single channel theory. According to this theory, processing the stimulus for the first response "captures the single-channel bottleneck of the decision-making/response-selection stage" (Wickens, 1984). In this way, the second stimulus cannot be handled until the first response has been executed. Keele (1973) however, found that the perceptual processing of the second stimulus can occur in an automatic fashion (in parallel) while the first response is still being executed. Therefore, perceptual processing of the second stimulus is free. The bottleneck occurs after this free processing and continues until the first response is completed.

Because of this bottleneck, the latency of several single responses, strung together and occurring very close in time (serial reaction time) will be more than the sum of the individual responses occurring alone.

Thus, the motor activity of responding is slower than the perceptual processing. In normal continuous typing, this problem is counteracted by the use of preview and lag. Since perceptual processing is faster and can occur in parallel with responding, the user may 'preview' or see several letters ahead of what is being typed. Thus, the response of each letter typed may 'lag' or fall behind what is being seen (Shaffer & Hardwick, 1970). The information that is previewed is held in working memory until it is required. Shaffer & Hardwick (1970) state that after stimuli have been perceived, they are translated into response codes and sent in a queue to working memory to await output. Wickens (1984) points out that although the information is held in working memory, it is only for a short time (around one second). Thus, as the information is held in working memory for such a short time, rehearsal, which demands resources, is not used.

The kind of typing which has been discussed so far is called continuous typing because input is continuously encoded from some medium and output is continuously being typed (Seibel, 1972). In this situation, the typist simply transcribes information. However, with the advent of computers, a second form of typing has recently been studied. Discrete typing reflects the rapid typing of a series of letters, numbers, or key sequences. It is called discrete because the information is typed in discrete units rather than one continuous flow. Editing tasks are representative of this kind of typing. The typist may only type one word on a line or a series of keys may be struck which delete a line.

As was stated earlier, stimuli can be encoded faster than they can be responded to. This leads to the decision complexity advantage. This advantage states that one can respond to several more complex decisions more rapidly (requiring fewer keystrokes) than many simple decisions requiring many responses (Wickens, 1984). Hence, in typing, one could type more information but less keystrokes by using keys which represent more information (in bits). Stenotype machines used by court stenographers convey more information per keystroke than the conventional typewriter. The keystroke rate for the stenographers is slower than for individuals using the QWERTY keyboard, but, they can type 200-300 wpm because of the increased information transmission rate. This concept is important in deciding upon editing commands for word processors. For instance, the decision complexity advantage indicates that commands should contain few keystrokes yet each keystroke should transmit more information than the keystrokes for normal text.

Another major factor in reaction time is the complexity of the response. A response that is more complex will take longer to execute, will take longer to choose, and will require more feedback (Wickens, 1984). Therefore, the initiation of a complex response will take longer than a simpler one, will physically take longer to execute and the operator must monitor the response more carefully before continuing with the next response. Sternberg, Knoll and Wright (1977) found that response time (per keystroke) was faster if letter strings were not only short but required at least one hand change. This is because, while one hand is executing a response, the other may be getting ready for the next response. However, as the number of hand changes and string length increased, response time began to increase. Therefore, response time is a function of both the types of finger movements as well as the length of the string to be typed.

Consequently, response sequences for computer commands should not only convey a great deal of information per keystroke, but the keystrokes themselves should not be so complex that they will take unreasonably long to initiate or execute. The user will probably always monitor an editing keystroke, especially if it is a keystroke which will delete a line or a word. In discrete tasks, the use of preview will probably not play a major role in overcoming monitoring time. However, if the user can quickly recover from any major errors, the user may not have to monitor a major editing keystroke as carefully as in the case when the error is unrecoverable.

This discussion of reaction time and typing has several implications for deciding command sequences for windows in editors. First, the command sequence required to open, close and otherwise use a window (or for that matter any editing task) should be short and not complex. Second, the key sequence should convey a lot of information per keystroke. Third, if more than one keystroke is required, it should contain at least one hand change. Fourth, the command should be able to be quickly undone. Thus, the key sequence for the editing command will be learned and produced quicker if it is a short mnemonic for what it actually does and can be undone quickly (Embley and Nagy, 1981).

The discussion also has implications for determining how windows can aid performance in typing tasks. First, as the number of responses increase, the time to complete the task will also increase. Thus, suppose a user needs to locate some information in a supplementary file. If the user was in a simple window environment, the user might have to execute two keystrokes to call up a window and find the information. However, if the user was in a non-window environment in which he or she had to switch back and forth between the supplementary file and the main file, the user might need to execute eight keystrokes in order to accomplish the same task. The sheer number of keystrokes would take longer to execute in the second case. Not to mention that the information would have to be remembered and rehearsed for a relatively long period of time, compared to the windowed situation.

In the example above, the user could also write down (take notes on) the information. Thus, the user would need to execute more responses than in the window condition. Although information can be handwritten at a rate of about 80 characters per minute (Seibel, 1972) this will add time, especially since this intermediate step can be skipped in the windowed situation.

Another implication of the typing research for windowing is that of preview. In the window situation, the user could read from one window, encoding information at a rapid pace, while typing that information into another window. In the non-window switch and memorization conditions, the user must rehearse the information obtained from the other file to hold it in working memory. This act limits the amount of information which can be held in memory. The user could write the information from the second file down as in the notes condition. However, the act of writing will slow the user down relative to the user in the windowed environment.

Along these same lines, if the information which the user must obtain from the other file is difficult to remember or not easily chunked, the effect of preview allows the user in the windowed situation to continue executing responses from the buffer of working memory while encoding the difficult information. In the non-windowed environment this cannot be performed.

Finally, it should be pointed out that it is quite possible for performance in the non-windowed environment to be faster than in the windowed environment. This could occur in at least three situations. First, if more responses are required to open up or adjust the windows, locate or move text, page up and down, rather than simply to get out of the application, find the information and get back in the application, then windows may attenuate performance. Second, if the total number of responses required to use the window system is more complicated than the non-windowed situation, windows may hinder performance. Third, if windows reduce the amount of information which may be previewed, performance will be degraded.

MENTAL WORKLOAD

The concept of mental workload may be very advantageous in determining the usefulness of windows. There is no single definition of mental workload. Generally, it is the idea that one can measure how busy an operator is. Mental workload can give estimates of how much a system is taxing an operator or the extent of work an operator can handle. Workload measurements can therefore be used to compare the effects and interactions of different tasks and computer systems on cognitive capacity.

It is conceivable that subjects will have more reserve cognitive capacity in a window environment than in a non-window environment. This could occur because subjects may not have as much to do in terms of responses or in terms of working memory as in the non-window case. It is also conceivable that workload may be the same in the window and non-window conditions, yet the task is performed faster in the window environment. If this occurs, the operator is more efficient in the window environment in that he or she is working quickly to complete the task and is not slowed down by many extraneous tasks such as extra response movements or extreme memory loading which may occur in non-window environments. In the present experiment, two workload measures were used to obtain estimates of reserve cognitive capacity which would allow us to better understand the effects of windows on performance. Therefore, a discussion of the different workload measures and their effectiveness at discriminating between different task levels is presented below.

Wickens (1984) discussed five criteria that workload measures should strive to achieve. The first, sensitivity, suggests that a measure should be sensitive to changes in task difficulty or resource demand. The second, diagnosticity, indicates that a measure should indicate what caused the change in workload. Third, selectivity refers to a measure which reflects changes in resource demands which are related to mental workload and not to changes which do not relate to mental capacity. Fourth, the measure should also not be intrusive. Finally, the measure should be reliable. Wickens also noted that most workload measures will not satisfy all of these criteria.

A number of different methods have been proposed to measure workload including using secondary tasks to assess reserve capacity (Hicks & Wierwille, 1979; Ogden, Levine, and Eisner 1979; Savage, Wierwille & Cordes, 1978) and measuring physiological changes of subjects while performing and not performing tasks (Casali & Wierwille, 1984; Hicks & Wierwille, 1979). The two methods which were used in this experiment are primary task measures and subjective questionnaires.

Primary task measures assess workload by increasing the difficulty of the primary task and assessing changes in performance. Its two major advantages are that it is unobtrusive and that the measures relate directly to the task. The major drawbacks of this technique are that it may be insensitive to changes (Hicks & Wierwille, 1979), it is difficult to compare diverse measures between tasks, and it is difficult to say how task difficulty, as defined by the experimenter, will load the subject (Wickens, 1984). Thus, unless the same dependent measures are used in experiments, one cannot compare tasks. Even if the same dependent measure is used, different operators may allocate resources differently, such that one operator may maintain maximum performance, while another operator may not maintain maximum performance throughout all levels of the task (Wickens, 1984).

The second type of measure which was used in this study involved assessing subjective mental workload with questionnaires designed to determine how much an operator was mentally taxed. This technique is simple to use and unobtrusive. Nevertheless, it is possible for the subject to confuse mental and physical workload, and under or overestimate the degree to which he or she was actually loaded. The next two sections evaluate, in more detail, the primary workload method and questionnaire which were used in this study.

Primary Workload

The primary workload technique may be performed by changing the difficulty of the computer task. This technique was performed by Bury, et al. (1985). They compared two levels of difficulty (high and low) for users performing several tasks which included a help task, a disk utility task and an editor task. Memory load was varied to create primary task loading, and task completion time was the measure of primary task difficulty. Thus, in the low memory load condition, subjects had to remember only a small amount of information. Whereas in the high memory load condition, subjects were required to remember a great deal of information. They found a main effect for difficulty meaning this primary measure was sensitive to task loading. They did not find an interaction of memory load with environment. Finding an interaction would have indicated that as mental workload increases (in terms of memory) performance in the two environments would differ. The expected difference would be that performance in the window environment would be faster than the non-window environment as memory load is increased. Although the primary task measure was found to be sensitive to workload changes, it may not have been diagnostic, or able to indicate where the workload changes occurred. It is also possible that the results were true and both environments require equal amounts of mental capacity.

It is quite likely that both environments required the same capacity, because subjects were found to have executed a large number of responses in the window environment. If this was the case, then primary task reaction time may be a discriminable workload measure and would be useful in the present study.

MCH Questionnaire

Human performance has often been assessed in terms of a taxonomy developed by Berliner, Angell, and Shearer (1964). They stated that human performance can be divided into four main processes: perceptual, mediational, communications and motor. Perceptual processes included searching for and receiving information and identifying objects, actions and events. Mediational processes included information processing such as computing, categorizing and translating as well as problem solving and decision making such as analyzing, choosing, and comparing. Communication processes involved answering, directing, instructing and requesting. Also motor processes consisted of simple or discrete motor activities and complex or continuous activities.

The editing task which was used in the present study was expected to affect two components of human performance. The first was psychomotor activity and the second was mediational processes. A set of four studies (Casali & Wierwille, 1983; Casali & Wierwille, 1984; Wierwille & Conner, 1983; and Wierwille, Rahimi & Casali, 1985) were performed to systematically evaluate over 20 workload techniques when the operator has been loaded in one of the four categories of human performance. Two subjective workload questionnaires, including the Modified Cooper-Harper workload scale, were used in each of these studies. With the exception of one non-significant result, each of the scales was able to significantly differentiate between primary task loadings in a relatively monotonic manner.

The finding that the opinion scales were useful in differentiating workload level in the four studies reviewed above, is consistent with a number of other studies using other tasks including driving simulators (Hicks & Wierwille, 1979) and tracking tasks (Hess, 1973). The use of rating scales is easy to obtain, unobtrusive to the primary task, and yet so powerful in detecting changes in workload that it would be remiss not to use one in any study of workload.

One of the most commonly used questionnaires is the Cooper-Harper scale. This scale, which is arranged in a 10 point decision tree format, was developed to assess aircraft handling qualities. However, because it was specifically designed for aircrews, Wierwille and Casali (1983) discuss a revised scale which can be used for almost any man/machine system. The revised scale is called the Modified Cooper-Harper (MCH) scale. This scale also ranges from 1 - 'operator mental effort is minimal and desired performance is easily obtained' to 10 - 'instructed task cannot be accomplished reliably'. An example of this scale is shown in Appendix E. This scale has been found to significantly relate to workload in a number of studies (Casali & Wierwille, 1983; Casali & Wierwille, 1984; Skipper, Reiger, & Wierwille, 1986).

There were several reasons for choosing the MCH scale over others. First, it may be used in its present form with no modifications. In using a modified scale which has not been tested, one is always unsure of the scales psychometric properties. Second, the scale has been validated a number of times (Casali & Wierwille, 1983; Casali, Wierwille, 1984; Skipper et al., 1986). Third, various versions of the MCH have been tested to determine which scale is most sensitive and the original version was found to be the best (Skipper et al., 1986). Finally, the MCH or Cooper-Harper has been found to reliably discriminate between workload loadings on all four types of human processes.

REVIEW OF TWO WINDOW STUDIES

To date there have been very few studies which have examined subjects' performance using windows. Two studies are reviewed here because of their pertinence to the present experiment. The first study (Silver, 1985) was concerned with comparing the placement of a source manuscript for an editing task. Subjects used a CRT as the target manuscript to make editing changes. Comparisons were made between two different placements of a hardcopy source manuscript and using the bottom of a split screen (window) as the source manuscript. The second study (Bury et al., 1985) directly compared subjects performance on three different types of tasks using both a windowed and non-windowed environment. Subjects were asked to complete tasks which required the use of a help window, a directory window or a second editing window. Both of these studies failed to find that performance was better using windows than not using windows. These studies are examined below in more detail and possible reasons for the null results are discussed.

Review of Silver (1985)

Silver (1985) thought that some of the deleterious effects of headaches and eyestrain which have been reported in the literature (Isenee and Bennett, 1983; Starr, 1984) may have been due to the simultaneous use of both hardcopy and CRT displays during text editing. She indicated that the user must adapt to the use of two completely different mediums requiring different luminances, contrasts, visual angles, as well as having different type fonts, pitches, line spacing, margins and different viewing distances. She also felt that the placement of hardcopy by most users created unnecessary neck, head and eye movements which would not only take time but lead to muscle fatigue and discomfort.

The experiment required subjects to read from a document on which changes had been marked (the source manuscript) and then delete certain words in a target manuscript located on the CRT. Silver (1985) compared three conditions for placement of the source manuscript. In the Hardcopy Random Placement condition the source manuscript was a hardcopy placed anywhere the subject wanted. In the Hardcopy Fixed Placement condition, the source document was placed on a document holder which minimized neck and head movements. Finally, in the Split Screen (window) condition, the source document was located in the lower half of the CRT display. In this way, subjects paged up or down in either the source or target document to make the required changes. Silver hypothesized that the window method would be superior in performance to the other two methods because the window would produce less stress on the ocular-visual and musculorskeletal systems and thus would make more efficient use of the necessary viewing space.

The subjects in the experiment edited the document using two of the three methods described above. To control for differences due to experience or ability, the order of the method used for editing was counterbalanced. Four groups were therefore created with 12 subjects per group. Subjects received some training prior to the experiment on how to use the computer editor and what the task entailed.

The dependent measure was a productivity ratio of seconds per correction. ANOVA's performed by Silver (1985) indicated there were no significant differences between the source document placement conditions. However, she did find an order effect. Productivity improved significantly in the two groups where the hardcopy followed the window condition as opposed to coming before the window condition. On examining data collected about the user's preferences and reasons for editing methods, she found that 58% of the subjects preferred the window. Half of these subjects indicated that it was easier to see in the window situation.

Silver (1985) pointed out that one possible reason for the null results was the difficulty in paging up and down in the window condition as compared to turning pages of the hardcopy. She indicated that subjects in the window condition had trouble remembering to scroll in the file. Also, scrolling required a three keystroke command sequence. Silver stated that some of the subjects had trouble executing this command. Therefore, Silver hypothesized that the reasons for the null results in the window condition could be because the time required to execute scrolling responses equaled the time saved by not having to exert many visual and muscular movements.

In light of the discussion of typing performance and reaction time, Silver (1985) may be correct in her assumption that the motor and cognitive aspects of the window task attenuated performance. Silver thought that page turning in the hardcopy conditions required less cognitive and motor capacity than executing the three key sequence necessary to scroll down in the window. This assumption was not tested in any manner. However, it seems reasonable to assume that the action of turning a page can be executed with less cognitive capacity than executing a relatively foreign key sequence. Also, as was discussed above, the more complex the key sequence, the more time it would take a user to initiate the sequence, to complete the sequence, and to monitor the action of the sequence. Therefore, for subjects who were not well practiced on using windows as well as executing the scroll command, it is possible that scrolling did take as long or longer than turning a page.

The finding that there were order effects with superior performance when the hardcopy condition was preceded by a window condition indicates that there were differential learning effects. This could be caused by the fact that the subjects did not receive any actual training on the task they were to perform. Thus, they were learning the task while in the first condition. Subjects who began the experiment in the window condition may have had a great deal of trouble learning the task at the same time that they were also learning the novel movements involved in executing the scroll command. Hence, learning effects may have also led to the null results.

The task, itself, was amenable to the finding that windows improve performance. This was because the task contained a working memory component. Thus, subjects found incorrect words on the source document, and held the words in working memory while searching for them in the target manuscript. Given that the subject could glance down in the window to find the word, the subject would have to spend less time and capacity rehearsing the word compared to when the word was on a hardcopy manuscript.

One final point about this study which has implications for the present study is that in the window condition the target manuscript was displayed in the top 11 lines of the CRT and the source manuscript was displayed in the bottom 10 lines of the CRT. This is in contrast to both of the hardcopy conditions where the target manuscript was displayed on all 22 lines of the CRT while the source manuscript was presented on regular 8 1/2 by 11 inch pages. Thus, the length of the displayed information was confounded. Search behavior may have been slower because more responses were required, when only 11 lines of information were displayed versus 22. The question of the effects of window size is of importance in the present study.

Review of Bury, Davis & Darnell (1985)

The second study (Bury et al., 1985) is more complicated than the first and directly compares performance on tasks using a window environment versus a non-windowed environment on the CRT. Bury et al. (1985) hypothesized that windows would be useful if the information in a window was supplemental to the primary task and the effect would be strongest as the tasks required more information to be held in working memory.

Eight subjects, in this experiment, performed a total of twelve tasks (6 in a window and 6 in a non-window environment). The tasks the subjects were to perform focused on the modification of a primary text file. The tasks varied on two dimensions: task type and memory load. Task type referred to the three different types of utilities the subject had to access. Four tasks required subjects to access help information, four required the subject to access directory information, and four tasks required subjects to access a secondary text file to edit. Within each of these tasks, memory load was varied such that in the high memory load situation, subjects needed to remember a relatively large amount of information to complete the task. In the low memory load condition, subjects needed only to remember a relatively small amount of information.

The eight subjects were randomly assigned to one of two groups differing only in whether they began in the window or non-window condition first. They then completed six tasks in either the windowed or non-windowed environment and then 6 more tasks in the other environment. Each subject was thoroughly trained before the experiment began.

The window system used in the experiment was a very powerful one allowing great flexibility. Users could open up a maximum of nine windows on the screen at any one time, they could move and change the size of and overlap any window they wished. In order to perform an action on a window such as scroll or edit, the window had to be on top of all the other windows. In other words, no other window could overlap a window the operator was using. The actions performed on windows, such as moving or changing window size, required that subjects use a one letter mnemonic (such as 'M' for move) at the corner of the window and then use the cursor to move, change the size of, ètc. the window and then to finally press enter.

Bury _et al._ (1985) did not find the hypothesized interaction between memory load and environment. Instead, they found significant main effects for environment, memory load and task type. Unfortunately, the main effect of environment indicated that subjects took significantly longer in the windowed environment (335 seconds vs 269 seconds) than in the non-windowed environment to complete the tasks. The high memory load tasks all took significantly longer to perform than the low memory load tasks. The directory task was found to take significantly longer than the other two tasks.

In an effort to determine why subjects completed tasks faster in the non-window environment, Bury _et al._ (1985) categorized the subjects' time (by examining keystroke information) into screen arrangement time, task time, time to first keystroke, and unclassified time. They found that subjects in the windowed condition spent 57% of their time in screen arrangement (192s) and 28% of their time (92s) completing the task with the rest being unclassified time. In the non-windowed environment, however, subjects spent 34% of their time (92s) arranging the screen and 48% (129s) in task time.

It appears from this post hoc analysis (assuming the definition of task time and screen arrangement time is correct) that windows can improve performance, but this effect may be overshadowed by the increased screen arrangement time necessary. In order for windows to be useful and increase productivity, the bottom line is task completion time, not simply time spent on the task itself. Therefore, these results indicate that the use of windows will actually reduce performance. The question of interest is why performance using windows was so poor in this study.

The most obvious reason for the results which Bury _et al._ (1985) found was the sheer number and complication of keystrokes which operators used to open and close, move, etc. windows. As Bury _et al._ indicated, users spent a great deal more time arranging screens. If the number of keystrokes were reduced, the results may have been different. The number of keystrokes could be reduced in at least two areas. First, the number of keystrokes in any one command could be dedicated to a PF (Programmable Function) key requiring only one or a very few keystrokes.

Second, task completion time can be reduced by limiting both the number of windows and the amount the user may modify them. In this way, users cannot spend so much time moving and arranging windows so that needed information can be viewed. The more windows users open, unless they are very familiar with the system, the more likely that they will have to be arranged for optimal viewing. This is especially the case when the windows overlap and important information will tend to be hidden by another window. Therefore, the window system used in this experiment gave the user the capability of executing a large number of keystrokes, many of which were probably unnecessary. These keystrokes took time to execute and hence, slowed the user down.

Reducing the number of windows could also reduce the amount of confusion of information between windows. It is quite likely that the subjects exhibited what Woods (1984) called the 'getting lost' phenomenon. This phenomenon refers to when users do not know their location in the system. Users who create five windows may 'get lost' when windows begin to overlap and they are not sure which window is which or which one they want to be in. This confusion could result in increased task completion time.

The size of the windows may have also affected performance. Windows which are small allow less information to be displayed than windows which are larger. Because less information is displayed, subjects may have to execute more keystrokes than when more information is displayed in a larger window. This is because the user will have to scroll more times in order to see the same amount of information. Bury et al. (1985) found that subjects in the window condition scrolled an average of 2.9 times a task while in the non-windowed environment, they scrolled an average of 1.2 times a task. Therefore, window size forced the users to execute more keystrokes in order to scroll to see the same amount of information as in the non-windowed environment.

Bury et al. (1985) varied the amount of memory load placed on the subjects for the task. There was a significant main effect for memory load indicating that tasks which had a high memory load were performed slower than tasks which had a low memory load. However, this effect may have nothing to do with memory load at all. Windows were thought to reduce the effects of memory load. It was assumed that as memory load increased, windows would be more useful because the subject need not have held as much information in working memory. Thus, one would expect an interaction between memory load and environment. Given that there was no interaction but rather a significant main effect, the same factors were affecting performance in the windowed and non-windowed environments. One of these factors could be that the tasks in the high memory load conditions required more keystrokes or more complicated keystrokes than the tasks in the low memory load conditions. The amount and complication of keystrokes would be similar in both environments resulting in a non-significant interaction. If this is true, then the benefits, if any, of windows may have nothing to do with working memory.

A second possible explanation for the non-significant interaction is that there may have been an interaction between memory load and environment except that it was masked by the large amount of screen arrangement time in the windowed environment compared to the non-windowed environment. This is a very plausible explanation in light of the significant main effect of environment. Unfortunately, Bury et al. (1985) did not have any manipulation check to determine if working memory was really being used and whether it was used more in the high versus low memory load conditions.

Although the two studies just reviewed found that task completion time was not reduced but rather increased when using windows, the question is still open as to whether this is always the case, or whether windows may be useful given the right circumstances. The next section discusses the situations in which windows might improve performance given the task which will be used in the proposed study.

RATIONALE AND HYPOTHESES

The rationale and hypotheses for this study is presented in this section. The rationale focuses on how the operator may perform using window and non-window strategies on the experimental tasks. The hypotheses follow this discussion and then the methods are presented in the next section.

Rationale for Window Performance

The tasks used in this experiment are referred to as 'list tasks'. This type of task is one in which the user is editing a letter or a similar document, and needs some information, such as a person's name, some items from a shopping list or a filename. The information which is needed is arranged in list form in another file. The user must find the information from the list, either place the information in working memory or take some notes on it, and finally type the information in the desired location in the main application. The list task is well suited for this study because the amount of information required to be remembered, the placement of the information and the number of keystrokes executed can all be controlled.

Performance of subjects in a window environment was compared to performance of subjects using one of three non-window strategies. In the window condition, subjects could open a window of a fixed size (10 lines long by 80 columns wide) and fixed placement (in the bottom half of the screen. The subjects had full editing capabilities in both windows and could move between them. In the Notes non-window environment, when subjects accessed the supplementary file, it was placed over the main file. Subjects could open the supplementary file and take notes on anything they wished. In the Switch condition, the subjects could open and close a supplementary file as often as they wanted, but they could not take any notes. In the Memorize condition, the subjects could only access the supplementary file once and they also could not take any notes, thus, they had to memorize any information needed.

These three strategies were thought to offer useful comparisons to the window environment allowing results to be interpreted in terms of memory and response selection and execution capacities. The Window condition had a relatively high response component and an extremely low memory component. The Memorization condition, however, contained a relatively high memory component. Differences between the Window condition and the Memory condition could be interpreted as differences between a relatively high response component (Windows) and a relatively high memory component (Memorization).

The Switch condition had both a relatively high memory and response component. The user had the option of varying the amounts of each component by accessing a file more often and attempting to remember less or vice versa. Differences between the Switch and Window conditions could be interpreted as differences in responses as well as differences in memory. The Notes condition also had a relatively high response component and a low memory component. Differences between the Window and Notes conditions could be interpreted in terms of high numbers of responses for note taking and low memorization requirements.

Generally, it is thought that as the amount of information which must be found is increased, performance using windows will be better than the three non-window strategies. This should be due to the relatively smaller number of responses which had to be executed in the Window condition as compared to the Switch and Notes conditions. The better performance in the Window condition will also be due to the reduction in time and effort required to hold the information in working memory as compared to the Switch and Memorize non-window conditions. The specific manner in which the amount of and types of responses and working memory affect performance in the four conditions is discussed below.

Working memory, as was stated above, holds 7 +/- 2 chunks of information. In the case where the list task is below this capacity, both window and non-window performance should be equal for both speed and accuracy, all else held constant. However, when working memory capacity is taxed, a window system which requires the operator to use less working memory capacity than a non-window strategy will out perform the non-window strategy. The very nature of a window system should require less working memory capacity of the operator than many non-window strategies. In the Window condition, the user need not commit information to memory, rather the operator may use the window as a memory aid. A window can be a very efficient memory aid in that the responses required to make notes are not needed and errors are not made copying from an original to note paper. The user need not waste time committing the information to memory as it is always present. Finally, the information is presented in the original form so the user is less likely to be confused by some other representation of the information.

However, the non-window strategies should require more working memory than the Window condition. For instance, in the Switch strategy the user had to find the information needed from a supplementary file and hold the information in working memory until it could be typed into the main file. If the user could not remember all of the information, the user was allowed to go back to the supplementary file to find and hold more information into working memory. In the Memorize condition, the user did not even have the luxury of accessing the supplementary file a second or third time. Instead, the user was thought to spend time memorizing the information. The Notes condition was expected to require a relatively small amount of working memory capacity because the user was not forced to attempt to commit information to memory, but rather could take and use notes as a memory aid. For this very reason, the Notes condition was thought to require more response capacity than other systems.

As memory load increases, a window strategy should require less responses than some non-window strategies but more than others. In the Window condition, the user was not able to view the entire screen, but rather some amount less than the full screen. Therefore, the user should have to scroll through the file in order to find information which would otherwise have been visible if the whole screen could have been viewed. Also, the user was expected to execute keystrokes in order to move between the two windows.

In the non-window Switch condition, as the amount of information needed increases, the user may have to access the supplementary file several times. These actions require the execution of a large number of keystrokes such as typing in the name of the file. Therefore, task completion time will be attenuated by the extra number of keystrokes. In the Notes condition, the subject may make the responses of taking notes. However, these responses can be executed slightly more quickly than keystroke execution (Seibel, 1972). In the Memorization condition, the user may not have to execute many responses at all. Most of the time will be spent memorizing information. A slightly more detailed discussion as to what subjects are expected to do in the window and three non-window conditions is presented below.

In the situation where the amount of information the operator must find is relatively small and does not exceed working memory, the operator in any non-window system will probably attempt to hold the information in working memory rather than take notes on the information. In this situation, in any non-window system, the user must first exit the main application, enter into the application which contains the information needed, and locate it. These actions require time consuming keystrokes to exit and enter the second environment. Once the information has been located, the user probably will, if the amount of information is small, commit the information to memory. This information must be rehearsed or it will tend to be forgotten. Nevertheless, in a non-window environment, there is plenty of opportunity for distractors. For instance, after the information is found and placed in working memory, the user must exit out of this application and enter back into the main application. Again there are time consuming keystrokes. Also, a great deal of energy must be spent in rehearsing the information.

When a user performs the same task in a window environment, a window may be opened and a list read into the window. These actions also require the execution of keystrokes, possibly as many as in the non-windowed situation. The user must then conduct the same search procedure as executed above. If the information which must be found is spread out in the file (rather than located sequentially) the search may require more keystrokes than in the non-windowed environment because the amount of information displayed will be less. Therefore, the user may have to page down or scroll down more than in the non-window conditions in order to locate information. Once the information is located, however, it need not be held in working memory for nearly as long as in the switch and memorize non-windowed conditions. The user need only hold the information in working memory for as long as it takes to type the information in the main application window. Thus, even if rehearsal is reduced or interrupted to some degree, it is less likely that the information would have been lost from working memory. Also, the number of distractions may be fewer in the window environment. Therefore, it is less likely that forgetting will occur because of interference in the window environment.

So far, only a list task which requires less than the capacity of working memory has been discussed. But what happens in a list task which requires more than the capacity of working memory? This situation may occur when the user must remember a large number of unfamiliar names and type them into a letter. As a task requires more information to be held in working memory, there will be less room for other types of processing. In other words, as more and more chunks of information must be remembered, the less one can attend to other things. Therefore, performance using windows and non-windows may be equal when only one or two chunks of information need to be held in working memory. However, as the number of chunks to be remembered increase, windows will begin to out-perform non-windows. This will occur as long as one can assume that the window system requires less working memory capacity than non-window systems.

When working memory capacity is exceeded in the Notes non-windowed condition, the user may write down the information which is required for the task. The user must either write down all of the information, or else use some sort of short hand or abbreviations. During this writing process, transcription errors may occur, which could affect the accuracy of the task in this non-window environment. Also, this writing will take time and resources from the operator. Once the information is written down, the user must exit out of the list and access the main application. The information will have to be decoded from what was written, and entered into the main application. Transcription errors may also occur at this stage. Errors could be held to a minimum by spending more time in transcribing the information (i.e. one can make the data more legible), which may expend more working memory capacity. At this point, the non-window environment will be similar to the window environment. The information will now be presented on paper next to the main application. Therefore, performance at this part of the task, will take a form similar to the window environment. The user can quickly find the information on the paper and type it into the main application.

The user may also elect to find and hold the information in working memory as in the Switch and Memorization non-window conditions. In the Switch condition the user could do this by entering and exiting the applications several times. This process should require a large number of keystrokes requiring time. Also, the user may have to hold information in working memory for a relatively long period of time and may begin to forget or confuse information. In the Memorization condition, the user will have to spend time memorizing the information. If the information is not properly memorized, the user may be unsure as to what the information was. Also, if the information is similar in semantic context, the user may build up proactive inhibition. As was stated earlier, proactive inhibition occurs when rehearsal is interfered with. Thus, in the non-window environment, when rehearsal may be reduced and inhibited, proactive inhibition may result in poor performance.

In the window environment, the user need not write down the information, because it is displayed at the same time and in a similar location to where the information is needed. Thus, transcription and decoding errors will be kept to a minimum. Instead, the user may fill working memory to capacity and place the information in the main application window. Once the information is placed in the appropriate location, the user may glance down to the list window and fill working memory again. This process can be repeated until the task is completed.

The information required to complete a task only has to be held in working memory for a short period of time (enough time to be typed in). Also, the user should be under no pressure to attempt to keep the information in short term memory. If something is forgotten, a quick glance at the window would aid memory. This cannot occur in the Switch and Memorization non-window conditions.

It is possible that proactive inhibition may also occur in the window environment. Nevertheless, it is unlikely because rehearsal will only have to be performed for a very short period of time and thus may not be disrupted. If however, rehearsal is interfered with, the user will have to spend more time switching between windows. This will slow down performance, but not to the degree that it would attenuate performance in the Switch and Memorize non-window conditions. This is because it will take longer to switch in and out of an application than it will to glance down at a window screen.

In this section, the manner in which performance on the list task should be affected in both window and non-window environments were discussed. As can be seen, as the amount of information required for a task is increased, the window environment should provide the best performance as long as the number and type of keystrokes are kept to a minimal level. As the number of keystrokes increases in the window condition (from scrolling and paging down or moving between windows) and the amount of information required is relatively small it is thought that performance will equal or be poorer than the non-window conditions. The next section presents the hypotheses for the present experiment.

Hypotheses

All of the independent and dependent variables are discussed in more detail in the Methods section. However, a brief summary here will aid in the understanding of the hypotheses. The experiment consisted of four independent variables in a 4 X 3 X (3 X 2) mixed design. The first between factor independent variable was Environment (Window, non-window Switch, non-window Notes and non-window Memorization). The second factor was a Verbal/Spatial blocking factor with three levels (Verbal, Spatial, and Equal). The two within factors were Memory Load with three levels (High, Medium and Low) and Complexity with two levels (High and Low).

Five dependent measures were used in the experiment. They included task completion times, the number of responses made, the number of scrolling keystrokes and the number of errors made. Also the ratings on the Modified Cooper-Harper rating scale (Wierwille & Casali, 1983). were used.

The hypotheses for the present experiment are presented below:

H1. There will be an interaction between Memory Load (High, Medium and Low) and Environment (Window, Switch, Notes, Memorization) when task completion time is used as the dependent measure. This is similar to the hypothesis set forth by Bury et al., (1985) but which was found to be non significant. In this study, different non-window strategies are thought to provide answers as to why no differences were found in the above study. It is expected that in the Low Memory Load condition there will be no difference between environments. However, as Memory Load increases, tasks will be completed faster in the Window condition than in the Memorization condition. The task times for the Notes and Switch conditions should lie in between the Window and Memorization conditions. The differences will be due to the relatively large number of responses in the Switch and Notes conditions as compared to the Window condition when Memory Load increases and to the time spent rehearsing information in the Memorization condition.

H2. There will be an interaction between Complexity (High and Low) and Environment (Window, Switch, Notes, Memorization) when task completion time is used as the dependent measure. It is expected that in the Low Complexity condition there will be no difference between environments. However, in the High Complexity condition, tasks will be completed faster in the Notes and Switch conditions than in the Window or Memorization conditions. In the Window condition keystrokes will be required to scroll through the files to find the information. In the Memorization condition more time will be spent finding and rehearsing the information in the High Complexity condition than in the other three conditions.

H3. There will be a significant interaction between Memory Load and Environment when the number of responses is used as the dependent measure. It is expected that in the Low Memory Load condition, the number of responses will not be different for the Environments. However, as Memory Load increases, it is expected that more responses will be executed in the Switch and Notes conditions than in the Window and Memorization conditions. This is because in the Switch condition, responses will be made accessing the supplementary file and in the Notes condition, responses will be made writing notes. In the Window condition, the only extra keystrokes should be scrolling and moving between windows. In the Memorization condition few keystrokes will be executed.

H4. There will be a significant interaction between Complexity and Environment when number of Scroll Editing Keystrokes is used as the dependent measure. It is expected that in the Low Complexity condition, there will be no difference between the environments for the number of scrolling keystrokes. However, in the High Complexity condition, subjects will scroll more times in the Window condition than in the non-window conditions. This is because less information can be displayed in the Window Environment than in the non-window conditions. Therefore, subjects will have to scroll more often to see the same amount of information in the Window as opposed to the non-window conditions.

H5. There will be a significant interaction between Memory Load and Environment for the number of errors. In the Low Memory Load condition there will be no difference between Environments. As Memory Load increases more errors will be made in the Memorization condition than in the other three conditions. This will occur because every condition, except the Memorization condition, has a memory aid of some type. Thus the user does not have to rely as much on rehearsal and working memory.

H6. There will be a significant interaction for Memory Load and Environment when the Modified Cooper-Harper workload rating scale is used as the dependent measure. In the Low Memory Load condition, there will be no difference between environments. As Memory Load increases, workload will be higher in the Memorization condition than in the Window condition. It is unsure how workload will be affected by the Switch and Notes conditions.

H7. There will be a significant interaction between Complexity and Environment when the Modified Cooper-Harper workload rating scale is used as the dependent measure. In the Low Complexity condition, there will be no difference between environments. In the High Complexity condition, however, workload will be higher in the Memorization condition than in the Window condition. It is unsure how workload will be affected by the Switch and Notes conditions.

METHODS

Overview

The 48 subjects in this experiment were placed in one of four environments. They performed six simple editing tasks which varied in memory load and complexity. Human performance in one windowed environment was compared to performance using three different strategies in a non-windowed environment. The subjects performed the tasks in a balanced Latin squares fashion in order to be able to account for practice effects. Also, to control for possible individual differences related to verbal and spatial abilities, subjects were placed into blocks within each environment based on scores on verbal and spatial tests.

SUBJECTS

Twenty-nine female and 19 male (a total of 48) undergraduate students at Virginia Polytechnic Institute and State University were used in this study. The subjects ranged in age from 18 to 25 years with a mean age of 20.35 years. All subjects were volunteers and were given experimental credit for the successful completion of the experiment. Each subject completed a consent form as well as a background questionnaire referring to his or her knowledge of computers. Subjects also completed a questionnaire relating to how motivated and comfortable with computers they felt at the beginning of the experiment. Examples of these forms are found in Appendix A. All subjects were required to have some minimal level of computer experience. Only subjects who had used a personal computer for more than 10 hours in the last two quarters of school were allowed to participate in this study.

The balanced Latin squares randomization of tasks required that subjects be included in the experiment in groups of six per between cell (or one subject for each unique combination of the within factors). As there were four Environment cells (the blocking factor was not considered in this requirement) a total of 24 subjects were required for this experiment. However, in order to increase the power of the experiment it was decided to use two groups of six subjects in each of the four environment cells or 12 subjects per cell. Thus, for the four environment cells, 48 subjects were required for the experiment.

DESIGN

The experimental design consisted of four factors in a 4 X 3 X (3 X 2) mixed design. The two between factors were Environment (Window, Notes, Switch, and Memorization) and Verbal/Spatial Blocks (Verbal, Spatial, and Equal). The two within factors were Memory Load (High, Medium, and Low) and Complexity (High and Low).

The first factor, Environment, consisted of four levels: Window, Non-Window Notes, Non-Window Switch and Non-Window Memorize. The Windowed condition allowed the user to open one window of a fixed size and screen placement (10 lines by 80 columns in the last 10 lines of the screen) and allowed the cursor to be moved between the two windows. This represented the simplest form of a window environment. The first non-window environment, the Notes condition, allowed the user to take notes on any information needed from a second file. The second non-windowed environment, the Switch condition, allowed the user to open and close a second file as many times as was required to find and remember the information needed to complete the task. The final non-windowed environment, the Memorization condition, required the subject to find and memorize information from a supplementary file which was required to complete the task. Each environment required the same number of keys to open and close the window or non-window screen once.

The second between subjects factor, Verbal/Spatial Blocks, consisted of blocking on a classification of subjects on a test of verbal and spatial ability. It was thought that verbal and spatial ability would relate to the experimental task and that by blocking, equivalent groups could be achieved and a reduction in error variance would result. Thus, this factor consisted of three levels: Subjects who were more verbal than spatial, subjects who were more spatial than verbal, and subjects who were equal on both verbal and spatial ability.

The within subjects factors consisted of the six tasks which the subjects were asked to perform. The six tasks required the subjects to open a main file, such as a memo to a colleague, and then locate information in a supplementary file to type into the main file. Given the environment, the subjects would have to hold information in working memory for different lengths of time and/or would have to execute a number of responses in order to encode the information and then decode the information into the main file.

The six tasks represented combinations of three memory load levels and two task complexity levels. The three memory load levels required subjects to find a small amount of information (2 words), a medium amount of information (4 words) or a large amount of information (8 words). The two task complexity levels called for subjects to find information which was either directly next to each other (low complexity) or to search a list for information which was farther apart (high complexity). The six tasks were presented to the subjects in a Latin Squares order to reduce the possibility of order effects. All of the words which the subject was to type into the main file were controlled for frequency of occurrence and meaningfulness as well as the number of letters.

The dependent measures consisted of the task completion time, the total number of keystrokes, the number of editing keystrokes, the number of errors made, and one workload measure. Task completion time was measured in seconds by the computer, and began after the subject had read a small set of instructions indicating what was to be edited. This ensured that task completion time did not include the time for subjects to read and understand the task. If this time were included, the variability of reading and comprehending the instructions could mask true differences in using the windows. Task completion time ended with the closing of all the windows.

Finally, a workload measure was taken. The Modified Cooper-Harper Rating Scale was used to obtain an estimate of the amount of cognitive effort which was used to complete each task in each of the environments.

APPARATUS

Computer System

IBM PC computers, equipped with 640K of memory, were used in this experiment. The computers drove standard IBM color monitors. This screen permits viewing of 25 lines of text with 80 characters per line. The text characters were represented by an 8 X 8 dot character matrix. Brightness and contrast could be adjusted by the user. Standard 83 key IBM keyboards were used. These keyboards have programmable function (PF) keys located at the left of the basic QWERTY keys. To the right of the QWERTY keys is a number pad with arrows which are used as the cursor keys. Two of the keys also work to allow the user to 'page up' or 'page down' rather than scroll through the text.

Description of Window Software

The word processor that was used in the experiment was developed using Pascal and was based on Borland Software's Turbo Editor Toolbox 'First-Ed' editor (Borland Software, 1985). This package was chosen because it is a full word processor which could be easily changed to meet the requirements of the present study. The word processor (from now on referred to as Windows) is similar to the WordStar package, having many of the same commands.

The Windows software program allowed the experimenter to specify in which of the four environments each individual subject would be placed.

There were five main functions which the program performed. First, the program gave each subject a four minute typing test. Second, the tutorial and all of the tasks were performed in the environment in which the subject was placed. In other words, in the Window condition, the program allowed a window to be opened once for each task. In the Switch condition, the subject opened and closed a second file as often as was necessary. Third, the subject reviewed, at his or her own pace, an on-line tutorial designed for each specific environment. Fourth, the Windows software allowed the user to perform five tutorial tasks, three criterion tasks and six experimental tasks. Finally, the Windows software collected the dependent measures and placed them in two separate files.

Typing Test. The typing test required the subject to type as many words as fast and accurately as possible in four minutes. The text for the test was similar to those used in Lojko (1972). The computer displayed one line of text at a time in the center of the screen and the subject began typing below the displayed text. After typing the line and pressing the Return key the next line of text would be displayed. This continued until the four minutes had passed. The computer counted only the number of words which were correctly typed. Speed was computed by dividing the total number of correctly typed words in the four minutes by 4, thus the number of words per minute (WPM) was obtained.

Environments. The Windows software allowed the experimenter to choose any one of four different environments for the subject. The Windows program allowed the user to execute very few file maintenance routines. The routines the user could execute were controlled by a single PF key in all of the conditions. In this way, subjects did not have to learn complicated syntax or keystroke sequences to perform actions such as opening and closing windows. This was thought to reduce the amount of time required to learn the software package as well as keep the memory component for file manipulation keys to a minimum. If an incorrect PF key was pressed (such as trying to close a window which was not open) an error message was displayed and by pressing the Escape key, the subject could erase the message and press the correct key. The six possible file maintenance routines were:

1. PF1: Open the main file - Window 1.
2. PF2: Close the main file - Window 1.
3. PF3: Open the supplementary file - Window 2.
4. PF4: Close the supplementary file - Window 2.
5. PF9: Move to Window 1 - Window condition only.
6. PF10: Move to Window 2 - Window condition only.

The Windows software was a full text editor allowing the user to move the cursor and type anywhere on the screen as well as page up and down to see information not yet displayed in a file. The user could insert text and delete text anywhere desired. In all of the environments, the program allowed the user to open a main window 25 lines long including a one line status line. The status line indicated the line and column the cursor was on as well as whether or not the Insert and Capital Lock keys were on. Also, the name of the file was displayed. Upon pressing PF1, the computer would ask the name of the file to be place in Window 1. The user would type the name in and after pressing return, the computer would place this file in the window.

The difference between the window and non-window environments was the size of the window used to display the supplementary information. All of the tasks required the subjects to access information from a supplementary file. In the Window condition, the supplementary file was placed in a window (Window 2) 10 lines long. In the non-window conditions, the supplementary file was placed in a window (Window 2) 23 lines long. In this way, in the non-window conditions, the subject could not see information from the main file. In all of the conditions, except the Switch condition, the subject could only open the second window once. This forced the subjects into using the strategies desired for each condition. In all of the environments, PF3 and PF4 opened and closed respectively the supplementary files. In all of the environments, when Window 2 was closed the information underneath it (the main file) was uncovered. In this way, in the non-windowed conditions, the user could finish the given task.

In the Window condition, pressing PF3 opened a second window in which the supplementary file was placed. Window 2 covered the information in the last 10 lines of Window 1. Window 2 was of a fixed size, 10 lines long including one status line and 80 columns wide. The status line was exactly the same as for Window 1. Upon opening Window 2, the computer would ask the name of the file to place into this window. Only in this condition could subjects use the PF9 and PF10 keys. The PF9 key moved the cursor to Window 1 and allowed the user to page up and down in the document in Window 1. The PF10 key moved the cursor to Window 2 and allowed the user to page up and down in this document. Full editing functions were capable in both windows. Window 2 could be opened and closed only once for each task. In this way, subjects were forced into the problems of not being able to view an entire 24 lines of text.

The Windows software performed the same way in both the Notes and Memorization conditions. The difference between these conditions was whether information was memorized or notes were taken. In these two conditions, pressing the PF3 key opened a window 23 lines long including a status line exactly the same as in Window 1. The new window, Window 2, covered all of the information displayed in the main window and was used to display the information from the supplementary file. The user could not see any information from the main file, only information from the supplementary file could be viewed. All of the editing functions allowed in Window 1 were also allowed in this window. Again, as in the window condition, the subject could only open Window 2 once. If the user opened, closed and then tried to open Window 2 a second time, an error message would result. Again, this forced the users to perform in the condition as they had been instructed.

The software for the Switch condition worked exactly as in the Notes and Memorization conditions except that the user could open and close Window 2 as often as required. In this way, the user did not have to commit all of the information at one time to memory.

Tutorial. An online, noninteractive tutorial program ran under the Windows software. This program was self-paced and individualized for the four conditions. General instructions and explanations of the status line, PF keys, etc. were first displayed, and then the computer performed a simple demonstration task for the subjects. The instructions and demonstrations for all of the conditions were exactly the same except for sections which were changed to reflect specific differences between conditions.

The major difference in the tutorial instructions was the display of window 2 in the Window condition and the display of window 2 in the three non-window conditions. In the Window condition, window 2 was displayed in the last 10 lines of the screen. However, in the three non-window environments, window 2 was portrayed as covering window 1.

The instructions in the Window condition explained and demonstrated the use of windows as well as how to move between the windows. In the three non-window environments, the use of non-windows was also explained and demonstrated. The instructions for all three non-window conditions were the same except for two paragraphs which were written for the specific non-window conditions. These paragraphs emphasized the strategies which were to be used in the experiment (memorize, take notes or switch between files). The instructions for the each of the tutorial conditions are found in Appendix B.

Software Collection of Dependent Measures. The Windows program allowed for the automatic timing of when a task was begun, when a window was opened and closed, and when the task was completed. The amount of time, in seconds, which a user spent on a task could then easily be derived. Counts of the number of alphabetic keys and the number of various editing keys were also collected.

Also, at various times the users might encounter an error related to opening and closing the windows such as trying to open a third window when two were already open. The user had to clear the error message with the ESCAPE key. A time could also be derived for how long it took the user to clear the error message. Finally, the program allowed for a count of how many errors a user made in the task.

The dependent measures were placed in to two separate data files. One file contained individual keystroke data. In this file, each time a PF key, editing key or scrolling key was pressed the time it was pressed was recorded. The second file contained summary data which included a count of all alphabetic keys pressed, a count of the number of times each editing or scrolling key was pressed as well as the total time spent performing each task, and the number of incorrect PF keys pressed. Finally, each of the actual files which the subjects used was saved. This allowed the examination of the files for errors.

Initial Written Instructions

A two page set of written instructions for each of the four environments was developed for the experiment. The instructions gave initial information as to the nature of the experiment and what was expected of the subjects. The only difference between the four sets of instructions was an explanation of the strategies which were to be used in completing the tasks. The instructions for the four environments are found in Appendix C.

Tasks

Instructions for Tasks. The instructions for each task were printed on 5 X 8 index cards. These instructions gave detailed instructions as to what was to be performed as well as a condensed 3 or 4 step set of instructions. The instructions indicated the filename of the main text file to be edited as well as the filename of the file which contained the required information. The instructions indicated to the subject how much and what kind of information was to be found and where it was to be placed. For example, the instructions might ask the subject to find 4 names which begin with the letter 'T' and place them at the end of the main text file. The subjects were allowed to refer back to the instructions while performing the tasks.

Tutorial Tasks. The five tutorial tasks were designed to allow the subject to become as familiar with the computer system as was possible. The tasks encouraged the subject to use many of the editing and scrolling keys and to 'play around' with the system. The tasks required the the subject to move the cursor around the computer screen, use the scrolling keys and even cause an error to demonstrate that it was permissible to make errors.

Criterion Tasks. The three criterion tasks were similar to the tutorial tasks and were designed to ensure that the subject had no major problems with the condition he or she was assigned to. If the subject was unable to complete two of the three tasks within six minutes for each task, the subject was rejected from the experiment.

Experimental Tasks. The six experimental tasks varied on the dimensions of Memory Load with three levels (High, Medium and Low) and Complexity with two levels (High and Low). These tasks were similar to the tutorial and criterion tasks except that the information which the subject was required to retrieve from the supplementary file was controlled in several ways. The instructions and the text files for the six experimental tasks are found in Appendix D.

In order to control the stimulus words (those words which the subjects were required to retrieve from the supplementary files) for ease of memorization, only words which had a high frequency of occurrence, meaningfulness and concreteness were used. All of the 28 words used for these tasks were found to have a frequency of greater than 50 per million words in the Thorndike and Lorge (1944) list. Twenty-six of the 28 words were empirically rated in terms of meaningfulness and concreteness by Spreen and Schulz (1966). The other two words, 'September' and 'February', were not rated by Spreen and Schulz although they were found to occur more than 100 times per million words (Thorndike & Lorge, 1944).

Meaningfulness was defined by Spreen and Schulz (1966) as the number of associations the word elicited by subjects in 30 seconds. In this study, the 28 words had a mean meaningfulness rating of 8.74 with a standard deviation of .97. Concreteness was defined on a scale of 1 to 7 with 7 referring to most concrete. Concreteness was defined in terms of sense-experience. The words used in this study had a mean concreteness rating of 6.2 with a standard deviation of 1.12.

Three levels of memory load were used. As memory load is a function of one's ability to chunk information, the actual memory load experienced by any two people for the same amount of information will differ. Although there are individual differences in memory load, the memory load levels chosen for this experiment were thought to be different enough so that all of the subjects would experience increasing difficulty as Memory Load was increased.

The Low Memory Load condition was operationally defined as having to locate and work with two words. It was thought that subjects would be able to easily hold two words in working memory. The Medium condition was defined as having to locate and work with four words. Four words were thought to require more effort to hold in memory than two words but yet not require a relatively large amount of effort. Finally, the High Memory Load condition was defined as having to locate and work with 8 words. This level was thought to require a large amount of effort to hold the information in memory.

Two levels of Complexity (High and Low) were used in this experiment. Complexity referred to the location of the information in the supplementary file. Information which was located in close proximity to other information was thought to be easier to locate as well as commit to memory than information which was not located close together. Therefore, Low Complexity was operationally defined as placing words consecutively one after the other in a list. High Complexity was defined as placing non-stimulus words in between the stimulus words in such a way that the stimulus words were not placed in close proximity to one another.

Questionnaires

Mental Workload Questionnaire. The questionnaire which was used in this experiment was the Modified Cooper-Harper (MCH) rating scale modified by Wierwille & Casali, 1983. The MCH is a decision tree which requires subjects to first evaluate the task according to the number and severity of errors and then according to the difficulty level perceived. The level of the tree at which the subject ends is associated with a number called the workload rating. This rating ranges from 1 to 10 with 1 being easy and 10 being impossible.

After completing each task, subjects used the MCH to rate their feeling of workload. An example of this scale and the instructions are shown in Appendix E.

Verbal / Spatial Questionnaire. Separate verbal and spatial tests were given to subjects to measure verbal and spatial aptitudes. The tests which were chosen were found to relate to text editing task performance (Gomez, et al., 1986) as well as computer hierarchical file searching (Vicente, 1987). The spatial test was number VZ-2 from the Kit of Factor-Referenced Cognitive Tests developed by the Educational Testing Service (Ekstrom, French & Harmon, 1976). This paper folding visualization test consisted of two parts with 10 questions and 3 minutes for each part.

The Nelson-Denny Reading Test Form C (1973) was used to measure verbal ability. This test is a basic vocabulary test consisting of 100 questions answered in ten minutes.

Templates. In order to aid the subjects in remembering the function of the PF keys, templates were developed. These templates were made of grey plastic and were cut in such a way as to fit over the whole set of 10 PF keys. Stenciled on the templates were the functions of the PF keys which were used in the experiment. Two types of templates were made, one for the window condition which included the function of PF9 and 10. The second one was for the non-window conditions and was exactly the same as those above with the exception of no definitions for PF9 and 10.

PROCEDURE

The subjects were required to participate in two testing sessions for this experiment. In the first testing session which lasted about 30 minutes, subjects completed the consent form, a short questionnaire and then completed the verbal and spatial tests. In the second testing session which lasted about an hour and a half, subjects went through a tutorial, performed five tutorial tasks, three criterion tasks and six experimental tasks.

Verbal / Spatial Blocking

The purpose of the first testing session was to administer the verbal and spatial tests to develop the blocking factor. In the first testing session, subjects completed a consent form and a short questionnaire referring to their experience with computers. After completing these forms, the experimenter read aloud the instructions for the spatial test while the subjects followed along. If there were any questions, they were answered. The subjects were given three minutes to complete each page of the paper folding test. After completing this test, the experimenter read aloud the instructions for the verbal test and the subjects were told that they had 10 minutes to complete the 100 questions. The tests were always administered in this order.

After 40 of the 48 subjects had completed the tests, the tests were scored and the blocks developed. The spatial test had a mean of 12.89 (out of a possible 20) and a standard deviation of 3.78. The verbal test had a mean of 65.13 (out of a possible 100) and a standard deviation of 17.37.

Subjects were placed into three categories based on their scores on the verbal and spatial tests. Subjects were placed in the first category, Verbal, if their verbal score was 1/2 a standard deviation above the mean verbal score and their spatial score was at or below the mean spatial score. Subjects were placed in the second category, Spatial, if their spatial score was 1/2 a standard deviation above the mean spatial score and their verbal score was at or below the mean verbal score. Subjects were placed in the third category, Equal, if they had equal abilities on both factors. Thus, they were placed in the Equal group if both their verbal and spatial scores were below, at, or above the mean verbal and spatial scores.

This categorization produced 16 Spatial subjects, 16 Verbal subjects, and 16 Equal subjects. The subjects from each of these groups were randomly assigned to the four conditions. The means for each of the blocks within each condition are shown in Table 1.

Insert Table 1 about here

Experimental Session

The subjects were tested in groups of about 5 and were placed between 4 and 10 feet from each other. The tasks were typed on 5 X 8 index cards and placed in front of the computer. The PF key templates were placed over the PF keys.

The subjects first completed a one page questionnaire regarding their motivation for the experiment. They then read a two page instruction sheet explaining the experiment. When all of the subjects had finished reading the instructions, the experimenter described the keyboard and explained that they would then take a four minute typing test. After completing the typing test, the experimenter read the instructions to the Modified Cooper-Harper Rating Scale while the subjects followed along. The subjects were then told to begin the Tutorial and to work at their own pace.

Tutorial. The Tutorial program explained, in more detail, how the text editor worked. The first part of the tutorial pictorially demonstrated the status line and either the window or non-window environments

Table 1A

Mean Verbal and Spatial Scores by Blocks *

Environment

Window

Spatial/Verbal		Verbal	Spatial
Spatial	$\bar{x}$	55.50	15.60
	sd	8.89	2.41
Verbal	$\bar{x}$	72.25	10.05
	sd	8.62	1.61
Equal	$\bar{x}$	64.00	13.70
	sd	20.56	4.64
Marginal Means	$\bar{x}$	63.92	13.12
	sd	14.43	3.74

Notes

Spatial/Verbal		Verbal	Spatial
Spatial	$\bar{x}$	56.75	15.15
	sd	8.06	3.60
Verbal	$\bar{x}$	83.00	11.15
	sd	13.29	2.59
Equal	$\bar{x}$	64.25	13.90
	sd	11.24	3.71
Marginal Means	$\bar{x}$	68.00	13.40
	sd	15.27	3.49

* n = 4 for each block.

Table 1B

Mean Verbal and Spatial Scores by Blocks *

Memorize

Spatial/Verbal		Verbal	Spatial
Spatial	$\bar{x}$	54.75	14.80
	sd	7.50	1.52
Verbal	$\bar{x}$	79.50	12.00
	sd	9.61	4.00
Equal	$\bar{x}$	65.00	10.10
	sd	22.05	4.16
Marginal Means	$\bar{x}$	66.42	12.30
	sd	16.90	3.89

Switch

Spatial/Verbal		Verbal	Spatial
Spatial	$\bar{x}$	57.50	14.90
	sd	12.15	1.27
Verbal	$\bar{x}$	80.25	9.15
	sd	17.17	1.33
Equal	$\bar{x}$	58.75	10.10
	sd	16.40	4.16
Marginal Means	$\bar{x}$	65.50	11.38
	sd	17.69	3.54

* n = 4 for each block.

depending on the condition in which the subject was placed. The Tutorial also discussed the location and use of the keys which operated the environments. The instructions differed slightly for each of the conditions in order to emphasize the specific strategies. The subjects completed this part of the Tutorial at their own pace, usually taking around 30 minutes.

After completing the initial Tutorial demonstrations, the subject began the five tutorial tasks. The subjects were instructed to read the instructions on the index card before beginning the tasks. In this way, reading time would not be included in the task time. Upon completing each task, the subjects were instructed to rate the task for workload using the Modified Copper-Harper Rating Scale.

The five Tutorial tasks were performed at the subject's own pace. During the time the subjects were performing the Tutorial tasks, the experimenter and an assistant answered individual subject's questions and gave the subjects any help which they may have needed. Subjects completed this part within 30 minutes.

After completing the Tutorial tasks, the subjects began the three Criterion tasks. These tasks were performed at the subjects own pace and only limited help was given to the subjects.

After completing the three criterion tasks, the subjects began the six experimental tasks. During these tasks the subjects were given no help. Again, the subjects worked at their own pace in completing the tasks. When the subject had completed the tasks, the computer displayed summary information for the subject. This information included the subject's words per minute and percentage of words typed correctly, average task time and the average criterion time.

Task Order. The experimental tasks were ordered in a Latin squares fashion to help counteract order effects. This arrangement, based on Wagenaar's (1969) method, was digram balanced, allowing each task to be preceded and followed by all other tasks only once. The three Memory Load levels (High, Medium and Low) and the two Complexity levels (High, and Low) yielded six unique combinations of tasks. The 6 by 6 matrix dictating the order of presentation of the tasks for the subjects can be found in Wagenaar (1969). Prior to the beginning of the experiment, the index cards were placed in a specific order, according to the Latin squares matrix, for each individual subject.

RESULTS

SUBJECT CHARACTERISTICS

The 19 males and 29 females in this study had a mean age of 20.35 years with a range of 18 to 25. Four of the subjects were freshmen, 16 were sophomores, 12 were juniors, and 16 were seniors. All but two of the subjects had used an IBM PC before. The other two subjects had used Apple computers. Twenty-three of the subjects had not used a program that had windows before, whereas 25 of the subjects had used a program that contained windows.

Nineteen of the subjects reported using only one computer program, 15 had used two computer programs, 10 had used three computer programs and four had used more than three programs. Twenty-eight of the subjects were liberal arts majors, 5 were business majors, 5 were engineering majors, and 9 were science majors. One subject reported using a PC less than one hour a month, 11 reported from 1 to 5 hours a month, 15 reported from 6 to 10 hours a month, 4 reported from 11 to 15 hours a month, 2 from 16 to 20 hours a month and 10 people reported using a PC more than 20 hours a month. Table 2 presents the frequency distribution of these subject characteristics broken down by environment. Subjects' responses to the motivation questionnaire given at the time of the experiment are presented in Table 3.

Insert Table 2 about here

Table 2

Frequencies of Subject Characteristics

	Environments			
	Window	Notes	Switch	Memory
Sex				
Male	4	3	6	6
Female	8	9	6	6
Major				
Business	2	0	1	2
Liberal Arts	8	8	8	4
Science	2	2	2	3
Engineering	0	2	0	3
Year				
Freshman	0	1	1	2
Sophomore	2	5	5	4
Junior	7	2	2	1
Senior	3	4	4	5
Used Windows Before				
Yes	6	7	5	5
No	6	5	7	7
# of Programs Used				
Zero	0	1	0	0
One	5	5	2	6
Two	2	2	6	5
Three	3	3	3	1
Four or more	2	1	1	0
# Hours on Computer				
1 hr./mth.	3	2	1	0
1-5 hrs./mth.	1	4	2	4
6-10 hrs./mth.	4	4	4	3
11-15 hrs./mth.	2	1	0	1
16-20 hrs./mth.	1	0	1	0
> 20 hrs./mth.	1	1	4	4

Table 3A

Means for Text Editor Questionnaire by Environments *

1) How do you feel about using the IBM PC right now?

1	2	3	4	5
Anxious	A little uneasy	Neutral	Comfortable	Very Comfortable

Window	Notes	Memorize	Switch
$\bar{x}$ = 3.41	$\bar{x}$ = 3.33	$\bar{x}$ = 3.08	$\bar{x}$ = 3.54
sd = 1.24	sd = 1.30	sd = 1.00	sd = 1.13

2) How do you feel about this experiment right now?

1	2	3	4	5
Anxious	A little uneasy	Neutral	Comfortable	Very Comfortable

Window	Notes	Memorize	Switch
$\bar{x}$ = 3.58	$\bar{x}$ = 3.08	$\bar{x}$ = 3.17	$\bar{x}$ = 3.00
sd = 1.00	sd = .67	sd = .94	sd = 1.09

3) Rate your own note taking ability?

1	2	3	4	5
Below Average		Average		Above Average

Window	Notes	Memorize	Switch
$\bar{x}$ = 3.50	$\bar{x}$ = 3.50	$\bar{x}$ = 3.50	$\bar{x}$ = 3.00
sd = .67	sd = .67	sd = .67	sd = .63

* n = 12 for each group.

Table 3B

Means for Text Editor Questionnaire by Environments (cont.)

4) How much time, in comparison to others, do you think it would take you to learn a simple word processing program?

1----------2----------3----------4----------5		
Longer Than Average	Average	Less Than Average

Window	Notes	Memorize	Switch
$\bar{x}$ = 3.17	$\bar{x}$ = 3.25	$\bar{x}$ = 3.42	$\bar{x}$ = 3.45
sd = 1.03	sd = .96	sd = .67	sd = 1.13

5) Rate yourself as to your ability to understand and use new computer packages:

1----------2----------3----------4----------5		
Poor	Average	Excellent

Window	Notes	Memorize	Switch
$\bar{x}$ = 2.83	$\bar{x}$ = 2.83	$\bar{x}$ = 3.00	$\bar{x}$ = 3.45
sd = .94	sd = 1.03	sd = .74	sd = 1.04

6) What is your motivation to participate in this experiment?

1----------2----------3----------4----------5		
Unmotivated	Neutral	Highly Motivated

Window	Notes	Memorize	Switch
$\bar{x}$ = 3.50	$\bar{x}$ = 3.58	$\bar{x}$ = 3.33	$\bar{x}$ = 3.54
sd = .67	sd = .79	sd = .65	sd = .69

* n = 12 for each group.

Insert Table 3 about here

MULTIVARIATE ANALYSIS

Before conducting the univariate analyses of variance (ANOVA) it is important to assess the effects of performing analyses on several dependent measures which are thought to be correlated. These dependent measures are task completion time, used for Hypotheses 1 and 2, the total number of responses, used for Hypothesis 3, and the number of scrolling keystrokes, used for Hypothesis 4. Multiple analysis of variance (MANOVA) can control for possible correlations and biasing among all dependent measures while testing all variables simultaneously (Hair, Anderson, Tatham & Grablowsky, 1979). Wilk's U (likelihood ratio) is the statistic preferred by many (Hair, et al., 1979) to test the significance of the MANOVA. The smaller the value of this statistic, the greater the implied significance.

A 4 X 3 X (3 X 2) MANOVA (Environment X verbal/spatial Blocking X Memory Load X Complexity) was performed using task completion time, total number of keystrokes and the number of scrolling keystrokes as dependent measures. The results, summarized in Table 4, indicate that there are significant main effects for Environment, U(9,83)=.44, p=.001, for Complexity, U(3,34)=.21, p=.000, and for Memory Load, U(6,31)=.09 p=.000. A significant interaction was found between Environment and Memory Load, U(18,88)=.13, p=.000. However, a significant interaction was not found between Complexity and Environment U(9,83)=.75, p=.330.

Table 4

Multiple Analysis of Variance on Task Completion Times

	Source	dv	DFh	DFe	U	P
Between Subjects	Env.	3	9	83	0.44	.001
	Blocking	3	6	68	0.74	.103
	Env. x Block.	3	18	97	0.49	.107
	S/EB	(Error Term)				
Within Subjects	Complex	3	3	34	0.21	.000
	Env.x Complex	3	9	83	0.75	.330
	Block. x Comp.	3	6	68	0.88	.641
	Env. x Block.x Complex	3	18	97	0.71	.801
	C X S/EB	(Error Term)				
	Memload	3	6	31	0.09	.000
	Env. x Mem.	3	18	88	0.13	.000
	Block. x Mem.	3	12	62	0.55	.067
	Env. x Block.x Memload	3	36	139	0.49	.918
	M X S/EB	(Error Term)				
	Comp.x Mem.	3	6	31	0.23	.000
	Env.x Comp.x Memload	3	18	88	0.67	.754
	Block. x Comp.x Memload	3	12	62	0.63	.217
	Env. x Block.x Complex x Mem.	3	36	139	0.28	.155
	C X M X S/EB	(Error Term)				

Where: dv = number of dependent measures
DFh = degrees of freedom for treatment effect
DFe = degrees of freedom for error effect
U = Wilk's likelihood ratio statistic

Insert Table 4 about here

The results indicated that univariate ANOVA's could be carried out to test for Hypotheses 1 and 3. Because there was no significant interaction between Complexity and Environment for the three dependent measures tested, Hypotheses 2 and 4 were not supported. The analyses for the individual hypotheses are presented below.

ANALYSES FOR HYPOTHESIS 1

The first hypothesis suggested that there would be a significant interaction between Memory Load and Environment with task completion time as the dependent measure. It was expected that as Memory Load increased, task completion time would be significantly longer in the Memorization condition than in the Window condition. The means and standard deviations for the task completion times of the six tasks are presented in Table 5.

Insert Table 5 about here

To test this hypothesis, a mixed, fixed effects 4 X 3 X (3 X 2) (Environment X verbal/spatial Blocking X Memory Load X Complexity) analysis of variance (ANOVA) was performed using the task completion time scores for each of the six tasks as the dependent measure. This ANOVA included the spatial/verbal blocking factor as it was thought that this factor might be able to account for some of the within group variance and reduce the error term for the ANOVA.

Table 5

Mean Task Completion Times (in seconds) *

		Memory Load			
Environment		High	Medium	Low	Marginal Means
Window	$\bar{x}$	113.63	84.91	91.65	96.73
	sd	23.31	15.98	25.48	15.40
Notes	$\bar{x}$	121.82	83.03	73.96	92.93
	sd	37.97	20.07	17.51	21.64
Switch	$\bar{x}$	133.68	92.23	74.58	100.17
	sd	31.47	30.93	25.20	24.15
Memorize	$\bar{x}$	142.96	94.38	82.28	106.54
	sd	23.16	54.34	17.71	26.43
Marginal Means	$\bar{x}$	128.02	88.64	80.62	99.09
	sd	30.82	33.05	22.32	22.15

		Complexity		
Environment		High	Low	Marginal Means
Window	$\bar{x}$	113.59	79.87	96.73
	sd	23.86	11.55	15.40
Notes	$\bar{x}$	103.42	82.45	92.93
	sd	26.17	18.70	21.17
Switch	$\bar{x}$	109.68	90.65	100.17
	sd	28.81	24.28	24.15
Memorize	$\bar{x}$	116.91	96.16	106.54
	sd	26.06	28.12	26.43
Marginal Means	$\bar{x}$	110.90	87.28	99.09
	sd	25.93	21.89	22.15

* n = 12 for each group.

In order to conduct the mixed ANOVA, two assumptions of homogeneity of variance must be met. The first assumption is for the between subjects factor of a mixed design and can be tested with Box's M test which can be based on an F statistic. Box's M tests that the variance-covariance matrix is equal across all levels of the between subjects factor. This test is very sensitive to departures from normality and Keppel (1982) indicates that the ANOVA is robust to even severe violations of this assumption. However, it will be reported as an indication of variance differences between the different environments. The second assumption of homogeneity of variance is that the variances for an effect are equal and the covariances are 0. Mauchly's test of sphericity is one method for testing this assumption and is based on a chi-square distribution. It is used when more than two groups are involved and thus will only be used for effects involving the Memory Load factor. If the assumption is violated the Greenhouse-Geisser epsilon correction to both the numerator and denominator degrees of freedom may be performed (Norusis, 1986). This correction is used to evaluate the F value and not to create a new one.

Mauchly's tests of homogeneity of variance were not significant for Memory Load, X^2 (2)=2.35,p=.31, or for the Memory Load by Complexity interaction, X^2 (2)=3.76,p=.152. Box's M test could not be estimated with the addition of the blocking factor. The hypothesized interaction between Memory Load and Environment, F(6,72)=2.02, p=.074 was not significant. However, significant main effects for Memory Load, F(2,72)=62.72, p=.000 and for Complexity were found, F(1,36)=73.77, p=.000. There was no significant main effect for Environment F(3,36)=.85, p=.475 or for Blocking F(2,36)=2.06,p =.143. The complete ANOVA table is presented in Table 6.

Table 6

Analysis of Variance on Task Completion Times with Blocking Factor

	Source	SS	DF	MS	F	P
Between Subjects	Env.	7207.18	3	2402.39	.85	.475
	Blocking	11605.10	2	5802.55	2.06	.143
	Env. x Block.	17969.11	6	2994.85	1.06	.404
	Within Cells	101604.85	36	2822.36		
Within Subjects	Complex	40161.09	1	40161.09	73.77	.000
	Env.x Complex	2490.46	3	830.15	1.52	.225
	Block. x Comp.	664.60	2	332.30	.61	.549
	Env. x Block.x Complex	1275.97	6	212.66	.39	.880
	Within Cells	19598.68	36	544.41		
	Memload	123614.08	2	61807.04	62.72	.000
	Env. x Mem.	11922.84	6	1987.14	2.02	.074
	Block. x Mem.	7454.07	4	1863.52	1.89	.121
	Env. x Block.x Memload	10079.27	12	839.94	.85	.598
	Within Cells	70954.79	72	985.48		
	Comp.x Mem.	1759.92	2	879.96	1.43	.245
	Env.x Comp.x Memload	1189.72	6	198.29	.32	.923
	Block. x Comp.x Memload	2082.27	4	520.57	.85	.499
	Env. x Block.x Complex x Mem.	9538.44	12	794.87	1.30	.240
	Within Cells	44168.90	72	613.46		
Total		467372.23	287	1628.47		

Insert Table 6 about here

The results from this ANOVA were disappointing. However, the results are not surprising given the low correlations between the six task times and the verbal and spatial scores. The correlations for the verbal scores with the six task times ranged from -.07 to -.25 with none significantly different from 0. The correlations for the spatial scores with the six task times ranged from -.06 to -.299 with only two correlations significantly different from 0. The blocks created from these variables had correlations ranging from .10 to .28 with no significant results. All of the correlations are presented in Table 7.

Insert Table 7 about here

According to Keppel (1982), when the blocking factor is uncorrelated with the dependent measure, the error term is based on scores which are as heterogeneous as those in a completely randomized experiment. The main problem with this is the loss of degrees of freedom in the error term caused by adding an uncorrelated blocking factor into the analysis. This can result in even lower F ratios than would have been found without using the Blocking factor.

Pooled Error Term Task Time Analysis

It was stated above that when a blocking factor does not correlate with a dependent measure, F ratios may actually decrease due to a loss of degrees of freedom in the error term. By collapsing down on the blocking factor (i.e. not including it in the analysis) one can use a

Table 7

Correlations Between Verbal and Spatial Tests and Task Completion Times *

Task	Spatial	Verbal	Blocks

1) High Memory Load,
High Complexity
	r = -.068
	p = .645
		-.252
		.085
			-.296
			.041

2) Medium Memory Load,
High Complexity
	r = -.299
	p = .039
		-.211
		.150
			.221
			.132

3) Low Memory Load,
High Complexity
	r = -.081
	p = .583
		-.197
		.179
			-.146
			.321

4) High Memory Load,
Low Complexity
	r = -.098
	p = .507
		-.072
		.626
			.156
			.288

5) Medium Memory Load,
Low Complexity
	r = -.287
	p = .048
		-.139
		.345
			.213
			.146

6) Low Memory Load,
Low Complexity
	r = -.174
	p = .238
		-.237
		.105
			.103
			.487

* n = 48 for each correlation.

pooled error term which may allow a more accurate assessment of effects. Therefore, a mixed fixed effects 4 X (3 X 2) (Environment X Memory Load X Complexity) ANOVA was performed without the blocking factor using the task completion time scores for each of the six tasks as the dependent measure.

For this analysis, Box's M was significant, M=141.36, F(63,4536) =1.672, p=.001. Mauchly's test was not significant for the Memory Load effect, X^2 (2)=1.679, p=.432, or for the Complex by Memory Load interaction, X^2 (2)=5.63, p=.06.

The hypothesized interaction, Memory Load by Environment was not significant for this ANOVA. However, again, main effects were significant for Complexity, F(1,44)=82.04, p=.000, and for Memory Load F(2,88)=61.47, p=.000. Finally, the main effect for Environment was not significant, F(3,44)=.81, p=.497. F(6,88)=1.98, p=.078. The complete ANOVA table is presented in Table 8.

Insert Table 8 about here

These results were again disappointing. The use of a pooled error term was not able to elucidate the hypothesized interaction. However, as the subjects completed the tasks in a balanced Latin squares fashion, it was thought that a Latin squares analysis, which takes into account practice effects, might accomplish the desired results.

Latin Squares Task Time Analysis

In the present experiment, tasks were ordered in such a fashion that subjects were given tasks in a balanced Latin squares order. The purpose

Table 8

Analysis of Variance on Task Completion Times

	Source	SS	DF	MS	F	P
Between Subjects	Env.	7207.18	3	2402.39	.81	.497
	S/Env.	131179.07	44	2981.34		
Within Subjects	Complex	40161.09	1	40161.09	82.04	.000
	Env.x Complex	2490.46	3	830.15	1.70	.182
	Comp.x S/Env.	21539.25	44	489.53		
	Memload	123614.08	2	61807.04	61.47	.000
	Env. x Mem.	11922.84	6	1987.14	1.98	.078
	Mem.x S/Env.	88488.13	88	1005.55		
	Comp.x Mem.	1759.92	2	879.96	1.39	.255
	Env.x Comp.x Memload	1189.72	6	198.29	.31	.929
	Mem.x Comp.x S/Env.	55789.60	88	633.97		
Total		485341.34	287	1691.08		

of this was two fold. First, if the tasks are given to subjects in the same order, differences could be due solely to a learning effect. Thus, by having the subjects perform the tasks in a Latin squares order, in which each task occurs before and after every other task only once, the possibility of differences occurring because of a learning effect are reduced by distributing the learning more or less evenly across all of the tasks. The second purpose is to be able to statistically account for the effects of practice. The analyses which have been conducted up to this point have not accounted for practice. In not accounting for possible practice effects, the error terms used in the ANOVA's will be artificially inflated. Therefore, a Latin squares analysis determines the amount of variance attributed to the practice effect and any interactions between practice and the other effects, and removes this variance from the error terms.

Keppel (1982) discusses two different methods for conducting a Latin squares analysis. The first, subtracting sources of variance, provides an actual estimate of the amount of variance accounted for by each practice effect and the interactions. The second method, allows the removal of practice effects from the individual scores directly before an ANOVA is performed and does not offer estimates of the amount of variance accounted for by each of the Practice effects. Typically it is preferable to remove the effects of practice directly when separate error terms are required to test terms in an ANOVA. In the present experiment this is the case. Therefore, the effects of practice were removed from the individual scores as outlined by Keppel (1982). It should be noted that although each score is adjusted, the actual task time means for each group will not change. This is because practice effects are adjusted by either adding or subtracting amounts based on testing position with the sum of the amounts equaling zero. Therefore, individual scores may go up or down but the average is still maintained.

The assumption of homogeneity of variance using Box's M test was significant, M=143.6, F(63,4536)=1.69,p=.001. The assumptions of homogeneity of variance using Mauchly's test were not significant for Memory Load, X^2 (2)=.700, p=.7, or for the Memory Load by Complexity interaction, X^2 (2)=3.58, p=.166. In this Latin squares analysis the between effect will not change, thus, the Environment main effect was not significant, F(3,44)=.81, p=.495. Again, the hypothesized interaction between Memory Load and Environment, F(6,80)=2.07, p=.069 was not significant. However, the main effects for Complexity, F(1,40)=67.75, p=.000, and Memory Load, F(2,80)=64.30, p=.00, were significant. It should be noted here that the degrees of freedom for the error term is adjusted downward because practice effects must be accounted for. The complete ANOVA table is presented in Table 9.

Insert Table 9 about here

The above analyses indicate that there is no significant interaction between Memory Load and Environment for task completion times. However, significant main effects for Memory Load and Complexity were found. As a main effect for Memory Load was found, two Bonferroni contrasts were performed to determine where the differences were. The first contrast was between the Low and Medium levels (80.62 sec. versus 88.63 sec.) and was not significant, t(47)=1.71, p=.095. The second contrast was between

Table 9

Latin Squares Analysis of Variance on Task Completion Times

	Source	SS	DF *	MS	F	P
Between Subjects	Env.	7239.41	3	2413.14	.81	.495
	S/Env.	131179.07	44	2981.34		
Within Subjects	Complex	40161.09	1	40161.09	67.76	.000
	Env.x Complex	2490.46	3	830.15	1.40	.230
	Comp.x S/Env.	23709.22	40	592.73		
	Memload	123614.08	2	61807.04	64.29	.000
	Env. x Mem.	11922.84	6	1987.14	2.07	.060
	Mem.x S/Env.	76916.74	80	961.45		
	Comp.x Mem.	1759.92	2	879.96	1.54	.200
	Env.x Comp.x Memload	1189.72	6	198.29	.34	.900
	Mem.x Comp.x S/Env.	45840.32	80	573.00		
Total		466022.87	267	1745.40		

* Error degrees of freedom are adjusted to account for practice effects.

the Medium and High levels (88.63 sec. versus 128.02 sec.) and was significant $t(47)=7.52$, $p=.000$. Therefore, the subjects did not take significantly longer to complete the Medium tasks as compared to the Low tasks, but they did take significantly longer to complete the High Memory Load tasks as compared to the other two levels of Memory Load. This is graphed in Figure 1. The main effect for Complexity indicates that subjects took significantly longer to complete tasks which were High in Complexity than those which were Low in Complexity (110.9 sec. versus 87.28 sec.). These results are graphed in Figure 2.

Insert Figure 1 about here

Insert Figure 2 about here

The difficulty in finding significant results for this dependent measure can be attributed to many factors. These include large individual differences in understanding and using the word processor as well as understanding the given task, using memory load levels which are not different enough from each other to produce environment differences, as well as the fact that task completion time is composed of many factors such as the number of responses executed, cognitive processing time and rehearsal time. More will be said about these factors in the discussion. However, one should realize that true differences might exist, but are masked by other factors.

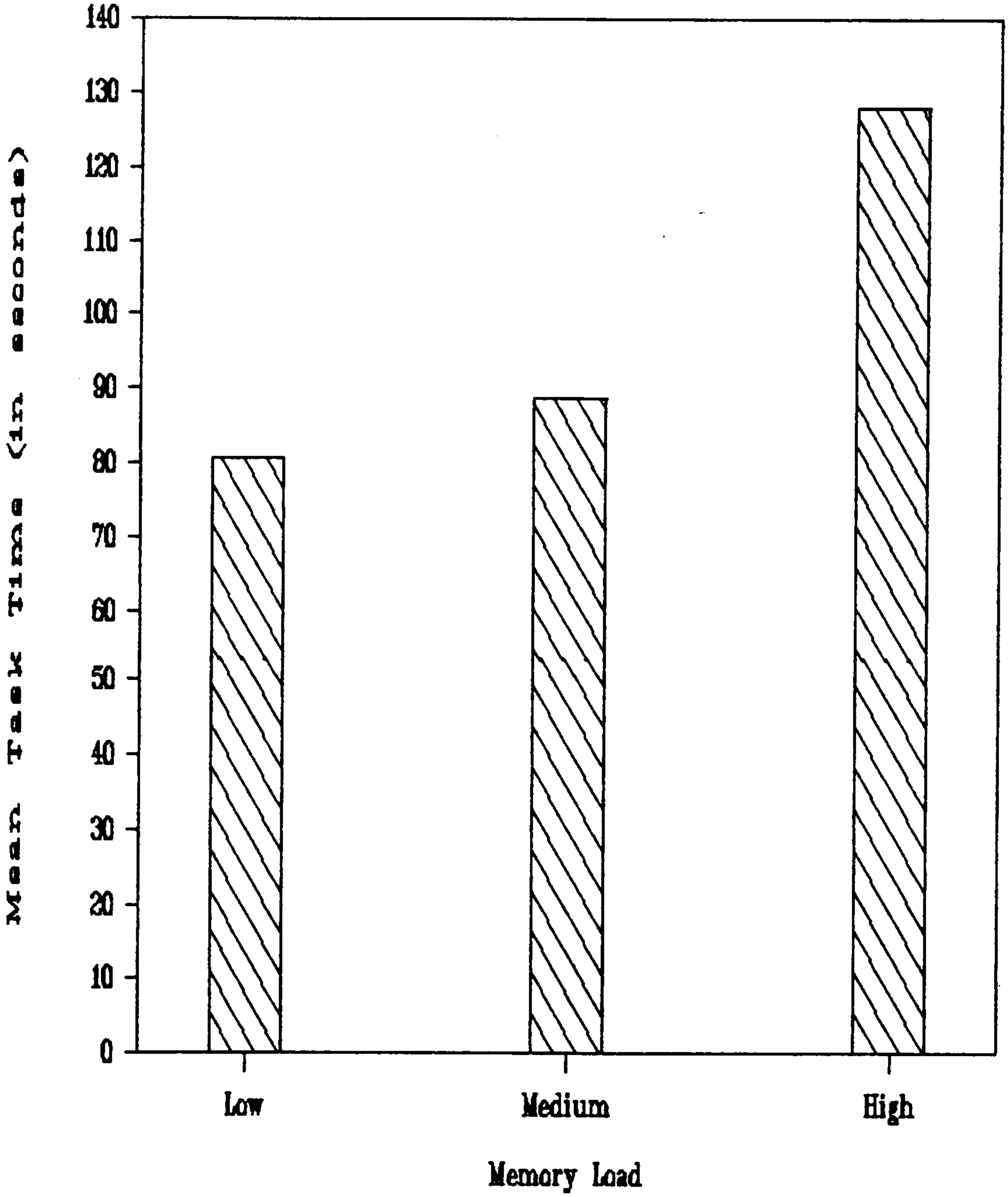


Figure 1. Memory Load Main Effects for

Task Completion Times.

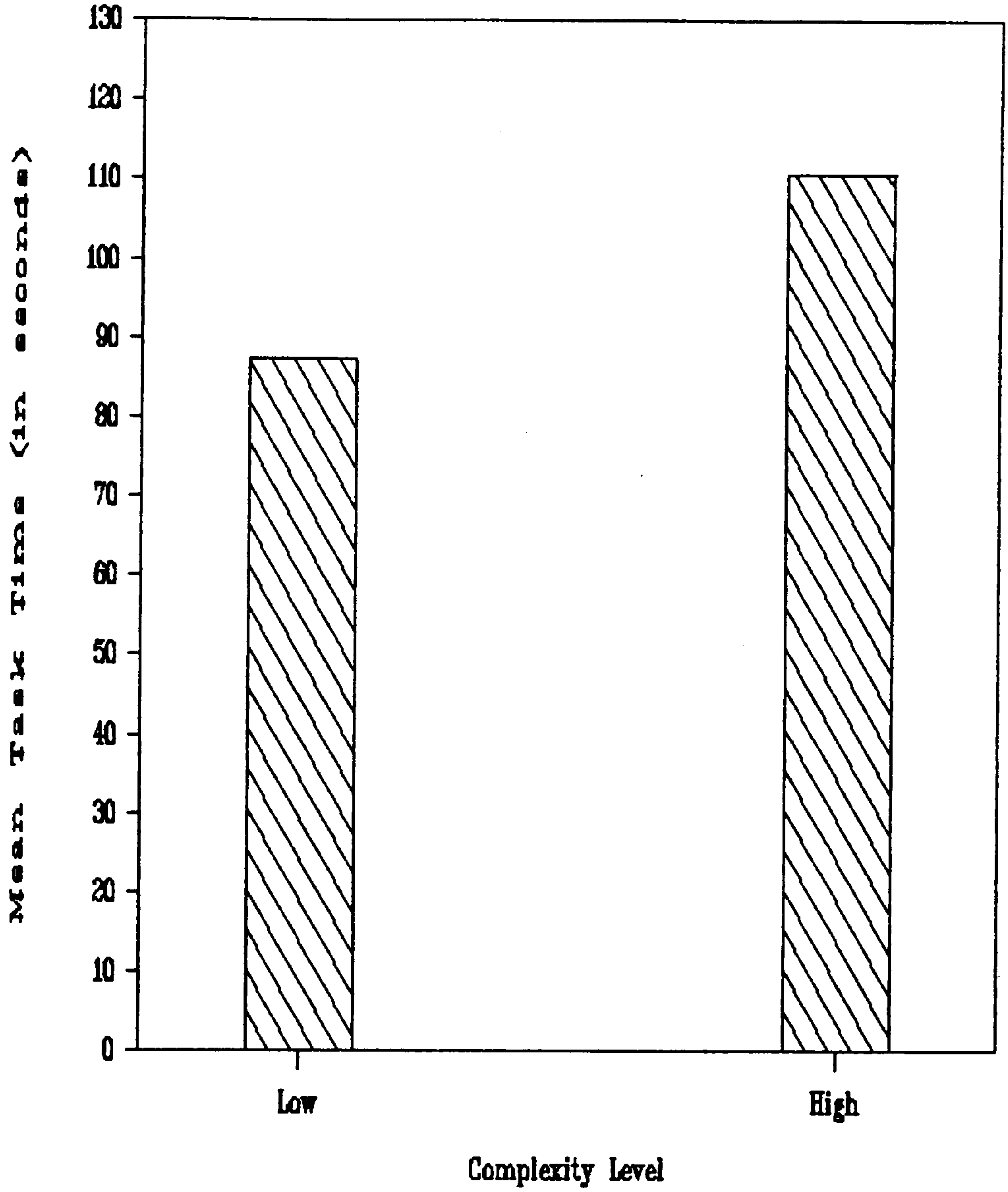


Figure 2. Complexity Main Effects for Task Completion Times.

ANALYSES FOR HYPOTHESIS 3

It was expected that there would be a significant interaction between Memory Load and Environment when the total number of responses was used as a dependent measure. It was hypothesized that as Memory Load increased, more responses would be executed in the Notes and Switch conditions than in the Window and Memorization conditions. In all of the environments, except the Notes condition, the total number of responses was defined in terms of the total number keystrokes executed. In the Notes condition, however, the number of responses made in writing notes had to be accounted for. Therefore, a count was made of the number of characters written on the note paper for each of the subjects for each of the tasks in the Notes condition. According to estimates in Seibel (1972) typing occurs about 0.625 times slower than writing. Therefore, the number of handwritten characters were multiplied by .625 and then added to the number of keystrokes executed in the Notes condition. This number was treated as the total number of responses executed. The means and standard deviations for the total number of responses executed are presented in Table 10.

Insert Table 10 about here

For this ANOVA, Box's M test of homogeneity of variance was found to be significant, M=150.03, F(63,4536)=1.77, *p*=.000. Also, Mauchly's test was significant for Memory Load, X^2 (2)=8.03, *p*=.018, but not for the Memory Load by Complexity interaction, X^2 (2)=2.41, *p*=.299.

Table 10

Mean Number of Total Responses Executed *

		Memory Load			
Environment		High	Medium	Low	Marginal Means
Window	$\bar{x}$	115.21	84.71	96.04	98.32
	sd	20.17	17.60	25.67	19.56
Notes	$\bar{x}$	124.46	93.59	87.20	101.75
	sd	23.24	18.24	12.62	13.74
Switch	$\bar{x}$	140.21	96.62	86.58	107.80
	sd	22.17	22.54	20.49	11.71
Memorize	$\bar{x}$	101.62	87.46	83.08	90.72
	sd	10.01	9.81	9.62	6.04
Marginal Means	$\bar{x}$	120.12	90.60	88.23	99.65
	sd	23.75	17.74	18.30	14.60

		Complexity		
Environment		High	Low	Marginal Means
Window	$\bar{x}$	107.61	89.03	98.32
	sd	24.16	16.59	19.56
Notes	$\bar{x}$	104.64	98.86	101.75
	sd	19.01	10.98	13.74
Switch	$\bar{x}$	110.86	104.75	107.81
	sd	13.91	16.53	11.71
Memorize	$\bar{x}$	94.03	87.42	90.72
	sd	8.91	5.39	6.04
Marginal Means	$\bar{x}$	104.29	95.01	99.65
	sd	18.05	14.67	14.60

* n = 12 for each group.

Therefore, the Geisser-Greenhouse correction was applied to the Memory Load effects.

The hypothesized interaction between Memory Load and Environment was significant, F(5.1,74.8)=4.45, p=.001 for this 4 X (3 X 2) mixed ANOVA on the total number of keystrokes. A significant Environment main effect, F(3,44)=3.27,p=.03 was found. A significant Complexity main effect was also found, F(1,44)=19.35, p=.000, as was a significant Memory Load main effect (with adjusted df), F(1.7,74.8)=64.48, p=.000. The main effects for Complexity can be interpreted because the interaction between Complexity and Environment was not significant, F(3,44)=2.17, p=.105. Therefore, more keystrokes were executed in the High Complexity level than in the Low Complexity level (104.29 versus 95.01). The complete ANOVA table is presented in Table 11.

Insert Table 11 about here

Simple Effects Analysis

The significant interaction between Memory Load and Environment using the total number of responses as the dependent measure suggested that responses were executed differentially in the environments as Memory Load increased. Therefore, an analysis of simple effects was warranted to determine exactly where the interaction was located. For this analysis, separate error terms for the different effects were used in accordance with Keppel (1982). This method tends to be more conservative than by using a single pooled error term (Keppel, 1982). A significant effect was found at the High Memory Load level, F(3,44)=8.312, p=.000. No

Table 11

Analysis of Variance on Total Number of Responses Executed

	Source	SS	DF	MS	F	P
Between Subjects	Env.	10973.23	3	3657.74	3.27	.030
	S/Env.	49170.96	44	1117.52		
Within Subjects	Complex	6191.76	1	6191.76	19.35	.000
	Env.x Complex	2086.37	3	695.46	2.17	.105
	Comp.x S/Env.	14075.87	44	319.91		
	Memload	60639.29	2	30319.65	64.48	.000
	Env. x Mem.	12558.68	6	2093.11	4.45	.001
	Mem.x S/Env.	41378.72	88	470.21		
	Comp.x Mem.	19063.31	2	9531.66	25.94	.000
	Env.x Comp.x Memload	2716.76	6	452.79	1.23	.298
	Comp.x Mem.x S/Env.	32333.21	88	367.42		
Total		251188.16	287	875.22		

significant effect was found at the Medium Memory Load level, F(3,44)=1.15, $\underline{p}$=.33, or at the Low Memory Load level, F(3,44)=1.09, $\underline{p}$=.36.

This simple effects analysis revealed that there were no differences in the number of responses executed in the Low and Medium Memory Load conditions. However, in the High Memory Load condition, more responses were executed in at least one environment. A Duncan's multimean test was conducted to determine where the difference for the High Memory Load level was located. The results of this test indicated that the fewest responses were executed in the Memorization (101.62) and Window (114.20) conditions as compared to the Switch condition (140.28). The summary tables for these effects are presented in Table 12. A graph of the interaction between Memory Load and Environment is presented in Figure 3.

Insert Table 12 about here

Insert Figure 3 about here

The finding that more keystrokes are executed in the Switch condition than in the Window and Memorization conditions was expected. In a high memory load situation, subjects in the Switch condition will tend to access the supplementary file several times. Each time this occurs, the subjects must execute keystrokes to open a new file as well as to type the name of the file when the computer asks for it. Many different types of keystrokes are executed in completing a task, however, and an examination of these keystrokes should indicate the problems or benefits of various strategies. For instance, one would think that more alphabetic keystrokes were executed in the Switch condition in the High Memory Load condition since the subject may access the supplementary file several

Table 12

Summary of Simple Effects F Tests on
Total Number of Responses Executed

Memory Load - High

Source	SS	DF	MS	F	P
Environment	9592.50	3	3197.50	8.31	.0002
S/Environment	16924.75	44	384.65		

Results of Duncan's Multiple Range Analysis *

Environment	Memory	Window	Notes	Switch
Mean Value	101.62	114.21	124.46	140.21
	————	————	————	————
		————	————	

Memory Load - Medium

Source	SS	DF	MS	F	P
Environment	1078.14	3	359.38	1.15	.338
S/Environment	13711.99	44	311.64		

Memory Load - Low

Source	SS	DF	MS	F	P
Environment	1095.31	3	365.10	1.097	.360
S/Environment	14638.09	44	332.68		

* Means with a common line do not differ significantly at $p<.05$.

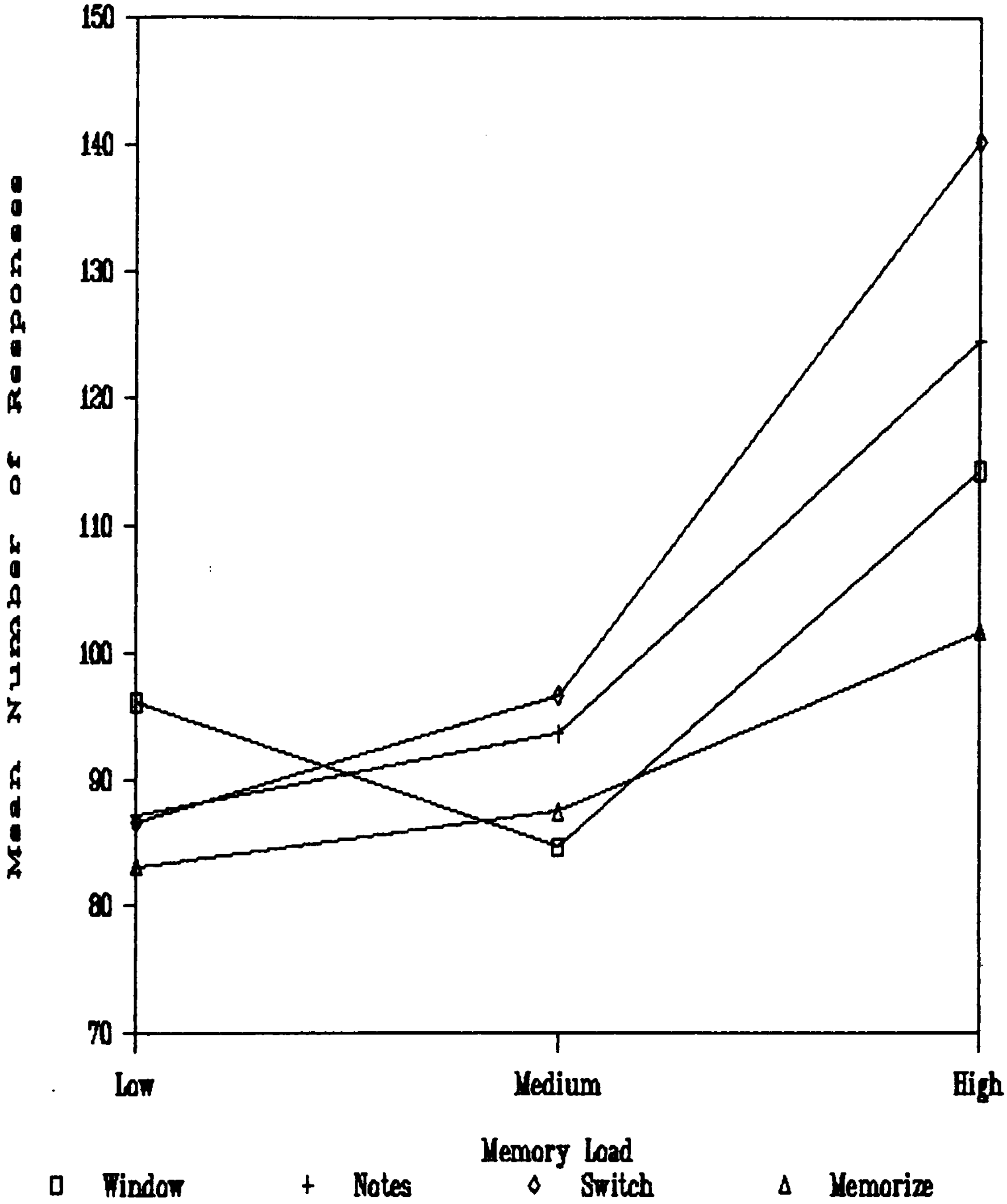


Figure 3. Environment by Memory Load Interaction for the Mean Number of Total Responses.

times and thus, will type the name of the file several times. Also, it would be expected that more scrolling keystrokes were executed in the Window condition since only part of the file can be seen by the subject at any one time. In order to test these assumptions, an exploratory analysis was conducted on the number of alphabetic keystrokes executed.

Alphabetic Keystroke Analysis

An exploratory 4 X (2 X 3) fixed effects mixed ANOVA was conducted using the number of alphabetic keystrokes as the dependent measure. The means and standard deviations for the number of alphabetic keys pressed are presented in Table 13.

Insert Table 13 about here

Box's M test for homogeneity of variance was significant, M=243.48, F(63,4536)=2.88, *p*=.000. Mauchly's test for the Memory Load effect was not significant, X^2 (2)=2.24, *p*=.326. However, Mauchly's test for the Memory Load by Complexity interaction was significant, X^2 (2)=6.33, *p*=.042.

The ANOVA on the number of alphabetic keys pressed revealed a significant interaction between Memory Load and Environment, F(6,88)=4.43, *p*=.001. The Complexity by Environment interaction was not significant, F(3,44)=1.75, *p*=.17. There was, however, a significant Environment main effect, F(3,44)=11.63, *p*=.000, and a significant Memory Load main effect, F(2,88)=77.64, *p* =.000. No significant Complexity main effect was found, F(1,44)=.57, *p*=.455. The complete ANOVA table is presented in Table 14.

Table 13

Mean Number of Alphabetic Keystrokes Executed *

Memory Load

Environment		High	Medium	Low	Marginal Means
Window	$\bar{x}$	53.79	44.92	44.25	47.65
	sd	6.14	3.79	5.77	3.58
Notes	$\bar{x}$	54.25	45.00	40.46	46.57
	sd	6.69	2.76	1.19	3.06
Switch	$\bar{x}$	66.33	51.33	48.37	55.35
	sd	7.78	6.28	11.33	6.75
Memorize	$\bar{x}$	49.21	45.58	43.50	46.10
	sd	4.19	4.51	4.02	3.14
Marginal Means	$\bar{x}$	55.89	46.71	44.14	48.92
	sd	8.87	5.14	7.08	5.71

Complexity

Environment		High	Low	Marginal Means
Window	$\bar{x}$	48.61	46.69	47.65
	sd	5.18	2.89	3.59
Notes	$\bar{x}$	45.61	47.53	46.57
	sd	4.18	2.73	3.06
Switch	$\bar{x}$	54.83	55.86	55.35
	sd	7.84	7.09	6.75
Memorize	$\bar{x}$	45.67	46.53	46.10
	sd	3.27	3.20	3.14
Marginal Means	$\bar{x}$	48.68	49.15	48.92
	sd	6.45	5.77	5.71

* n = 12 for each group.

Table 14

Analysis of Variance on the Number of Alphabetic Keystrokes Executed

	Source	SS	DF	MS	F	P
Between Subjects	Env.	4061.39	3	1353.80	11.63	.000
	S/Env.	5121.28	44	116.39		
Within Subjects	Complex	16.06	1	16.06	.57	.455
	Env.x Complex	148.56	3	49.52	1.75	.171
	Comp.x S/Env.	1244.72	44	28.29		
	Memload	7329.25	2	3664.63	77.64	.000
	Env. x Mem.	1254.53	6	209.09	4.43	.001
	Mem.x S/Env.	4153.56	88	47.20		
	Comp.x Mem.	206.19	2	103.10	2.60	.080
	Env.x Comp.x Memload	189.53	6	31.59	.80	.575
	Mem.x Comp.x S/Env.	3486.94	88	39.62		
Total		27212.01	287	94.81		

Insert Table 14 about here

Simple Effects Analysis

The significant Memory Load by Environment interaction for the number of alphabetic keystrokes executed suggests that keystrokes are executed differentially in the four environments over the three Memory Load levels. A simple effects analysis was conducted to determine what caused this difference. A significant difference was found for the High Memory Load level, F(3,44)=16.01, p=.000, the Medium Load level, F(3,44)=5.63, p=.002, and the Low Memory Load level, F(3,44)=2.84, p=.048.

Duncan's multimean analysis was performed to determine where the differences found in the simple effects analysis were located. This analysis revealed that in the High Memory Load level, more alphabetic keystrokes were executed in the Switch condition (66.33) than in the other three conditions (Window - 53.79, Notes - 54.25, Memorization - 49.20) This was also true for the Medium Memory Load level. In the Low Memory Load level, more alphabetic keystrokes were executed in the Switch condition than in the Notes condition (see Table 12). Therefore, the significant Environment main effect can be interpreted. The results of the simple effects analysis are summarized in Table 15. A graph of the means collapsed on Complexity are presented in Figure 4.

Insert Figure 4 about here

Insert Table 15 about here

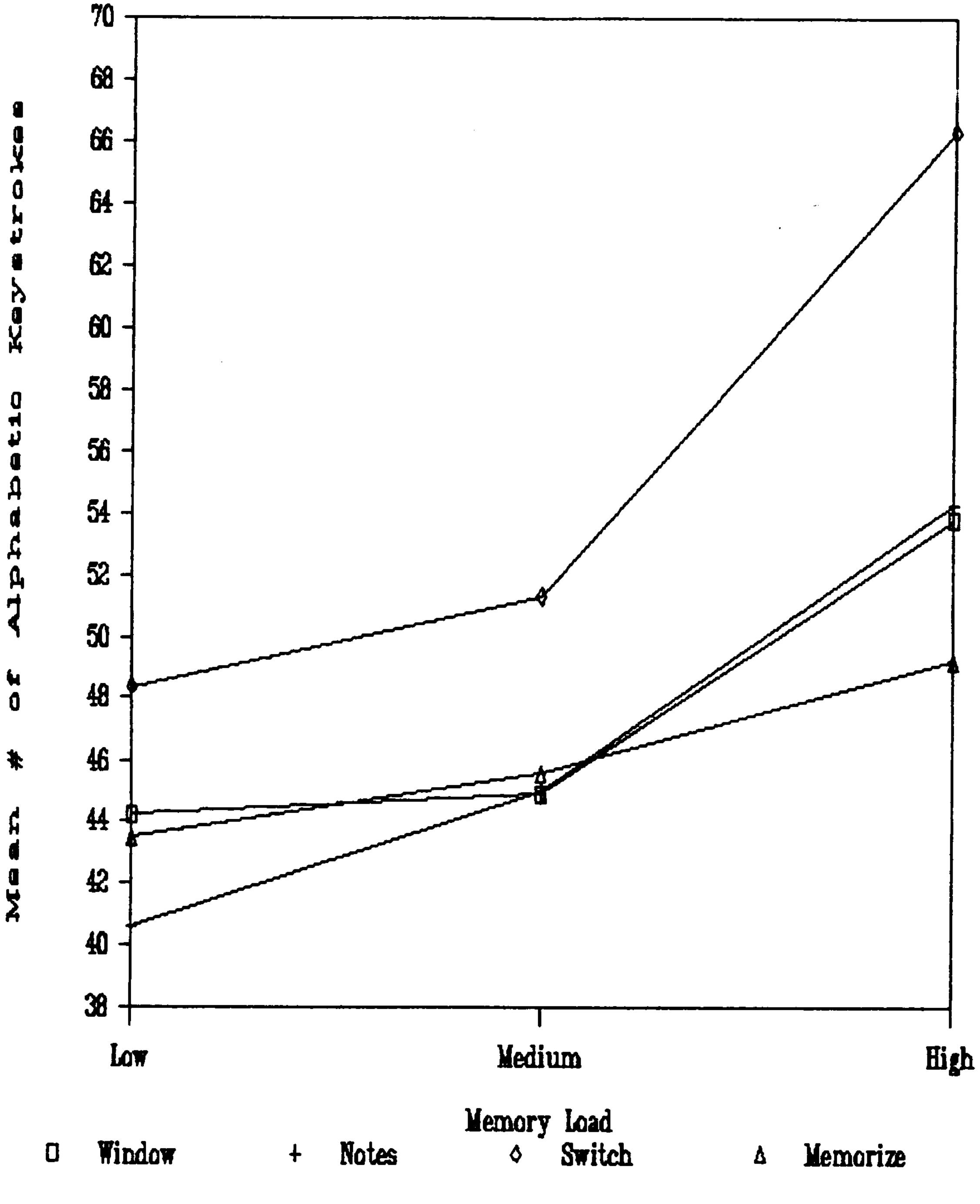


Figure 4. Environment by Memory Loac Interaction for the Mean Number of Alphabetic Keystrokes Executed.

Table 15

Summary of Simple Effects F Tests on the Number of Alphabetic Keystrokes Executed

Memory Load - High

Source	SS	DF	MS	F	P
Environment	1929.60	3	643.20	16.01	.0000
S/Environment	1767.37	44	40.17		

Results of Duncan's Multiple Range Analysis *

Environment	Memory	Window	Notes	Switch
Mean Value	49.21	53.79	54.25	66.33
	------------	------------	------------	

Memory Load - Medium

Source	SS	DF	MS	F	P
Environment	345.42	3	115.14	5.63	.002
S/Environment	899.00	44	20.43		

Results of Duncan's Multiple Range Analysis *

Environment	Window	Notes	Memory	Switch
Mean Value	44.91	45.00	45.58	51.33
	------------	------------	------------	

Memory Load - Low

Source	SS	DF	MS	F	P
Environment	382.94	3	127.64	2.849	.048
S/Environment	1971.04	44	44.80		

Results of Duncan's Multiple Range Analysis *

Environment	Notes	Memory	Window	Switch
Mean Value	40.46	43.50	44.25	48.37
	------------	------------	------------	
		------------	------------	------------

* Means with a common line do not differ significantly at $p<.05$.

The analysis of alphabetic keystrokes indicates that fewer alphabetic keystrokes were executed in the Notes condition than in the Switch condition over all levels of Memory Load. This finding is interesting and appears to be a function of slightly fewer keystrokes executed in the Low Memory Load condition than in the other conditions. The significant interaction indicates that as memory load increases, more alphabetic keys are pressed in the Switch condition than in the other three conditions. This was probably caused by entering the supplementary file several times as memory load increased. In order to access the supplementary file, the user had to type the file name in each time. The more frequently the user accesses the supplementary file, the more time he or she spends opening and closing the window.

ANALYSES FOR HYPOTHESIS 4

Hypothesis 4 stated that there would be a significant interaction between Complexity and Environment when the number of scrolling keys was used as the dependent measure. Scrolling keystrokes included any use of the up or down cursor control keys. It was expected that more scrolling would occur in the Window condition than in the non-window conditions as Complexity increased. Although the MANOVA indicated that the hypothesized interaction between Complexity and Environment was not significant, an exploratory analysis was conducted to determine if significant differences occurred in some other location. Therefore, a 4 X (2 X 3) mixed, fixed effects ANOVA was performed using the number of up and down cursor keystrokes as the dependent measure. The means and standard deviations for the scrolling keys is presented in Table 16.

Insert Table 16 about here

Box's M test for homogeneity of variance was significant, M=195.13, F(42,3233)=3.39, *p*=.000. Mauchly's test was also significant for the Memory Load effects, X^2 (2)=9.92, *p*=.007, and for the Complexity by Memory Load interaction, X^2 (2)=16.04, *p*=.000. Therefore, the Geisser-Greenhouse correction was used for these effects.

The ANOVA revealed a significant interaction between Memory Load and Environment, F(4.974,72.952)=5.35, *p*=.000. The ANOVA confirmed the findings of the MANOVA that there was no significant effect for the hypothesized interaction between Complexity and Environment, F(3,44)=.64, *p*=.591. There was also no significant Environment effect, F(3,44)=1.74, *p*=.173. The main effects for Complexity, F(1,44)=43.21, *p*=.000, and Memory Load, F(1.7,72.9)=9.57, *p*=.000, were significant. The Complexity main effect indicates that more scrolling keystrokes were executed in the High Complexity condition than in the Low Complexity condition (25.54 versus 15.70). The complete ANOVA table is presented in Table 17.

Insert Table 17 about here

Simple Effects Analysis

A graph of the mean number of scrolling keystrokes presented in Figure 4 indicates that the interaction between Memory Load and Environment appears to be in the High Memory Load level with the Switch

Table 16

Mean Number of Scrolling Keystrokes Executed *

		Memory Load			
Environment		High	Medium	Low	Marginal Means
Window	$\bar{x}$	23.29	16.37	28.62	22.76
	sd	19.47	15.83	21.34	18.15
Notes	$\bar{x}$	18.50	12.58	17.54	16.21
	sd	10.46	6.74	6.48	6.52
Switch	$\bar{x}$	35.58	20.62	18.29	24.83
	sd	8.44	15.33	6.15	4.39
Memorize	$\bar{x}$	18.87	18.21	19.00	18.69
	sd	6.73	8.36	4.94	5.25
Marginal Means	$\bar{x}$	24.06	16.95	20.86	20.62
	sd	13.80	12.22	12.32	10.47

		Complexity		
Environment		High	Low	Marginal Means
Window	$\bar{x}$	29.25	16.28	22.76
	sd	22.50	15.36	18.15
Notes	$\bar{x}$	19.94	12.47	16.21
	sd	10.49	3.80	6.52
Switch	$\bar{x}$	30.00	19.67	24.83
	sd	9.31	5.00	4.39
Memorize	$\bar{x}$	22.97	14.42	18.69
	sd	7.32	4.55	5.25
Marginal Means	$\bar{x}$	25.54	15.71	20.62
	sd	13.98	8.75	10.47

* n = 12 for each group.

Table 17

Analysis of Variance on the Number of Scrolling Keystrokes Executed

	Source	SS	DF	MS	F	P
Between Subjects	Env.	3277.36	3	1092.45	1.74	.000
	S/Env.	27650.14	44	628.41		
Within Subjects	Complex	6962.00	1	6962.00	43.21	.000
	Env.x Complex	311.58	3	103.86	.64	.591
	Comp.x S/Env.	7089.75	44	161.13		
	Memload	2437.90	1.658	1218.95	9.57	.000
	Env. x Mem.	4091.24	4.474	681.87	5.53	.000
	Mem.x S/Env.	11205.86	72.952	127.34		
	Comp.x Mem.	13035.77	2	6517.89	38.83	.000
	Env.x Comp.x Memload	1101.31	6	183.55	1.09	.373
	Mem.x Comp.x S/Env.	14772.58	88	167.87		
Total		91935.49	287	320.33		

* Degrees of freedom are adjusted by the Greenhouse - Geisser Epsilon.

condition. An analysis of simple effects was conducted to statistically determine where the interaction was located. The simple effects analysis revealed a significant difference at the High Memory Load level, F(3,44)=5.05, p=.004 . There was no difference at the Medium Memory Load level, F(3,44)=.92, p=.439, or at the Low Memory Load level, F(3,44)=2.325, p=.087. A Duncan's multimean test indicated that in the High Memory Load condition, more scrolling keystrokes were executed in the Switch condition than in the other three conditions (35.58 versus 23.29, 18.5, and 18.87). The results from the simple effects analysis are summarized in Table 18. Figure 5 graphically presents the interaction between Memory Load and Environment for the number of scrolling keys executed.

Insert Table 18 about here

Insert Figure 5 about here

The simple effects analysis indicated that more scrolling keystrokes occurred in the Switch condition than in the other three conditions in the High Memory Load level. This is thought to be due to the fact that the up and down cursor keys actually have two functions. The first is to move the cursor up and down, while the second is to scroll or move actual lines of text up or down in the window. This second function occurs only when the cursor is at the bottom of the window and the down arrow is pressed causing the text to scroll upward. In the three non-window conditions, the up and down arrow keys were probably used more often to move the cursor up and down rather than move text. However, in the Window condition, this same key was probably used more often to move text up and

Table 18

Summary of Simple Effects F Tests for the Number of Scrolling Keystrokes Executed

Memory Load - High

Source	SS	DF	MS	F	P
Environment	2294.10	3	764.70	5.05	.004
S/Environment	6658.21	44	151.32		

Results of Duncan's Multiple Range Analysis *

Environment	Notes	Memory	Window	Switch
Mean Value	18.50	18.87	23.29	35.58
	————	————	————	

Memory Load - Medium

Source	SS	DF	MS	F	P
Environment	413.85	3	137.95	.92	.434
S/Environment	6610.77	44	150.24		

Memory Load - Low

Source	SS	DF	MS	F	P
Environment	976.35	3	325.45	2.329	.087
S/Environment	6159.02	44	139.97		

Results of Duncan's Multiple Range Analysis *

Environment	Notes	Switch	Memory	Window
Mean Value	17.54	18.29	19.00	28.62
	————	————	————	
			————	————

* Means with a common line do not differ significantly at $p<.05$.

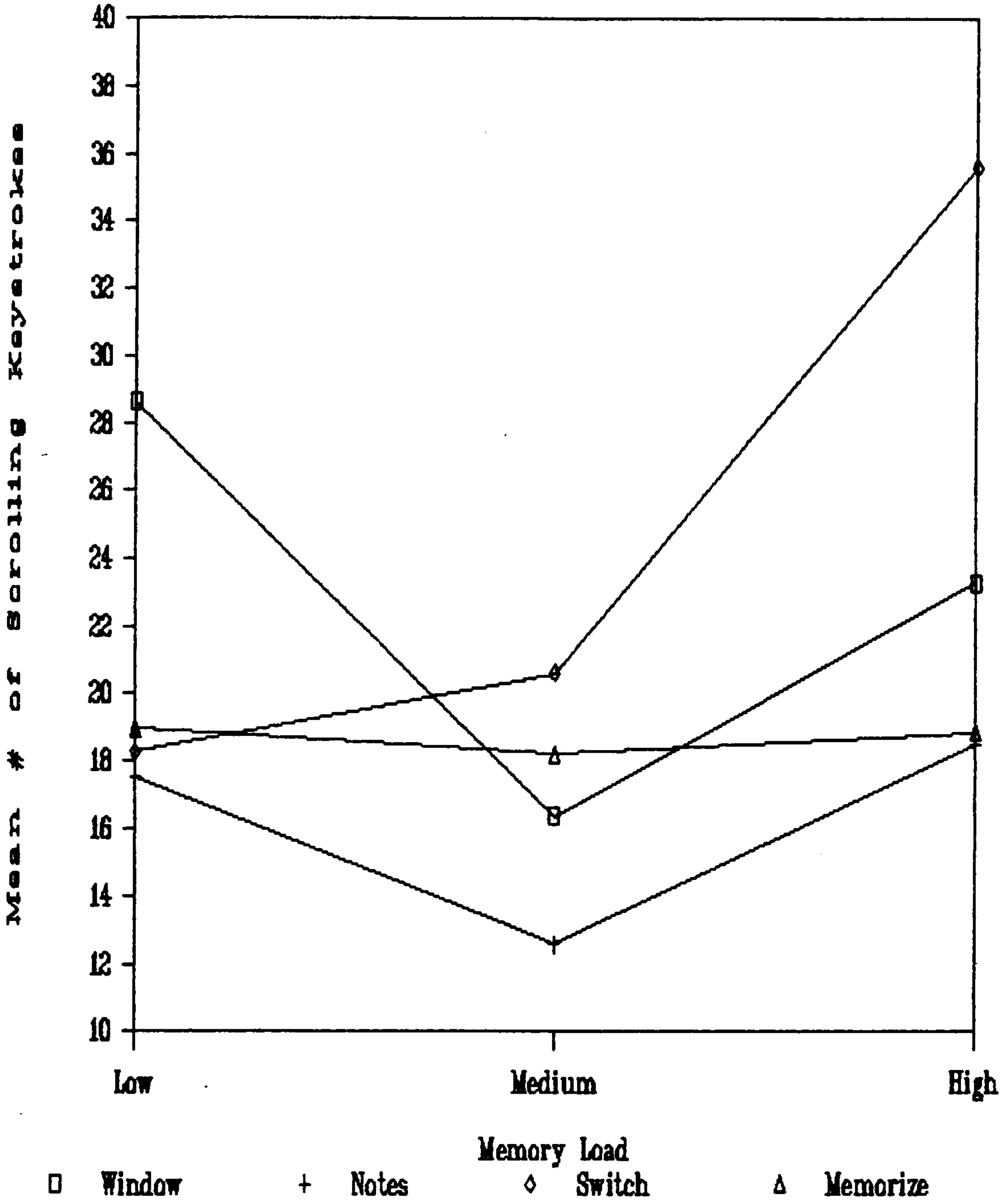


Figure 5. Environment by Memory Load Interaction for the Mean Number of Scrolling Keystrokes Executed.

down. The large number of scrolling keystrokes executed in the Switch condition under High Memory Load was probably a function of entering the supplementary file more than once and moving the cursor upon each access of the supplementary file. Unfortunately, it is impossible to separate the functions of this key. However, an examination of the number of Page Up and Page Down keys executed may illuminate the expected interaction between Complexity and Environment.

After examining the page up and down keystroke data, it was found that not all subjects used the paging keys for all tasks and in fact that for the Low Memory Load level, a large number of subjects did not use the paging keys. Therefore, it was thought inappropriate to use parametric statistics to analyze the data. The use of chi-square analysis, however, could aid in the understanding of the use of the page up and down keys. The number of subjects who had used the page up and down keys was compared to the number of subjects who had not used the keys for each of the four environments. A chi-square analysis was performed on each of the three Memory Load levels and on each of the two Complexity levels. Although this could be considered as capitalizing on chance, this analysis offers the best understanding of the relationship between the different environments for the paging keystrokes.

A significant Environment difference was found on the High Memory Load level, X^2 (3)=13.02, *p*=.005, but not on the Medium Load level, X^2 (3)=4.44, *p*=.217. A significant effect was also found in the Low Memory Load level, X^2 (3)=20.57, *p*=.000. For Complexity, a significant effect was found for the Low Complexity Level, X^2 (3)=16.58, *p*=.000, but not for the High Complexity Level, X^2 (3)=3.75, *p*=.289.

The observed frequencies for the number of people who used the paging keys are presented in Table 19. In examining the cells for the High Memory Load effect, 10 of the 12 subjects in the Window condition paged at least once compared to 5 of 12 in the Notes condition and 3 and 2 of 12 from the Switch and Memorization conditions respectively. In the Low Memory Load condition, 9 subjects in the Window condition paged compared to 3, 0, and 1 in the Notes, Switch and Memorization conditions respectively. Finally, in the Low Complexity condition, 9 subjects again paged in the Window condition and 3, 0, and 3 subjects paged in the Notes, Switch and Memorization conditions respectively. A graph of the mean number of paging keystrokes executed for Complexity and Environment is shown in Figure 6.

Insert Table 19 about here

Insert Figure 6 about here

Chi-square analysis is typically very sensitive to sample size. Any significant findings using this technique indicate that a strong relationship probably exists. The results indicate that a large number of subjects in the Window condition, under any memory load or complexity level, use the paging keys as compared to subjects in the other conditions. Subjects in the non-window conditions tended to use the paging function in the High and Medium Memory Load conditions rather than in the Low Memory Load condition. Also, few subjects in the non-window conditions used the paging function in the Low Complexity level, whereas many more used the function in the High Complexity level. Therefore, it appears that regardless of the situation, many subjects using a window

Table 19A

Chi Squares and Frequency Distributions for the Number of Subjects Who Executed Paging Keystrokes by Memory

High Memory Load
Chi-Square = 13.029, DF = 3, P = .005.

Environ.	Window	Notes	Switch	Memorize	Row Total
No Page	0 = 2	7	9	10	28
Page	0 = 10	5	3	2	20
Column Total	0 = 12	12	12	12	48

Medium Memory Load
Chi-Square = 4.443, DF = 3, P = .217.

Environ.	Window	Notes	Switch	Memorize	Row Total
No Page	0 = 3	3	7	6	19
Page	0 = 9	9	5	6	29
Column Total	0 = 12	12	12	12	48

Low Memory Load
Chi-Square = 20.571, DF = 3, P = .0001.

Environ.	Window	Notes	Switch	Memorize	Row Total
No Page	0 = 3	9	12	11	35
Page	0 = 9	3	0	1	13
Column Total	0 = 12	12	12	12	48

Table 19B

Chi Squares and Frequency Distributions for the Number of Subjects Who Executed Paging Keystrokes by Complexity

High Complexity
Chi-Square = 3.750, DF = 3, P = .290.

Environ.	Window	Notes	Switch	Memorize	Row Total
No Page	0 = 3	2	5	6	16
Page	0 = 9	10	7	6	32
Column Total	0 = 12	12	12	12	48

Low Complexity
Chi-Square = 16.582, DF = 3, P = .0009.

Environ.	Window	Notes	Switch	Memorize	Row Total
No Page	0 = 3	9	12	9	33
Page	0 = 9	3	0	3	15
Column Total	0 = 12	12	12	12	48

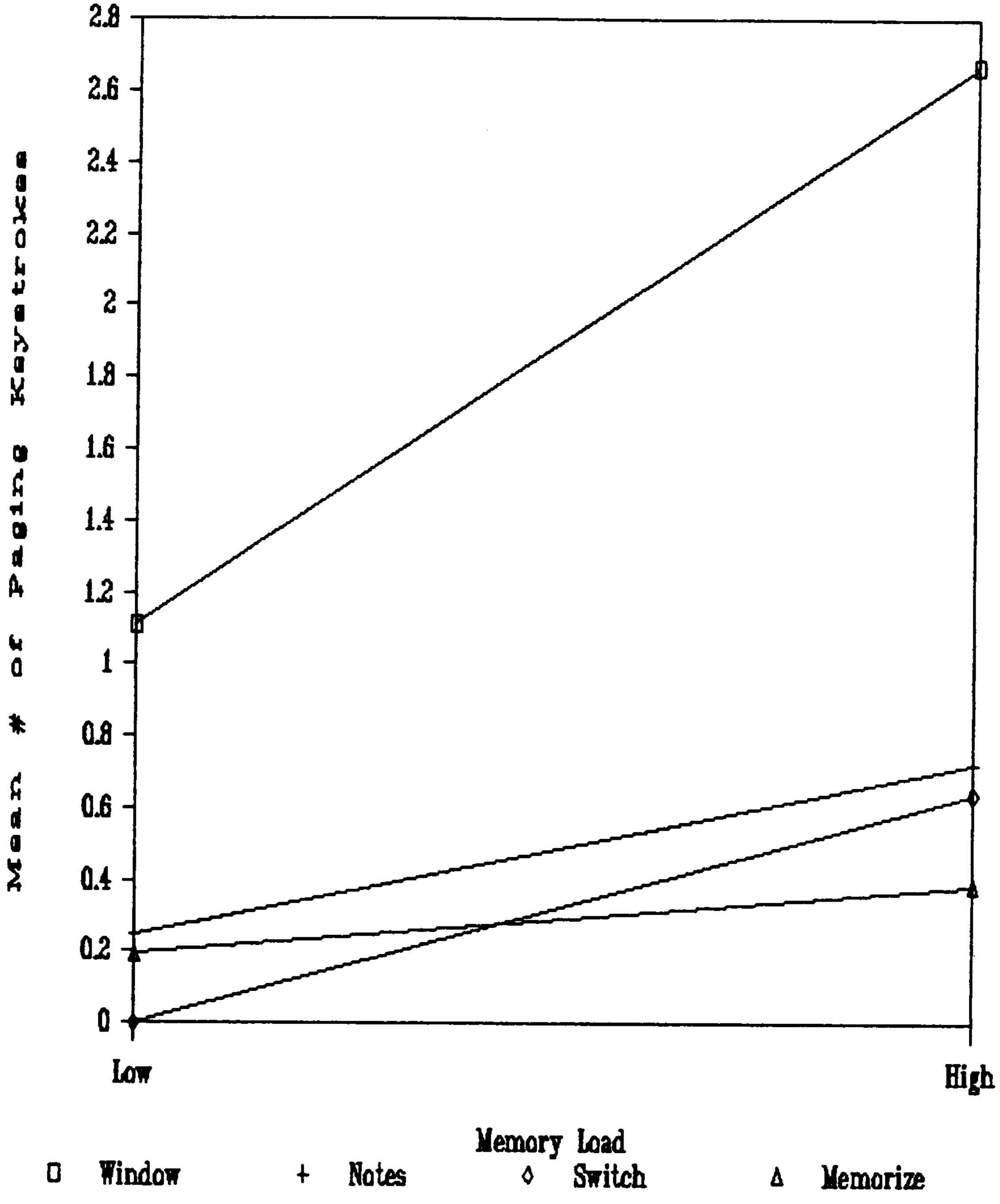


Figure 6. Mean Number of Paging Keystrokes

Executed by Complexity Level.

system will tend to page up and down. However, the use of these keys in the non-window situations will tend to depend on the memory load and the complexity of the situation.

ANALYSES FOR HYPOTHESIS 5

Hypothesis 5 stated that there would be a significant interaction between Memory Load and Environment when the number of errors was used as the dependent measure. It was expected that more errors would occur in the Memorization condition than the three other conditions as Memory Load increased. After examining the error data, it was found that too few errors were made to be able to make use of parametric ANOVA's. However, chi-square analysis was used to compare the number of subjects who made one or more errors to those who made no errors in each of the environments for the different memory loads and complexity levels.

The errors the subjects made seemed to fall into three categories: 1) Word errors, or errors of omission, insertion or incorrect spelling of a word; 2) Order errors, or typing the words in the wrong order; and 3) Format errors, or errors such as adding or indenting a line or otherwise changing the appearance of the document. A chi-square analysis using each of these types of errors as dependent measures was performed to compare the Environments on each of the three Memory Load levels and two Complexity levels. Because the number of errors varied greatly for the different environments and typically was low, it was decided to compare the number of subjects who had made no errors to those who had made one or more errors.

Format Errors

At least a few subjects in each condition made a minimum of one format error. As discussed above, errors were classified as format errors if the subject somehow changed the document from the way it was meant to appear by adding a line, moving words over, or erasing words. Five people in the Window condition made at least one error, 5 subjects in the Notes condition also made an error. Eight subjects in the Memorize and 8 in the Switch conditions made an error.

The chi-square analyses indicated no significant effect in the High Memory Load condition, X^2 (3)=1.77, *p*=.62, in the Medium Memory Load condition, X^2 (3)=6.75, *p*=.083, or in the Low Memory Load condition, X^2 (3)=1.77, *p*=.62. There was also no effect in the Low Complexity condition, X^2 (3)=5.71, *p*=.12, or in the High Complexity condition, X^2 (3)=1.73, *p*=.63. These results indicate that format errors were equally distributed among the four conditions. A summary of the chi-square analyses is presented in Table 20.

Insert Table 20 about here

Order Errors

Examination of the data for the number of order mistakes indicated that no subjects made errors in the Window and Notes conditions. However, six out of twelve subjects in the Memorization conditions made errors in at least one level of the following analyses and two subjects in the Switch condition made at least one error. The chi-square analyses indicated a significant Environment difference in the High Memory Load level,

Table 20A

Chi Squares and Frequency Distributions for the Number of Subjects Who Committed Format Errors by Memory Load

High Memory Load
Chi-Square = 1.778, DF = 3, P < .620.

Environ.	Window	Notes	Switch	Memorize	Row Total
No Format Errors	0 = 9	8	6	7	30
Format Errors	0 = 3	4	6	5	18
Column Total	0 = 12	12	12	12	48

Medium Memory Load
Chi-Square = 6.750, DF = 3, P = .080.

Environ.	Window	Notes	Switch	Memorize	Row Total
No Format Errors	0 = 9	11	6	6	32
Format Errors	0 = 3	1	6	6	16
Column Total	0 = 12	12	12	12	48

Low Memory Load
Chi-Square = 1.778, DF = 3, P = .620.

Environ.	Window	Notes	Switch	Memorize	Row Total
No Format Errors	0 = 8	10	10	8	36
Format Errors	0 = 4	2	2	4	12
Column Total	0 = 12	12	12	12	48

Table 20B

Chi Squares and Frequency Distributions for the Number of Subjects Who Committed Format Errors by Complexity

High Complexity
Chi-Square = 1.731, DF = 3, P = .630.

Environ.	Window	Notes	Switch	Memorize	Row Total
No Format Errors	0 = 8	9	8	6	31
Format Errors	0 = 4	3	4	6	17
Column Total	0 = 12	12	12	12	48

Low Complexity
Chi-Square = 5.706, DF = 3, P = .127.

Environ.	Window	Notes	Switch	Memorize	Row Total
No Format Errors	0 = 9	8	4	5	26
Format Errors	0 = 3	4	8	7	22
Column Total	0 = 12	12	12	12	48

X^2 (3)=12.95, p=.005, but not for the Medium Memory Load level, X^2 (3)=3.911, p=.27, or the Low Memory Load level, X^2 (3)=6.26, p=.09. The analysis also indicated no significant Environment difference in the Low Complexity Condition, X^2 (3)=3.91, p=.27. However, a significant Environment effect was found in the High Complexity condition, X^2 (3)=12.95, p=.005.

Examination of the cells in Table 21 for the significant High Memory Load condition indicated that more subjects made order errors in the Memorization condition (5 out of 12) than in the other conditions. Also, more subjects in the Memorization (5 out of 12) condition made order errors in the High Complexity condition. This indicates that it is more difficult to type words in the correct order in either a high memory load or a high complexity situation when one has to memorize the information versus the three other methods tested.

Insert Table 21 about here

Word Errors

Word errors included omission of a word, inclusion of a wrong word in the text, and misspelling of a word. No word errors occurred in the Window condition. However, a total of 4 of 12 subjects in the Notes condition, 5 subjects in the Switch condition, and 5 subjects in the Memorization condition made at least one word error in all of the tasks. Despite these numbers, there were no significant chi-squares. For High Memory Load, X^2 (3)=6.22, p=.10, for Medium Memory Load, X^2 (3)=3.9, p=.27, and for Low Memory Load, X^2 (3)=3.91, p=.27. For Low Complexity,

Table 21A

Chi Squares and Frequency Distributions for the Number of Subjects Who Committed Order Errors by Memory Load

High Memory Load
Chi-Square = 12.952, DF = 3, P = .005.

Environ.	Window	Notes	Switch	Memorize	Row Total
No Order Errors	0 = 12	12	11	7	42
Order Errors	0 = 0	0	1	5	6
Column Total	0 = 12	12	12	12	48

Medium Memory Load
Chi-Square = 3.911, DF = 3, P = .271.

Environ.	Window	Notes	Switch	Memorize	Row Total
No Order Errors	0 = 12	12	11	10	45
Order Errors	0 = 0	0	1	2	3
Column Total	0 = 12	12	12	12	48

Low Memory Load
Chi-Square = 6.261, DF = 3, P = .100.

Environ.	Window	Notes	Switch	Memorize	Row Total
No Order Errors	0 = 12	12	12	10	46
Order Errors	0 = 0	0	0	2	2
Column Total	0 = 12	12	12	12	48

Table 21B

Chi Squares and Frequency Distributions for the Number of Subjects Who Committed Order Errors by Complexity

High Complexity
Chi-Square = 12.952, DF = 3, P = .005.

Environ.	Window	Notes	Switch	Memorize	Row Total
No Order Errors	0 = 12	12	11	7	42
Order Errors	0 = 0	0	1	5	6
Column Total	0 = 12	12	12	12	48

Low Complexity
Chi-Square = 3.911, DF = 3, P = .271.

Environ.	Window	Notes	Switch	Memorize	Row Total
No Order Errors	0 = 12	12	11	10	45
Order Errors	0 = 0	0	1	2	3
Column Total	0 = 12	12	12	12	48

X^2 (3)=5.85, p=.12, and for High Complexity X^2 (3)=6.95, p=.07. Table 22 summarizes the complete Chi-square analyses for words errors.

Insert Table 22 about here

At least a third of the people in each of the non-window conditions made at least one error. However, no one in the Window condition made an error. The failure of the chi-square statistic to detect a significant environment difference may be due to the fact that the expected frequencies for all of the distributions for this dependent variable were less than 5, in two cases as low as 1. Chi-square analysis is not very powerful when the sample size is small.

The results from the error data suggest several things. First, format errors most probably occur because the subject is in a hurry and does not want to take the time to correct the error. To the subject, a more serious error is leaving a word out or misspelling a word rather than indenting a line. The finding of equal numbers subjects making this type of error in the four conditions suggest that a speed/ accuracy trade-off is probably occurring but is happening equally in all of the environments. Differences in task completion times, then, are probably are not due to differential speed/accuracy trade-off motivations.

The second interesting finding is that when memorizing information, it is difficult to recall the exact order when there is a lot of information to remember or when the information is difficult to find. There was no problem in getting the correct order in the other three environments. Finally, probably the most serious errors are word errors as these represent an error completing the primary task. Not one error was made

Table 22A

Chi Squares and Frequency Distributions for the Number of Subjects Who Committed Word Errors by Memory Load

High Memory Load
Chi-Square = 6.222, DF = 3, P = .101.

Environ.	Window	Notes	Switch	Memorize	Row Total
No Word Errors	0 = 12	8	9	7	36
Word Errors	0 = 0	4	3	5	12
Column Total	0 = 12	12	12	12	48

Medium Memory Load
Chi-Square = 3.911, DF = 3, P = .271.

Environ.	Window	Notes	Switch	Memorize	Row Total
No Word Errors	0 = 12	12	11	10	45
Word Errors	0 = 0	0	1	2	3
Column Total	0 = 12	12	12	12	48

Low Memory Load
Chi-Square = 3.911, DF = 3, P = .271.

Environ.	Window	Notes	Switch	Memorize	Row Total
No Word Errors	0 = 12	12	11	10	45
Word Errors	0 = 0	0	1	2	3
Column Total	0 = 12	12	12	12	48

Table 22B

Chi Squares and Frequency Distributions for the Number of Subjects Who Committed Word Errors by Complexity

High Complexity
Chi-Square = 6.958, DF = 3, P = .073.

Environ.	Window	Notes	Switch	Memorize	Row Total
No Word Errors	0 = 12	10	8	7	37
Word Errors	0 = 0	2	4	5	11
Column Total	0 = 12	12	12	12	48

Low Complexity
Chi-Square = 5.854, DF = 3, P = .120.

Environ.	Window	Notes	Switch	Memorize	Row Total
No Word Errors	0 = 12	10	11	8	41
Word Errors	0 = 0	2	1	4	7
Column Total	0 = 12	12	12	12	48

in the Window condition. Yet, a large number of subjects made errors in the other conditions. In the Notes condition, the subjects might have incorrectly transcribed the information. In both the Switch and Memorization conditions, the subject's memory is probably to blame. In the Window condition, the window offers the most exact representation of the original data to the subject than any of the other methods. There is much less chance of adding a word or misspelling a word when the document is directly in front of the subject.

ANALYSES FOR HYPOTHESES 6 AND 7

It was hypothesized that there would be a significant interaction between Complexity and Environment and between Memory Load and Environment when the Modified Cooper-Harper workload ratings were used as the dependent measures. Workload ratings were thought to be higher in the Memorization condition than in the Window condition as Memory Load increased. The means for the Modified Cooper-Harper rating scale are presented in Table 23.

Insert Table 23 about here

The Modified Cooper-Harper workload scale results in ratings which are ordinal in level of measurement rather than interval. Parametric statistical techniques, such as analysis of variance, assume that the dependent measures are derived from an interval scale. However, nonparametric tests do not make this assumption. Accordingly, the scores from the rating scale technically should be analyzed using a non-

Table 23

Mean Modified Cooper-Harper Scale Ratings *

		Memory Load			
Environment		High	Medium	Low	Marginal Means
Window	$\bar{x}$ sd	2.292 0.752	2.042 0.542	2.042 0.689	2.125 0.528
Notes	$\bar{x}$ sd	2.583 1.145	2.458 0.542	2.042 0.656	2.361 0.631
Switch	$\bar{x}$ sd	4.000 2.246	3.333 1.934	2.750 1.790	3.361 1.868
Memorize	$\bar{x}$ sd	4.750 1.948	2.917 1.062	3.000 1.567	3.555 1.292
Marginal Means	$\bar{x}$ sd	3.406 1.884	2.687 1.232	2.458 1.312	2.851 1.324

		Complexity		
Environment		High	Low	Marginal Means
Window	$\bar{x}$ sd	2.444 0.770	1.806 0.540	2.125 0.528
Notes	$\bar{x}$ sd	2.417 0.534	2.306 1.049	2.361 0.631
Switch	$\bar{x}$ sd	3.583 1.837	3.139 1.935	3.361 1.868
Memorize	$\bar{x}$ sd	3.889 1.526	3.222 1.305	3.555 1.292
Marginal Means	$\bar{x}$ sd	3.083 1.410	2.618 1.399	2.851 1.324

* n = 12 for each group.
Means and not medians are presented.

parametric test. Unfortunately, the author is unaware of any non-parametric technique which can be used to test a fixed effects, mixed design model with more than two levels of the within factors. Nevertheless, it was decided to use the Kruskal-Wallis one way ANOVA to examine Environment differences on the three Memory Load levels and the two Complexity levels. Although it is possible that this is capitalizing on chance, it allows for an approximation of testing an interaction by essentially conducting a simple effects analysis. Differences found using this non-parametric technique would suggest that true differences existed and conventional parametric techniques would then be used to determine if an interaction was present between the within and between factors.

The Kruskal-Wallis one way ANOVA's employed on each of the Memory Load levels indicated that significant Environment differences, corrected for ties, existed in the High Memory Load level, X^2 (48)=15.45, p=.001 and the Medium Memory Load level, X^2 (48)=7.82, p=.05, but not for the Low Memory Load level, X^2 (48)=2.44, p=.48. The Kruskal-Wallis test also demonstrated significant Environment differences on the two complexity levels, X^2 (48)=9.16, p=.027 for the High Complexity level, and X^2 (48)=11.66, p=.008 for the Low Complexity level. These tests indicate that differences exist between Environments on Memory Load and Complexity and there is a possibility of an interaction.

A 4 X (2 X 3) fixed effects, mixed ANOVA was then performed to test for the interaction effects. The hypothesized interaction between Complexity and Environment was not significant, F(1,44)=.89, p=.452. However, the hypothesized interaction between Memory Load and Environment was significant, F(2,88)=3.58, p=.003. A significant Environment main effect was found, F(3,44)=4.18, p=.011, as well as a Complexity main effect, F(1,44)=11.82, p=.001, and a Memory Load main effect, F(2,88)=17.93, p=.000. This indicated that the Memory Load by Environment interaction probably existed and a simple effects analysis was performed to determine where the differences were. The Complexity main effect indicates that subjects judged the workload to be higher in the High Complexity condition than in the Low Complexity condition (3.08 versus 2.61). The complete ANOVA table is presented in Table 24.

Insert Table 24 about here

Simple Effects Analysis

A parametric simple effects analysis was conducted on the workload scores. A significant effect was found in the High Memory Load level, F(3,44)=6.087, p=.001. A marginally significant effect was found in the Medium Memory Load level, F(3,44)=2.75, p=.053 and no significant effect was found in the Low Memory Load level, F(3,44)=1.76, p=.167. A Duncan's post hoc analysis indicated that, as hypothesized, in the High Memory Load condition, subjects in the Memorization and Switch conditions had higher workload scores (4.75 and 4.00) than subjects in the Window or Notes conditions (2.29 and 2.58). In the Medium Workload condition, subjects in the Window condition had lower workload scores than those subjects in the Switch condition (2.04 versus 3.33). The results for the Medium condition should be interpreted with caution given the marginally significant F ratio. In the Low Memory Load condition, there were no statistical differences between conditions. Table 25 shows the complete

Table 24

Analysis of Variance on Modified Cooper-Harper Scale Ratings

	Source	SS	DF	MS	F	P
Between Subjects	Env.	109.70	3	36.57	4.18	.011
	S/Env.	385.04	44	8.75		
Within Subjects	Complex	15.59	1	15.59	11.82	.001
	Env.x Complex	3.54	3	1.18	.89	.452
	Comp.x S/Env.	58.04	44	1.32		
	Memload	46.97	2	23.48	17.93	.000
	Env. x Mem.	28.12	6	4.69	3.58	.003
	Mem.x S/Env.	115.25	88	1.31		
	Comp.x Mem.	4.72	2	2.36	1.93	.151
	Env.x Comp.x Memload	6.20	6	1.03	.85	.537
	Mem.x Comp.x S/Env.	107.42	88	1.22		
Total		880.59	287	3.07		

results of the simple effects analysis. The mean workload scores for the Memory Load by Environment interaction are graphed in Figure 7.

Insert Figure 7 about here

Insert Table 25 about here

The results from the workload questionnaire indicate that subjects found it more difficult to perform tasks in the Switch and Memorization conditions than the Window and Note conditions as Memory Load increased. Workload score for the Notes condition was probably low because note taking is a very familiar response for most people and as such, subjects probably felt comfortable in what they were doing. Subjects in the Window condition did not have to expend much mental energy to complete their tasks because all of the information was right in front of them. Subjects in the Switch condition may have had to expend energy in order to access the supplementary file several times. In the Memorization condition, the problems occurred from the fact that information had to be memorized, placing the subject under a great deal of stress.

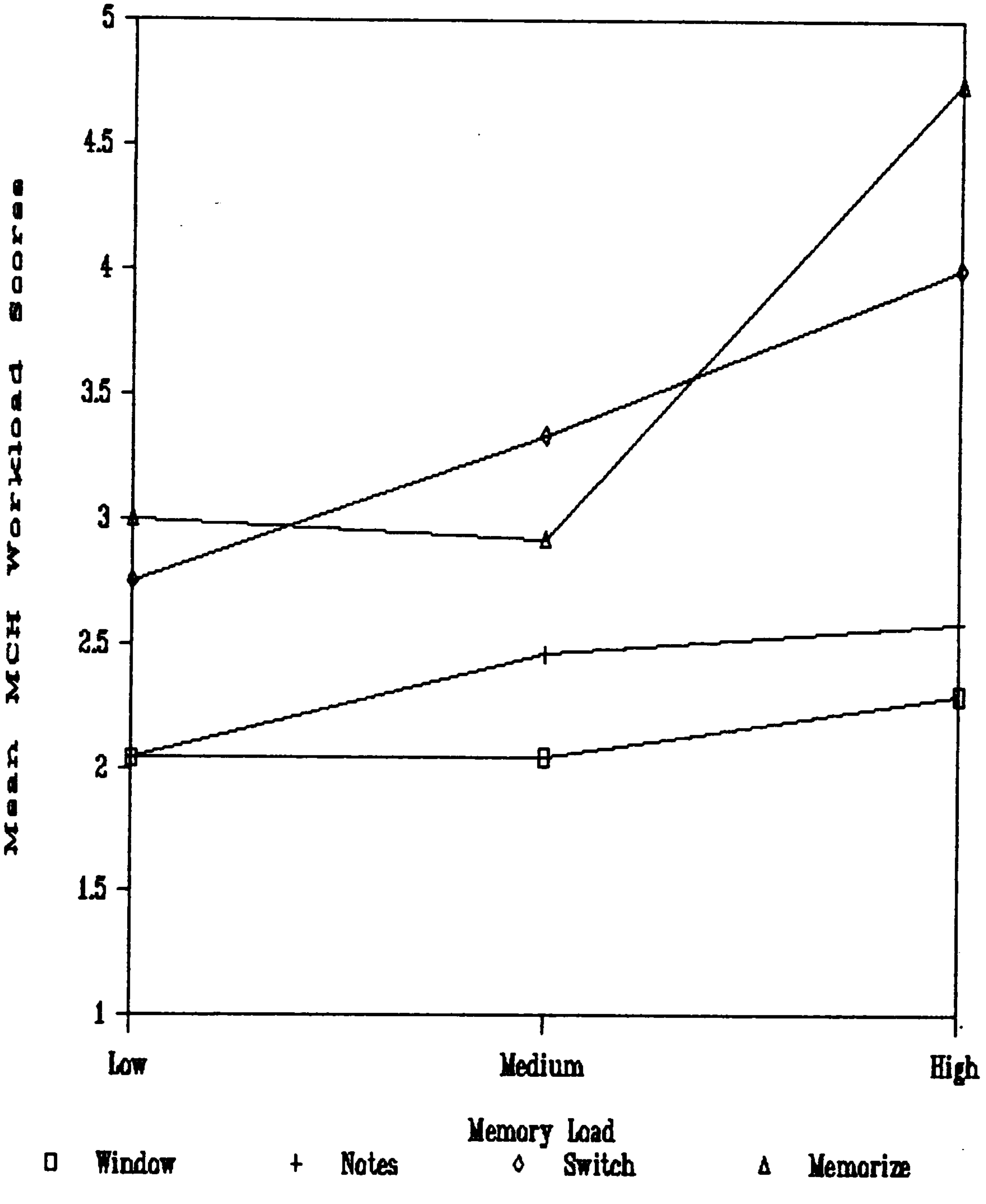


Figure 7. Environment by Memory Load Interaction for the Mean Modified Cooper-Harper Workload Ratings.

Table 25

Summary of Simple Effects F Tests for Modified Cooper-Harper Scale Ratings

Memory Load - High

Source	SS	DF	MS	F	P
Environment	48.93	3	16.31	6.09	.001
S/Environment	117.89	44	2.67		

Results of Duncan's Multiple Range Analysis *

Environment	Window	Notes	Switch	Memory
Mean Value	2.292	2.583	4.000	4.750
	————	————	————	————

Memory Load - Medium

Source	SS	DF	MS	F	P
Environment	11.27	3	3.76	2.75	.053
S/Environment	60.04	44	1.36		

Results of Duncan's Multiple Range Analysis *

Environment	Window	Notes	Memory	Switch
Mean Value	2.042	2.458	2.916	3.333
	————	————	————	
		————	————	————

Memory Load - Low

Source	SS	DF	MS	F	P
Environment	8.71	3	2.90	1.769	.167
S/Environment	72.21	44	1.64		

* Means with a common line do not differ significantly at $p<.05$.

DISCUSSION

This study was designed to answer two main questions. First, why would performance using a window environment be better than performance using a non-window environment? Second, when or what task parameters determine the usefulness (defined in terms of task time and errors) of a window environment? These questions were answered in relation to a task which required subjects to search for information from a supplementary file and place that information in a main file.

It was hypothesized that the reason why windows would be more useful than non-windows was that they would reduce the amount of resources required to complete a task by acting as efficient memory aids. The operator need not have to spend time and energy writing information, memorizing information, and/or accessing files several times. Rather the user could view the needed information directly and use the concept of preview to find information in one window while typing it in another window. It was also thought that tasks performed on a system which required the user to execute fewer, less complicated keystrokes, and which reduced the amount of time and energy the user expended in locating and remembering information would be faster than a system that did not reduce these things. Windows were believed to offer a system which could reduce the number of responses a user was required to execute to complete a task, as well as the amount of time information had to be held in memory over many non-window systems.

Generally, the hypotheses proposed for this study were supported. However, some results occurred which were not expected. The most disappointing result was the difficulty in finding an interaction between Memory Load and Environment on task completion times. A short summary of the findings for each hypothesis will be presented below. Following this, the results will be discussed in terms of the advantages and disadvantages of using windows for tasks. Finally, some predictions and implications from this study will be presented.

Hypotheses 1 and 2. A MANOVA was performed on task completion times, total number of responses, and the number of scrolling keystrokes executed to ensure results on correlated dependent measures could be interpreted. This analysis indicated that a significant interaction between Memory Load and Environment existed. However, the interaction between Complexity and Environment was not significant. There were also significant main effects for Environment, Complexity and Memory Load. Univariate ANOVA's were thus carried out on the significant effects.

Hypothesis 1 stated that there would be an interaction between Memory Load and Environment when task completion time was used as the dependent measure. It was expected that as Memory Load increased, tasks would be performed faster in the Window environment than in the Memorization condition because subjects in the Window condition would not have to spend time memorizing information. No interaction was found with a univariate ANOVA although a main effect for Memory Load was found with it taking longer to complete High Memory Load tasks (128.02 seconds) than Medium (88.63 seconds) and Low (80.62 seconds) Memory Load tasks.

Hypothesis 2 stated that there would be an interaction between Complexity and Environment using task completion time as the dependent measure. It was expected that tasks would be completed slower in the Window environment than in the Notes and Switch conditions because subjects would have to spend time scrolling through the window. The MANOVA indicated that there was no significant interaction between Environment and Complexity although a main effect did occur for Complexity (High - 110.9 seconds versus Low - 87.28 seconds).

Hypothesis 3. This hypothesis stated that more responses would be executed in the Switch and Notes conditions than in the Window and Memorization conditions as Memory Load increased. For the most part, this hypothesis was supported. More responses were found to occur for the Switch condition (140.28) than the Window (114.28) and Memorization (101.61) conditions in the High Memory Load condition. Also, more responses were found to occur in the Notes condition (124.45) than in the Memorization condition for High Memory Load. The number of responses executed is an important dependent measure because it is thought to be related to task completion time. The more responses which must be executed in a system to complete a task, the more time it will take to complete the task, all else held constant.

The total number of responses executed in completing tasks is a bottom line dependent measure which indicates very little about the actual benefits and drawbacks of various environments. It is important to break down the total number of responses into its major component parts in order to be able to make accurate conclusions as to why task performance using certain strategies should differ. In this vein, one might expect that more alphabetic keystrokes would be executed in the Switch condition than in the other conditions as Memory Load increased because the users would access the supplementary file several times. An exploratory analysis was conducted on the number of alphabetic keystrokes executed. More alphabetic keystrokes were found to be executed in the Switch condition (66.33 and 51.33) than in all three other conditions for both the High and Medium Memory Load conditions.

Hypothesis 4. This hypothesis stated that there would be an interaction between Environment and Complexity when the number of scrolling keystrokes was used as the dependent measure. In the Window condition, the subject could view about half of the information that subjects in the non-window conditions could view. Therefore, in the High Complexity situation, in which information is located in several parts of the document, subjects in the Window condition were expected to execute more scrolling keystrokes than in the non-window conditions.

This hypothesis was not supported. However, an exploratory analysis was carried out to determine if there were other possible differences. The exploratory analysis revealed a significant interaction between Memory Load and Environment. The interaction was rather surprising. Subjects in the Switch condition were found to execute more scrolling keystrokes (35.58) in the High Memory Load condition than subjects in the Notes, Memorization or Window conditions (18.50, 18.87, 23.29).

The failure to find that more scrolling keystrokes were executed in the Window condition in the High Complexity tasks prompted the question:

Were subjects using some other method to scroll such as using the page up and down keys more often than subjects in the other conditions? Exploratory chi-square analyses demonstrated that more subjects in the Window condition used paging keystrokes than subjects in the other conditions in the High and Low Memory Load and Low Complexity conditions. Again, there was no difference in the High Complexity situation.

Hypothesis 5. A significant interaction was expected between Environment and Memory Load for the number of errors made. As Memory Load increased, it was thought that more errors would be made in the Memorization condition than in the other three conditions. The Memorization condition was the only condition in which an external memory aid (i.e. other than mnemonics) was not available to the subject. This hypothesis was supported.

Examination of the number of people who made format errors indicated that there were no differences among the Environments on Complexity or Memory Load. However, examination of order errors indicated that more subjects in the Memorization condition made errors than subjects in the other conditions under High Complexity and High Memory Load situations. Finally, no word errors occurred in the Window system. However, a number of subjects in each of the other environments made errors of this type especially in the High Memory Load and High Complexity situations.

Hypotheses 6 and 7. An interaction was expected between Memory Load and Environment for workload ratings from the Modified Cooper-Harper rating scale. This rating scale attempted to objectify the subjective experience of mental workload. It was expected that as Memory Load increased, more workload would be experienced by subjects in the Memorization condition than in the Window condition. This hypothesis was supported. Less workload was experienced by subjects in the Window (2.29) and Notes (2.58) conditions than in the Switch (4.00) and Memorization (4.75) conditions in the High Memory Load situation. Also, less workload was experienced for the subjects in the Window condition (2.04) than in the non-window conditions (Notes - 2.45, Switch - 3.33, Memorization - 2.9) for the Medium Memory Load situation. There was no significant difference for workload ratings on the Low Memory Load level.

Advantages of Windows

As stated above, windows are thought to reduce the amount of resources used by an operator by acting as efficient memory aids. An efficient memory aid should reduce the number of responses required by the user as well as reduce the amount of time information has to be held in working memory. The advantage of windows as a memory aid should not occur in all situations, but rather should occur only under certain conditions, particularly high memory load. For instance, in a situation where a user must find a very small amount of information, it may not make any difference which system the user chooses. The task can probably be performed relatively quickly with little difference in task completion times in many environments. However, as the amount of information which the user must locate and use increases, the advantage of windows over non-windows in terms of task completion time should increase.

One cannot effectively discuss the advantages and disadvantages of windows over non-windows without first discussing the different mechanisms or strategies which operators may use in the non-window environment to cope with the problems of finding and remembering information. There are at least three common strategies which operators can use in a non-window situation to complete a search task such as that used in this study. The first is simply to attempt to memorize the information. An operator can also take notes on information and finally, an operator can switch back and forth between two or more files. The reason for possible advantages of window systems will be different for these three non-window environments. This next section uses evidence from the present study to discuss some of the advantages and disadvantages of using windows.

The specific advantages of windows which will be discussed include 1) a reduction in the amount of time information must be held in working memory because the window acts as a memory aid presenting supplementary information simultaneously with a main file; 2) a reduction in the number of responses which must be executed to complete tasks because subjects need not access files several times or take notes; 3) a reduction in the amount of errors because supplementary information can be directly viewed rather than held in memory or transformed into notes and 4) a reduction in the mental workload required to complete tasks because window systems reduce memory resources as well as response and selection resources. The disadvantages of windows include, 1) difficulty in locating information which is complex in presentation because the amount of information displayed is less than that of a non-window environment; and 2) execution of extra keystrokes when memory loads or complexity of information is low because users are unsure if they have seen 'the whole picture' even though they have.

Memory Aid

Windows are thought to act as very efficient memory aids. Operators using window systems can view information from a supplementary file directly. This capability reduces the need for time consuming memorization and rehearsal processes. The user can use preview to encode information from one window and at the same time begin typing information into the other window. However, this capability should only be demonstrated when working memory capacity has been exceeded. In the present study, there were no differences among the environments in the amount of time it took to complete tasks in either the low, medium or high Memory Load conditions.

One possible reason why the hypothesized interaction did not occur could be due to the inclusion of levels of Memory Load which were not large enough to demonstrate the desired effect. It appears that differences were beginning to occur in the High Memory Load condition when eight pieces of information were required. A graph of the non-significant means for task completion times shown in Figure 8 indicates that a trend may be beginning to occur such that at higher levels of memory load, significant differences among the environments may have been found. For example, if memory load had been increased to 20 pieces of information, the difference between the Window and Memorization conditions should have been much larger because in the Memorization condition, the user would have needed to spend a great deal of time rehearsing the information.

In contrast, however, the operator in the window system should not need to spend any time rehearsing the information as the information could always be displayed.

Insert Figure 8 about here

The levels of the Memory Load factor were chosen based on the number of 'chunks' thought to be below or above working memory capacity. The Low Memory Load was thought to easily be held in working memory while the Medium Memory Load was thought to be a little more difficult and the High Memory Load was thought to exceed working memory capacity. The contrast analysis on the main effects for Memory Load indicated that task completion times for the Low and Medium levels did not differ (80.61 seconds versus 88.63 seconds) yet both differed from the High level (128.02). It may be that the chosen levels were not different enough to allow an effect to occur.

The concept of 'chunks' is very individualized and when given information to memorize, subjects will tend to create chunks in order to be able to use the information. It is very possible that subjects in the Memorization condition were able to quickly develop a useable level of chunks (i.e. 7+/-2) from the information they were required to find. If more information had been required, the subjects may not have been able to chunk the items as quickly. This would also be true of the subjects in the Switch condition. Had they been required to find more information, they would have taken longer and had to access the supplementary file more often. Therefore, it is highly likely that a 'floor effect' occurred in

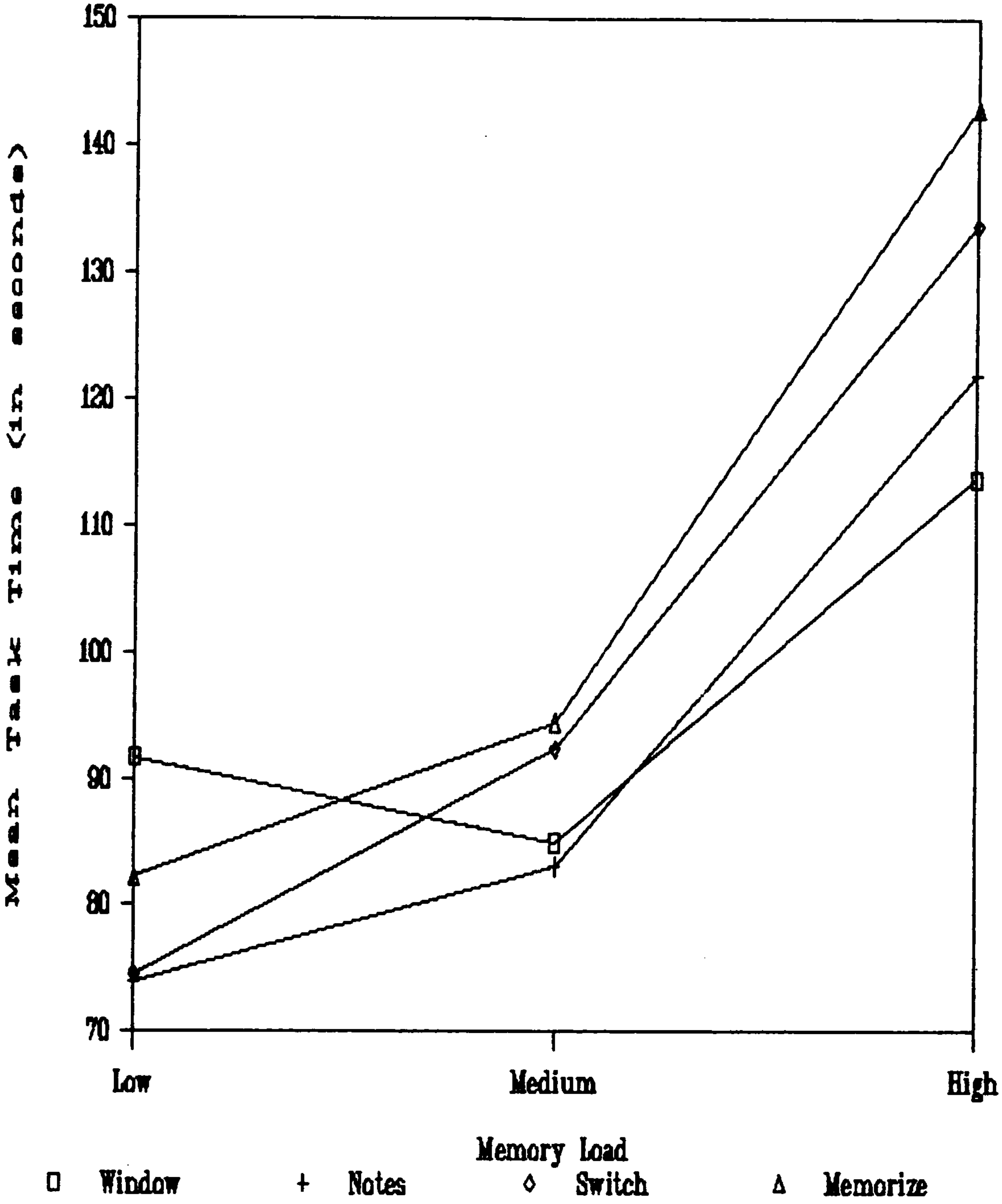


Figure 8. Plot of Task Completion Times for Environments by Memory Load.

this experiment in which the levels of the independent measure were not different enough for an effect to occur.

As memory load increases, one would think that operators using a memorization strategy would spend increasing amounts of time rehearsing the information. An indication of the amount of time which was required to memorize information in this study is provided by comparing the time subjects in the three non-window environments spent in the supplementary files. The difference between the Memorization and the Notes and Switch environments was essentially the amount of effort and time information must be rehearsed and held in working memory. Very little time and effort was required in the Notes condition, whereas somewhat more time was required for the Switch condition. Most of the time taken to complete a task in the Memorization condition was thought to occur as time spent rehearsing information. The user of a non-window memorization strategy was essentially given no memory aid except one's own ability to memorize information.

It would be expected that there should be no difference for the amount of time subjects spent in the supplementary files when memory load is below working memory capacity. However, as memory load increases, more time should be spent in the supplementary files in the Memorization condition than in the Switch or Notes conditions. Therefore, a mixed 3 X (2 X 3) (non-window environments X Complexity by Memory Load) ANOVA was performed on the amount of time spent in a window for the non-window environments to determine if these predictions were true. The means for the amount of time spent in a window are presented in Table 26. As expected, a significant interaction was found between the non-window

Table 26

Means for Time Spent in Windows for Non-window Strategies

Memory Load

Environment		High	Medium	Low	Marginal Means
Notes	$\bar{x}$	56.22	36.61	28.06	40.30
	sd	19.43	11.17	8.34	11.42
Switch	$\bar{x}$	61.75	43.84	26.61	43.93
	sd	16.46	21.08	8.14	11.50
Memorize	$\bar{x}$	70.90	35.14	24.60	43.55
	sd	15.79	20.07	5.40	11.02
Marginal Means	$\bar{x}$	62.96	38.53	26.29	42.74
	sd	17.88	17.90	7.34	11.11

Complexity

Environment		High	Low	Marginal Means
Notes	$\bar{x}$	40.92	39.67	40.30
	sd	10.01	15.02	11.42
Switch	$\bar{x}$	47.21	40.66	43.93
	sd	14.98	10.54	11.05
Memorize	$\bar{x}$	46.08	41.02	43.55
	sd	13.42	9.62	11.02
Marginal Means	$\bar{x}$	44.74	40.44	42.74
	sd	12.90	11.63	11.11

Table 27

Analysis of Variance on Amount of Time Spent in Windows for Non-window Conditions

	Source	SS	DF	MS	F	P
Between Subjects	Env.	574.26	2	287.13	.37	.691
	S/Env.	25348.57	33	768.14		
Within Subjects	Complex	16803.86	1	16803.86	73.16	.000
	Env.x Complex	16.03	2	—8.01	.03	.966
	Comp.x S/Env.	7579.93	33	229.69		
	Memload	50193.18	2	25096.59	85.29	.000
	Env. x Mem.	3250.79	4	812.70	2.76	.035
	Mem.x S/Env.	19421.27	66	294.26		
	Comp.x Mem.	2523.80	2	1261.90	6.27	.003
	Env.x Comp.x Memload	326.53	4	81.63	.41	.804
	Mem.x Comp.x S/Env.	13289.91	66	201.36		
Total		139328.13	215	648.04		

environments and Memory Load, F(4,66)=2.76, p=.035. The complete ANOVA table is presented in Table 27.

Insert Table 26 about here

Insert Table 27 about here

An analysis of simple effects shown in Table 28 indicates that there was no significant interaction in any of the places of interest. This finding was surprising and it is unsure why this occurred. However, the pattern of results as shown in Figure 9 indicates that the interaction should be where it was expected, namely in the High Memory Load condition between the Notes and Memorization conditions. Although not significant, the pattern does indicate that if memory load had been higher, there may have been more of a divergence between these methods. If this occurred, then support would be given to the contention that because no memory aid is provided to users in the Memorization condition, task completion times are slower than when using a memory aid of some type.

Insert Table 28 about here

Insert Figure 9 about here

Reduction in the Number of Errors

Additional evidence that a window system acts as an excellent memory aid comes from error data. In this study, errors were classified into three types: 1) format errors related to preserving the original integrity of the document; 2) order errors related to typing the words in the

Table 28

Summary of Simple Effects F Tests on Amount of Time Spent in Windows for Non-window Conditions

Memory Load - High

Source	SS	DF	MS	F	P
Environment	1319.93	2	659.96	2.20	.126
S/Environment	9875.40	33	299.25		

Memory Load - Medium

Source	SS	DF	MS	F	P
Environment	520.45	2	260.22	.803	.456
S/Environment	10694.30	33	324.07		

Memory Load - Low

Source	SS	DF	MS	F	P
Environment	72.15	2	36.07	.656	.526
S/Environment	1815.21	33	55.01		

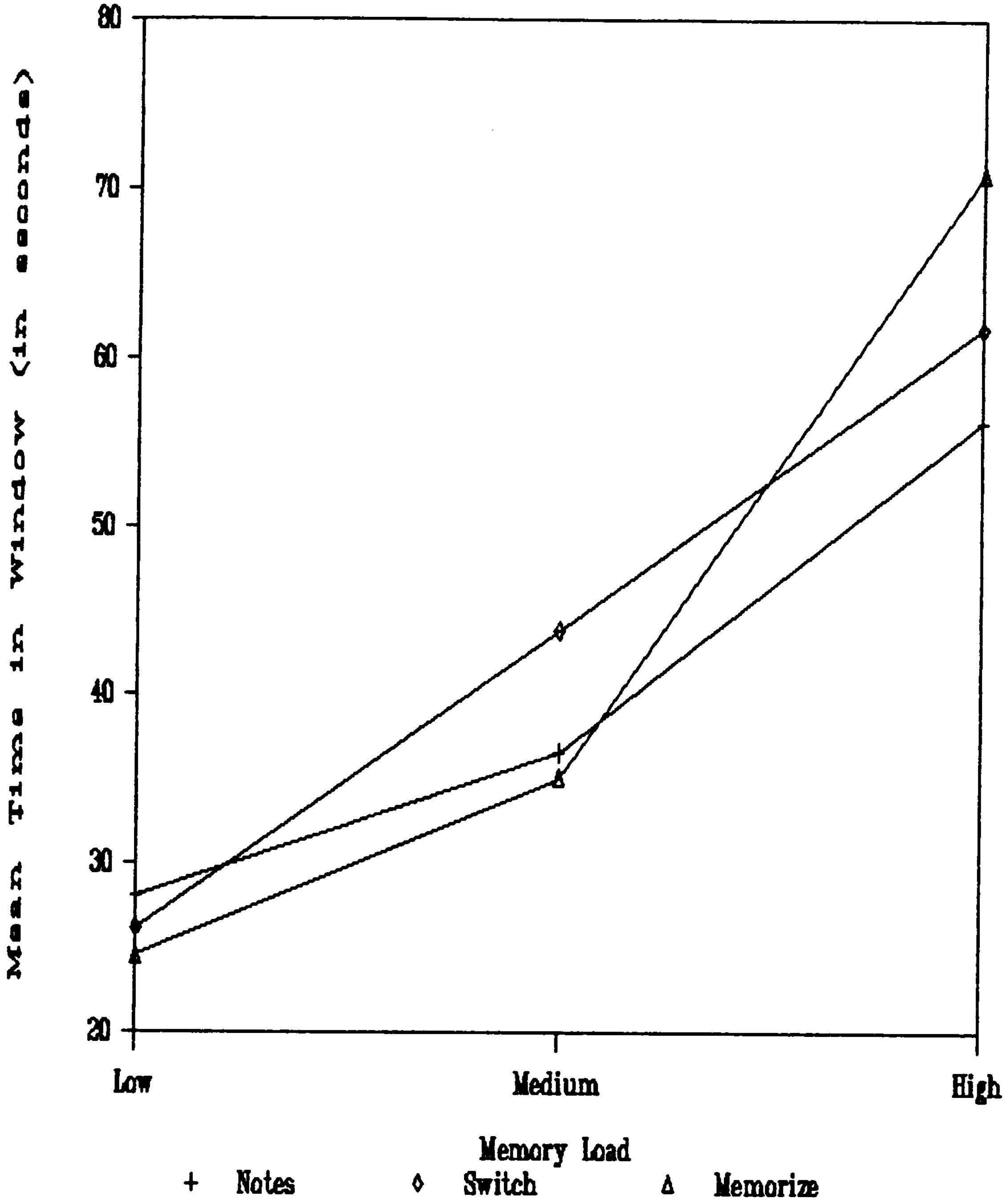


Figure 9. Non-window Environments by Memory Load Interaction for the Amount of Time Spent in Windows.

correct order in the document; and 3) word errors related to adding or misspelling words, using the wrong case or missing a word. Examination of the number of people who made format errors indicated that there were no differences between the Environments on Complexity or Memory Load. Format errors were thought to represent problems subjects had with using some of the formatting keys such as the insert or delete keys where they inserted spaces when they shouldn't have, etc. Failure to correct the errors may have been due to the subjects being unsure how to correct the problem or to a speed accuracy trade off. The finding that equal numbers of subjects made errors in the four environments indicates that although the speed accuracy trade off was occurring, it was occurring by equal numbers of subjects.

Order errors, on the other hand, were thought to represent problems with memory or memory aids in which the integrity of the original order of the words was somehow transformed or lost. More subjects in the Memorization condition made errors than subjects in the other conditions under High Complexity and High Memory Load situations. This indicates that subjects in the Memorization condition did have trouble in retaining the original order of the words and thus demonstrates a weakness of this system.

Finally, word errors were also thought to represent problems with memory and memory aids of systems. No errors of this type occurred in the Window system, indicating a strength of this system - since the original data can be viewed directly with no transformation, errors of this nature should not occur. However, a number of subjects in each of the other environments made errors of this type especially in the High Memory

Load and High Complexity situations. The failure to find statistical significance is thought to be a function of the low power of chi-square analysis with low sample size.

The results of the error analyses indicate that one major strength of the Window system is the exact representation of the supplementary file to the user. In each of the non-window systems, the users must rely on a memory aid of some sort which may not allow for a correct representation of a supplementary file. The inherent problems with the memory aids of each of the non-window systems is evidenced by the number of subjects making word errors and order errors in these environments.

Reduction in the Number of Responses

An efficient memory aid must reduce the number and complication of responses required by a system to complete a task. A system which requires fewer and less complicated keystrokes to be executed in order to complete a task is thought to reduce task completion time. All else held constant, the more responses which are executed, the more time will be taken to complete a task.

There was not a one to one relationship between task completion time and the number of responses executed in this study. This cannot be expected as many factors determine task completion time including searching for information and orientation in a file. However, the relationship is thought to be higher for systems which require more response resources than systems which require more memory resources. This can be demonstrated by examining the correlations between task time and the number of responses. One would think that when memory load is low, the correlation between task time and responses for all environments should be high and similar. However, as memory load increases, only environments which rely heavily on responses as memory aids should have high correlations. Environments which rely on memory resources should have much lower correlations.

Table 29 shows the correlations between task time and the number of responses for each environment. As expected, the correlations for the Window and Notes environments were all high and significant with the exception of two. In the Switch and Memorization conditions the highest correlations were found in the Low Memory Load situation. A relatively high, though nonsignificant correlation was also found in the High Memory Load condition for the Switch environment. If Memory Load had been higher, this result may have been significant. If so, the result may be due to accessing the supplementary file several times. In the Medium Memory Load condition for the Switch environment, the low and nonsignificant correlation indicates that apparently subjects were memorizing the information.

Insert Table 29 about here

The advantage of the window system should only be evidenced when working memory capacity is exceeded. Therefore, there should be no difference between environments when working memory capacity is not exceeded. The results of the present study support this contention. There was no difference between the environments for the total number of responses in the Low and Medium Memory Load levels. However, in the High Memory Load level, significantly more responses were executed in the

Table 29

Correlations Between Task Completion Times and the Number of Responses Executed *

	Memory Load		
Environment	High	Medium	Low

Window	r = .518
	p = .084
		.773
		.003
			.762
			.004

Notes	r = .664
	p = .018
		.727
		.007
			.535
			.073

Switch	r = .538
	p = .071
		.062
		.848
			.618
			.032

Memory	r = -.452
	p = .140
		.367
		.240
			.518
			.085

* n = 12 for each correlation.

Switch condition (140) than in the Memorization (101) and Window conditions (115). Also, more responses were executed in the Notes (124) condition than in the Memorization condition.

In the Memorization condition, subjects were limited in the keystrokes which they could execute because they could only access the supplementary file once, they had to use their minds as memory aids and they could view most of the text. Therefore, in this condition, subjects really only needed to execute alphabetic keystrokes and scrolling keystrokes. In the Window condition, the subjects viewed the supplementary and main files simultaneously, they could only open the window once and could not take notes.

More responses were thought to occur in the Notes condition because rather than attempting to remember information, subjects could take notes on the information. The action of taking notes required responses and as such was thought to take time. Information can be written at a faster rate than information can be typed (Seibel, 1972). Thus, the number of written notes was adjusted so that the two types of responses could be compared. Subjects in the Switch condition were also expected to execute a large number of responses because they would have to access the supplementary file several times in the High Memory Load condition. Each time they accessed the supplementary file, they were required to type the name of the file into the computer.

As discussed earlier, using the total number of keystrokes as a dependent measure is relatively uninformative providing little ability to make predictions. However, by the use of a slightly more fine grained analysis one can make much more accurate predictions for performance.

Therefore, the total number of keystrokes was broken down into alphabetic keystrokes, scrolling keystrokes and paging keystrokes. The number of paging keystrokes executed was found to be a disadvantage for windows and will be discussed later. However, several findings of interest for the number of alphabetic and scrolling keystrokes will be discussed here.

First, an exploratory analysis using the number of alphabetic keystrokes executed was conducted. This analysis indicated that more keystrokes were executed in the Switch condition (66.33 and 51.33) than in all three other conditions for both the High and Medium Memory Load conditions (Window - 53.8, 44.9; Notes - 54.2, 45.0; Memorization - 49.2, 45.58). These extra alphabetic keystrokes occurred when the subjects accessed the supplementary file several times. If memory load was increased to, for example, 20 pieces of information, the user would access the supplementary file a large number of times. Task time should increase at least by the amount of time it takes to open a window and type in the filename multiplied by the number of times the supplementary file was accessed.

Examination of the number of scrolling keys executed by subjects points to a drawback of the Switch non-window strategy. Subjects in the Switch condition executed more scrolling keystrokes (35.58) in the High Memory Load condition than subjects in the Notes, Memorization or Window conditions (18.50, 18.87, 23.29). Each time the user accessed the supplementary file or returned to the main file, the cursor located itself in a 'home' position in the top left hand corner of the screen. The user then moved the cursor down to the spot where he or she wanted to type. The large number of scrolling keystrokes executed in the Switch condition for High Memory Load tasks then, is a function of the default setting of a cursor always being located in the top left corner of a file after it had been opened. This is not simply an attribute of the computer system used in this experiment, rather it is an attribute which is used in almost every system.

The user of a Switch strategy must constantly navigate him or herself to the location of the needed information in the supplementary file each time it is accessed. Along with this, the user must find his or her place in the main file once the information has been located from the supplementary file. With files which are large, this will take time and in the process of navigating through the files, it is likely that the rehearsal processes will be disturbed and the user will forget the information which took so long to find.

Reduction of Mental Workload

Measures of mental workload can provide estimates of the amount of mental effort an operator must expend to use a system in order to complete a task. There are thought to be several different types of mental resources; two of these were expected to influence behavior in this study. First, a system which required a large amount of time and effort to place and hold information in working memory should require more cognitive resources from a user than a system which required less time and effort. Second, as the number and complexity of responses increase, the amount of mental effort required should also increase.

The results from the Modified Cooper-Harper rating scale indicated that subjects found tasks in the Memorization and Switch conditions more mentally demanding than subjects in the Notes and Window conditions when Memory Load was high. Subjects in the Memorization condition were forced to memorize eight pieces of information. Not only is this demanding, but the subjects were given only one chance to memorize the information. If they forgot the information, there was no way to complete the task. Subjects in the Switch condition also found it difficult to perform their tasks. They were allowed to access the supplementary file as often as they wished. However, this in itself may be difficult in that the subjects must remember how to access information, they must locate the information, and then get back into the main file. While they are performing these tasks, rehearsal may be interrupted and they may not be able to hold as much information in working memory as they had hoped.

Mental effort in the Window condition was low because users were not forced to use as much energy in maintaining information in working memory as users of memorization strategies. The information was continuously displayed in the Window condition. No special effort was required to develop strategies to place the information in working memory. Also, the subject in the Window condition did not rehearse information, and if information was forgotten, the user could simply glance down at the needed data.

Users of the Window condition also did not have to execute a large number of responses as in the Switch condition. As the number of complicated responses executed increase, the number of mistakes and difficulties can rise. The extra keystrokes can disrupt rehearsal processes causing the user to forget information, incorrectly type information in or to access the supplementary file again in order to ensure that the correct data was used. All of this increases the mental effort required by the user.

Although subjects in the Notes condition executed a large number of keystrokes, mental effort was judged to be relatively low by users. This is probably a function of two things. First, one would expect that subjects in this condition spent about the same amount of minimal time and effort placing and holding information in working memory as in the Window condition. Second, the responses of taking notes is a very familiar one to any college student and can be executed with very little difficulty. As you recall, it was thought that not only is it the amount of responses but also the complexity of those responses which will determine the usefulness of windows. Therefore, as the number of complex responses required to complete a task were probably equal in both the Window and Notes conditions, it is not surprising that there was no difference on the workload measure.

Disadvantages of Windows

The two major disadvantages of window systems are really related to the same thing. Window systems offer users a much smaller viewport to information with which they are working than non-window systems. Because of this, users of window systems must execute keystrokes in order to scroll through and view the information. A second problem which can occur, is that users can 'get lost' in the window and have difficulty locating information.

The first problem of executing keystrokes in order to view information was demonstrated with the examination of paging keystrokes. Paging keystrokes allowed the user to move an entire page of text up or down as opposed to scrolling which moved text up or down one line at a time. More subjects in the Window condition were found to use paging keystrokes in the High and Low Memory Load conditions than subjects in the three non-window environments. In order to view information, subjects were required to execute keystrokes. This may have been what provided null results in the Silver (1985) and Bury _et al._ studies.

The finding that subjects in the Window condition did not execute more scrolling keystrokes than subjects in the non-window conditions is not surprising because paging is a more efficient use of keystrokes. The decision complexity advantage states that the more information a keystroke can convey, the less time will be spent executing keystrokes. This is true of the scrolling and paging keys. The number of paging keystrokes executed was only a small percentage of the total number of keystrokes executed. However, it does indicate that subjects in this condition were at a small disadvantage when compared to non-window systems.

The second disadvantage of a window system is associated with problems of navigating through text. The finding that more subjects used paging keystrokes in the Low Complexity situation than in the other conditions indicates that these subjects were unsure if they had found the correct information. They paged up and down in order to view the whole document to ensure that they had not missed something. If the document was longer, and the subjects had to go farther in the text, it is very likely that this effect could slow performance drastically.

Individual Differences

It has long been known that some people find tasks on computers relatively easy to perform while others find tasks much more difficult to perform (Robertson, 1985). These individual differences are typically large and can increase within group error to such a large extent that true differences on systems are not found. The large individual differences also make it extremely difficult to design software and hardware systems because it is unclear what individual characteristics the designer should plan for. It is possible that the reason for not finding a significant interaction between Memory Load and Environment for task completion times is because individual differences were contributing to within group variability.

In the present study, individual differences were accounted for by blocking on subjects' verbal and spatial ability. According to Gomez, et al. (1986) spatial ability was significantly related to text editing performance using a line editor in which commands were issued by line number rather than moving a cursor on a screen. According to Gomez, et al. (1986) text editing involved spatial components related to searching text for information. They also found that the Nelson-Denny Reading Test was related to those portions of task performance which involved reading, such as reading a tutorial manual.

The results from Gomez, et al. (1986) were supported by findings from Vicente (1987). Subjects in the Vicente (1987) study performed tasks in which they were to locate a specific piece of information in a hierarchical file system using twelve search commands. For this task, significant correlations were found for both the verbal and spatial tests. However, the spatial test was found to relate to more dependent measures than the verbal test. This finding is consistent with the results of the present study. The two significant correlations in this study involved the spatial test.

Apparently, the tasks in the other two studies were more spatially and verbally oriented than the tasks employed in this study. It is also possible that the low relationship was due to a restriction of range of scores on the tests. This would have reduced the correlations and resulted in the non-significant blocking variable effect. As it stands, the tests which were used had so little correlation with the dependent measure that when they were used as a blocking variable in the ANOVA model, the effect was essentially to use degrees of freedom with little or no gain. For the tasks in the present study, it may have been better to use a test which evaluates memory capacity instead of the verbal and spatial tests. For example, a test of digit span may have been more useful in accounting for individual differences and reducing the within group variance. Or, if range restriction was the cause, to use subjects who have a wider range of scores on the verbal and spatial tests.

Predictions

Some predictions can be made from the findings of this study. First, when memory load is low, it should make very little difference what type of system is used. This was shown in this study in the form of task completion times, number of keystrokes executed and workload measures.

However, as memory load increases, the different non-window strategies should become less and less useful in terms of task completion times, number of keystrokes executed and workload measures.

As memory load begins to increase to about eight pieces of information, or about the capacity of working memory, task performance of operators using a window system should begin to outperform operators using a non-window memorization strategy. This should occur because working memory resources are limited and no memory aid is available. Findings from this study indicate that the differences will show themselves as increased workload ratings for the memorization strategy.

As memory load continues to increase, task performance for operators using the non-window strategy of switching between files should begin to deteriorate as compared to performance using windows. The deterioration should be caused by a combination of an increase in executing alphabetic and cursor keystrokes as well as increased workload due to executing the keystrokes and placing and rehearsing information in working memory.

Finally, as memory load increases even more, the non-window strategy of note taking should no longer be effective. This should occur in two instances. First, if a large number of notes must be taken in order to complete a task, the execution of these responses should attenuate task completion time. Second, if the information is of a form in which notes are difficult to take and a true representation of the data is needed, then performance using notes should be inferior to window systems.

The results from the Bury et al. (1985) study can now be explained in terms of the non-window system which was used. Subjects in that study were allowed to take notes. As notes are a relatively common response which can be executed quickly, and memory load was not extremely high, it is not surprising that significant results were not found. Results from this study indicate that not only should task completion times not differ for a window and notes system, but workload should be low in both systems. The results of the Silver (1985) study are also easily explained in the same way. Memory load was low and the number of responses the subjects were required to execute in the window condition were somewhat complicated for the subjects to perform.

Implications

It was proposed in this study that a system which reduces the amount of resources required for selecting and executing responses as well as reducing the amount of working memory resources will result in better performance than a system which does not, all else held constant. Given this, the findings from this study and the predictions just made allow us to make some suggestions for the design of window software. These suggestions will be made specifically in relation to the use of windows for working on search tasks such as the ones used in this experiment. However, some of the suggestions can be generalized to windows which are used for other purposes such as help windows.

First, a window system should be simple to use. The more difficult, in terms of commands or function keys which must be remembered and executed the less useful windows will be compared to non-window systems. The commands for using the window system should not be complicated to remember. If this occurs, then simply using the window system will be cognitively difficult.

Gaylin (1986), in a study designed to determine the most used functions that experienced computer operators used, found that the users spent most of their time moving between windows rather than creating, moving, or resizing windows. This indicates that, although the other functions should be available, the keystrokes required to move between windows should be kept simple and easy to remember. Also, as shown in this study, scrolling and paging keystrokes should be simple and quick to use. For instance, the programmable function keys could be used to open or move between windows, or a mouse may be used.

Second, along these same lines, the window system should use defaults to free the user from making decisions about the size and placement of the window. If a user is forced into making these decisions each time a window is opened, the usefulness of windows will decline. This was demonstrated in the Bury et al. (1985) study and also in an experiment by Bly and Rosenberg (1986). Task performance for tasks which varied in the complexity of the presentation of information was compared for two window systems. In the first window system the user had to change the size, shape and location of the window, whereas in the second one the system managed these activities. They found that overall, performance on the window system which was automatically managed was better than on the system which required user management.

Third, one of the major advantages of a window system is the ability to free the user from having to hold information in working memory. For tasks in which operators will need to locate a large amount of supplementary information, a window could be very beneficial. However, for the most part, it will not be worth the effort to design a window system for tasks in which the user's working memory capacity is not exceeded.

Fourth, windows allow the user to view the actual data. Therefore, a window system should be most needed and useful for information which cannot be easily reproduced in note form or held in working memory or when accuracy is of prime importance. When a task will take on these characteristics, and the user will be working in several files, it will be important to supply the user with a window system.

Fifth, windows should contain a status line and some method for helping the user to navigate through information. For example, the status line could indicate the total number of lines in a file as well as the current line number. This information can also be displayed graphically with a horizontal or vertical straight scale indicator with a moving pointer. This should enable users to orient themselves in the file and consequently should reduce task completion time. Also, windows should be of a large enough size so that the operator is able to view enough information to understand where he or she is in a file.

Sixth, it is possible that task performance using windows will decrease as the number of windows an operator uses increase. As more windows are used, the operator may be forced to spend more mental energy to sort out in what window needed information is in, more keystrokes will be executed moving between four or five windows, the operator will spend more time managing the size and placement of windows. This effect occurred in the Bury _et al._ (1985) study.

Conclusions

One of the original contentions of this study was that a system which could reduce the number and complexity of the response resources as well as reduce the amount of working memory resources required to complete a task would be better than a system which could not. This study has shown that a window system can be just such an efficient system by reducing workload and the number of responses executed over various non-window strategies.

The advantages of windows will not be evident until working memory capacity has been exceeded and even then, some non-window strategies can be more useful in terms of task completion time than a window system. A window system should only become advantageous when it results in a reduction in the number of responses and working memory resources required to complete a task over a non-window strategy. In the present study, the hypothesized interaction for task completion time between Memory Load and Environment did not occur, and is thought to be due to a failure to adequately load memory as well as a failure to account for individual differences on the tasks.

As with any computer system, the effectiveness with which an operator can use a window system interacts with even more than task parameters. The ease with which the system can be learned, the capabilities of the system, various personal characteristics such as memory capacity, and the help offered by the system will all determine task performance.

REFERENCES

Adams, J. (1980). Learning and memory. Homewood, IL: The Dorsey Press.

Alden, D., Daniels, R., & Kanarick, A. (1972). Keyboard design and operation: A review of the major issues. Human Factors, 14, 275-293.

Allport, D.A., Antonis, B. & Reynolds, P. (1972). On the division of attention: A disproof of the single channel hypothesis. Quarterly Journal of Experimental Psychology, 24, 255-265.

Baddeley, A. & Hitch, G. (1974). Working memory. In G. H. Bower (Ed.), The psychology of learning and motivation (Vol. 8). New York: Academic Press.

Berliner, C., Angell, D., & Shearer, J. W. (1964). Behaviors, measures and instruments for performance evaluation in simulated environments. Paper presented at The Symposium and Workshop on the Quantification of Human Performance, Albuquerque, NM.

Bly, S. & Rosenberg, J. (1986). A comparison of tiled and overlapping windows. In CHI 1986 Proceedings, 101-106. New York: ACM.

Borland Software (1985). Turbo editor toolbox: Version 1.0. Scotts Valley, CA: Borland International.

Broadbent, D. (1958). Perception and communication. London: Pergamon Press.

Bury, K., Boyle, J., Evey, R. & Neal, A. (1982). Windowing versus scrolling on a visual display terminal. Human Factors, 24, 385-394.

Bury, K. , Davies, S. & Darnell, M. (1985). Window management: A review of issues and some results from user testing. IBM Human Factors Technical Report HFC-53. San Jose, CA: Human Factors Center, General Products Division.

Casali, J. & Wierwille, W. (1983). A comparison of rating scale, secondary-task, physiological, and primary-task workload, estimation techniques in a simulated flight task emphasizing communications load. Human Factors, 25, 623-641.

Casali, J. & Wierwille, W. (1984). On the measurement of pilot perceptual workload: A comparison of assessment techniques addressing sensitivity and intrusion issues. Ergonomics, 27, 1033-1050.

Davies, S. , Bury, K. & Darnell, M. (1985). An experimental comparison of a windowed vs a non-windowed operating system environment. Proceedings of the Human Factors Society, 29th Annual Meeting, 250-254.

Davis, R. (1965). Expectancy and intermittency. Quarterly Journal of Experimental Psychology, ??, 75-78.

deGroot, A. (1966). Perception and memory versus thinking. In B. Kleinmuntz (Ed.) Problem solving. New York, NY: Wiley Publishing Co.

Ekstrom, R. , French, J. & Harmon, H. (1976). Manual for kit of factor-referenced cognitive tests. Princeton: ETS.

Embley, D. & Nagy, G. (1981). Behavioral aspects of text editors. Computing Surveys, 13, 33-70.

Gaylin, K. B. (1986). How are windows used? Some notes on creating an empirically based windowing benchmark task. In CHI 1986 Proceedings. 96-100. New York: ACM.

Gomez, L., Egan, D. & Bowers, C. (1986). Learning to use a text editor: Some learner characteristics that predict success. Human Computer Interaction, 2, 1-23.

Gopher, D., Brickner, M. & Navon, D. (1982). Different difficulty manipulations interact differently with task emphasis: Evidence for multiple resources. Journal of Experimental Psychology: Human Perception and Performance, 8, 146-157.

Hair, J., Anderson, R., Tatham, R. & Grablowsky, B. (1979). Multivariate Data Analysis With Readings. Tulsa, OK: The Petroleum Publishing Company.

Herdman, C. & Friedman, A. (1985). Multiple resources in divided attention: A cross-modal test of the independence of hemispheric resources. Journal of Experimental Psychology: Human Perception and Performance, 11, 40-49.

Hess, R. (1973). Nonadjectival rating scales in human response experiments. Human Factors, 15, 275-280.

Hicks, T. & Wierwille, W. (1979). Comparison of five mental workload assessment procedures in a moving-base driving simulator. Human Factors, 21, 129-143.

Howard, D. (1983). Cognitive psychology. New York, NY: Macmillan Publishing Company.

Isenee, S. & Bennett, C. (1983). The perception of flicker and glare on computer CRT displays. Human Factors, 25, 177-184.

Jeffries, R. (1985). On the threshold of supercheap RAM. PC, 4, (26), 92-94.

Kahneman, D. (1973). Attention and effort. Englewood Cliffs, NJ: Prentice-Hall.

Keele, S. W. (1973). Attention and human performance. Pacific Palisades, CA: Goodyear.

Keppel, G. (1982). Design & analysis: A researcher's handbook. Englewood Cliffs, NJ: Prentice-Hall, Inc.

Keppel, G. & Underwood, B. (1962). Proactive inhibition in short-term retention of single items. Journal of Verbal Learning and Verbal Behavior, 1, 153-161.

Kerr, B. (1973). Processing demands during mental operations. Memory and Cognition, 4, 401-412.

Kramer, A., Wickens, C. & Donchin, E. (1985). Processing of stimulus properties: Evidence for dual-task integrality. Journal of Experimental Psychology: Human Perception and Performance, 11, 393-408.

Lojoko, G. R. (1972). Typewriting techniques for the technical secretary. Englewood Cliffs, NJ: Prentice-Hall, Inc.

Melton, A. W. (1963). Implications of short-term memory for a general theory of memory. Journal of Verbal Learning and Verbal Behavior, 2, 1-21.

Miller, G. (1956). The magical number seven, plus or minus two: Some limits on our capacity for processing information. Psychological Review, 63, 81-97.

Moray, N. (1967). Where is attention limited? A survey and a model. Acta Psychologica, 27, 84-92.

Nelson-Denny Reading Test, Form C. (1973). Boston: Houghton Mifflin.

Norusis, M. J. (1986). Advanced statistics, SPSS/PC+. Chicago, IL: SPSS, Inc.

Ogden, G., Levine, J. & Eisner, E. (1979). Measurement of workload by secondary tasks. Human Factors, 21, 529-548.

Peterson, L. & Peterson, M. (1959). Short-term retention of individual verbal items. Journal of Experimental Psychology, 58, 193-198.

Reitman, J. (1974). Without surreptitious rehearsal, information in short-term memory decays. Journal of Verbal Learning and Verbal Behavior, 13, 365-377.

Robertson, I. (1985). Human information-processing strategies and style. Behavior and Information Technology, 4, 19-29.

Rumelhart, D. & Norman, D. (1982). Simulating a skilled typist: A study of skilled cognitive-motor performance. Cognitive Science, 6, 1-36.

Savage, R., Wierwille, W. & Cordes, R. (1978). Evaluating the sensitivity of various measures of operator workload using random digits as a secondary task. Human Factors, 20, 649-654.

Scapin, D. (1981). Computer commands in restricted natural language: Some aspects of memory and experience. Human Factors, 23, 365-375.

Seibel, R. (1972). Data entry devices and procedures. In H.S. Van Cott & R.G. Kinkade (Eds.), Human engineering guide to equipment design. Washington D.C.: U.S. Government Printing Office.

Shaffer, L. & Hardwick, J. (1970). The basis of transcription skill. Journal of Experimental Psychology, 84, 424-440.

Shulman, H. & Greenberg, S. (1971). Perceptual deficit due to division of attention between memory and perception. Journal of Experimental Psychology, 88, 171-176.

Silver, L. (1985). Displays, acquisition and performance. Proceedings of the Human Factors Society - 29th Annual Meeting, 1985 620-624.

Skipper, J., Reiger, C. & Wierwille, W. (1986). Evaluation of decision-tree rating scales for mental workload estimation. Ergonomics, 29, 585-599.

Spreen, O. & Schulz, R. (1966). Parameters of abstraction, meaningfulness, and pronounciability for 329 nouns. Journal of Verbal Learning and Verbal Behavior, 5, 459-468.

Starr, S. (1984). Effects of video display terminals in a business office. Human Factors, 26, 347-356.

Steinbrecher, D. (1984). Automation outlook: Are windows the answer to improved productivity or do they just make computers as chaotic as messy desktops?. Office Administration and Automation, p. 10.

Sternberg, S., Knoll, R., & Wright, C.E. (1977). Experiments on temporal aspects of keyboard entry. In J.P. Duncanson (Ed.), Getting it together: Research and applications in human factors. Santa Monica, CA: Human Factors.

Thorndike, E. & Lorge, I. (1944). The teachers word book of 30,000 words. New York: Teachers College, Columbia University.

Vicente, K. (1987). *Accommodating individual differences in searching a hierarchical file system.* Unpublished master's thesis, Virginia Polytechnic Institute and State University, Blacksburg, VA.

Wagenaar, W.A. (1969). Note on the construction of digram-balanced Latin squares. *Psychological Bulletin, 72,* 384-386.

Wickens, C. (1976). The effects of divided attention on information processing in manual tracking. *Journal of Experimental Psychology: Human Perception and Performance, 2,* 1-13.

Wickens, C. (1980). The structure of attentional resources. In R. Nickerson & R. Pew (Eds.), *Attention and performance VIII.* Hillsdale, NJ: Erlbaum Associates.

Wickens, C. (1984). *Engineering psychology and human performance.* Columbus, OH: Charles E. Merrill Publishing Company.

Wickens, C., Mountford, S. & Schreiner, W. (1981). Multiple resources, task-hemispheric integrity, and individual differences in time-sharing. *Human Factors, 23,* 211-229.

Wierwille, W. & Casali, J. (1983). A validated rating scale for global mental workload measurement applications. *Proceedings of the 27th Annual Meeting of the Human Factors Society*

Wierwille, W. & Conner, S. (1983). Evaluation of 20 workload measures using a psychomotor task in a moving-base aircraft simulator. *Human Factors, 25,* 1-16.

Wierwille, W., Rahimi, M. & Casali, J. (1985). Evaluation of 16 measures of mental workload using a simulated flight task emphasizing mediational activity. *Human Factors, 27,* 489-502.

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.

APPENDIX A

SUBJECT CONSENT FORM AND QUESTIONNAIRES

Informed Consent

This study involves comparing performance for two types of computer screens: one with a window and one without a window. You will be asked to perform some tasks in only one of these conditions. We are interested in how people perform in these environments and not on their ability to learn a program. The project will last a total of approximately two hours.

You will first be asked to complete two short tests (15 minutes total); a verbal and a spatial test. These are NOT tests of intelligence, they will be used only for the purpose of placing you in the conditions for the experiment. All scores on these tests will be kept strictly confidential.

You will then be asked to attend an experimental session which will last about 1 1/2 hours. You will perform a total of 14 simple editing tasks. You will be tutored in how to use the system and will perform several tasks in a tutorial program. Then you will perform several more tasks in an experimental condition. Your performance in both accuracy and time will be measured. After each task, you will be asked to rate the difficulty of the computer system you are using. We are not interested in individual performance, but rather how people, as a group, perform with using these two types of computer screens.

Your identity and any records of your performance will remain confidential. Your participation is voluntary and you may discontinue participation at any time with no penalty other than the loss of points for the experiment.

For more information, or if you have any questions, please call:
Timothy J. O'Keefe at 552-0355 or
A.M. Prestrude, Ph.D.
or see:
Steven J. Zaccaro, Ph.D. or
C. D. Waring, Ph.D. from the Human Subject's Committee.

Under the conditions set forth above, I ________________ voluntarily agree to participate in this study.

Signed ________________ Date ______________

I.D. # ________________ Phone ____________

Computer Knowledge Questionnaire

Name: ______________________ Age: _________ Date: _____/____/_____

Year in School (Circle one): Fr Sop Jr Sr Grad

Please answer the following questions as best you can:

1. Have you ever used a computer before: Yes No
2. If yes, what type (Circle all that apply): a) IBM PC (including the AT and compatibles) b) DEC Rainbow c) Apple Computers d) Tech's mainframe system e) don't know
3. Circle the computer programs which you have used: a) Xedit on the Mainframe b) Readiwriter c) Wordstar d) Word Perfect e) Microsoft Word f) Superwriter g) Script PC h) Einsteinwriter i) Volkswriter j) Word Proof k) Leading Edge Word Processor l) Other ________________
4. Have you ever used a program with windows? Windows consist of the ability to split the computer screen into at least two portions with different pieces of information in each.
   Yes No If yes, what program ? ______________________
5. How often have you used a personal computer in the last school year: a) less than 1 hour/month b) from 1 to 5 hours/month c) from 6 to 10 hours/month d) from 11 to 15 hours/month e) from 16 to 20 hours/month f) greater than 20 hours/month

The main emphasis of this study is on how people perform on two different software screen environments and not on their ability to learn a new piece of software. Thus, the software used in this study is made to be especially easy to use. This study involves comparing performance for two types of computer software screens: one with a window and the other without a window. By windows, I mean the computer screen can be split into two or more parts or windows. The information in these windows can be different. For instance, you may be working in a text file in one window, and a second window may contain information, such as notes, relevant to the text file you are working on. Some of you will be working in a Window environment, while others will be working in a Non-Window environment.

In the present study, you will first go through a Tutorial which will familiarize you with the software which you will use. You will then perform a total of 14 simple editing tasks which basically involve making some changes to several text files. The project will last approximately 1 hour and a half.

This preliminary part of the experiment will last about 20 minutes. You will first take a spatial test and then a verbal test. These are NOT tests of intelligence, they will be used only for the purpose of placing you in the conditions for the experiment. All scores on these tests will be kept strictly confidential.

Computer Experiment Questionnaire

Date: _____/____/_____

Name: _____________________ Sex: Male Female

Major: ____________________

Please answer the following questions as best you can:

1. How do you feel about using the IBM PC right now?

1-----------2-----------3-----------4-----------5

Anxious	A little uneasy	Neutral	Comfortable	Very Comfortable

2. How do you feel about this experiment right now?

1-----------2-----------3-----------4-----------5

Anxious	A little uneasy	Neutral	Comfortable	Very Comfortable

3. Rate your own note taking ability?

1-----------2-----------3-----------4-----------5

Below Average		Average		Above Average

4. How much time, in comparison to others, do you think it would take you to learn a simple word processing program?

1-----------2-----------3-----------4-----------5

Longer Than Average		Average		Less Than Average

5. Rate yourself as to your ability to understand and use new computer packages:

1-----------2-----------3-----------4-----------5

Poor		Average		Excellent

6. What is your motivation to participate in this experiment?

1-----------2-----------3-----------4-----------5

Unmotivated		Neutral		Highly Motivated

APPENDIX B

TUTORIAL INSTRUCTIONS

Window Instructions

Screen 1

This is considered to be WINDOW 1. The Status line at the top of the screen indicates what line and column the cursor is on, the name of the file in the window, and whether or not the Insert mode is on.

To access a file into this window, Press the F1 key. A question above the Status Line will ask what file you want to read into the window. Type the name of the file you want to edit and press the RETURN key. The cursor will remain BELOW the Status line when typing the name of this file. This process is demonstrated on the next screen.

Screen 2

In this screen, the F1 key was pressed. As you can see, the program asks for the name of the file you wish to edit. Here, the computer is typing in the name. Normally you would type the name in and the computer would access the file and place it in the window.

Once a file is read into WINDOW 1, you can use the PageUp and PageDown keys at the right of the keyboard to SCROLL through the file. The Arrow keys may also be used to move around in the file.

Screen 3 To open a SECOND WINDOW, press the F3 key. The last 10 lines of the screen will be covered over by WINDOW 2 and another status line. Again, you will be asked what file you want to read in to the window. Type the filename in as you did with WINDOW 1.

Screen 4

This is considered to be WINDOW 2. It was opened with the F3 Key. To access a file, type in the name of the file you want. You may again use the PageUp and PageDown keys as well as the Arrow keys to move around this file. To move to WINDOW 1, press the F9 key. You may then type information into WINDOW 1 while looking at WINDOW 2. To move back to WINDOW 2, press the F10 key. To close WINDOW 2, press the F4 Key.

Screen 5

Again, to move the cursor to Window 1 from Window 2 press the F9 key. To move the cursor to Window 2 press the F10 Key. The PAGEUP and DOWN keys as well as the ARROW keys may be used to move around the file. The INSERT key (located in the lower right of the keyboard) may also be used. If you try to open too many windows, you will get an error message located above the Window 1 Status Line. Clear the error statement with the ESCAPE key located in the upper left corner of the keyboard and begin again. When you are finished any task, make sure that you close Window 2 first with the F4 key and then close Window 1 with the F2 key. When you press the F2 key, you will be asked whether or not you want to save the file. You should answer with "Y". This question is asked in case you made a mistake and pressed F2 before you really wanted to.

Screen 6

Supplementary information which you will need to complete any task will be found in WINDOW 2. YOU MUST USE THE F9 AND F10 KEYS TO MOVE BETWEEN THE WINDOWS TO PLACE THIS INFORMATION INTO WINDOW 1. You will NOT be allowed to take notes in this condition and you will only be allowed to open WINDOW 2 ONCE for each task. A simple demonstration of a task will now be presented.

In this demonstration, two filenames which start with the letter "P" must be found in a file called DIRECT.DIR and typed into a file called JONES.DOC. JONES.DOC is the main file and DIRECT.DIR is the supplementary file. Therefore, JONES.DOC will be placed in WINDOW 1 and DIRECT.DIR will be placed in WINDOW 2. Spaces where information must be typed into the main file will be marked by Question Marks (???s). You will type over the question marks with the appropriate information.

Screen 7

This screen shows that the F1 Key was pressed. As you can see, the "Read File" message is displayed. You would type the name "JONES.DOC" and then press the return key in order to access this file. The filename "JONES.DOC" may be typed in either upper, lower or mixed case. It is typed in upper case here for emphasis.

Screen 8

4100 Jee Street
Springfield, MA 38921

Mr. Paul D. Jones
Jones Inc.

Mr. Jones:

Enclosed are the computer files which you requested.
They are: Spool.exe, Write.com, ?????.???, and
?????.???. I am sure they will solve your computer
problems.

Sincerely,

Sara Drayer

This screen shows the text for JONES.DOC. The ??s show the position and number of letters for the words which must be typed into JONES.DOC. The next step is to open WINDOW 2 and access the file DIRECT.DIR.

Screen 9

4100 Jee Street
Springfield, MA 38921

Mr. Paul D. Jones
Jones Inc.

Mr. Jones:

Enclosed are the computer files which you requested. They are: Spool.exe, Write.com, ?????.???, and ?????.???. I am sure they will solve your computer problems.

Sincerely,

Screen 10

In this screen, WINDOW 2 was opened with the F3 key. Notice the "File Read" message at the top of the screen. Just type in the name of the supplementary file "DIRECT.DIR" and after pressing the RETURN key, the file will be placed in WINDOW 2. Notice also, WINDOW 2 covers name of the author of the letter. The cursor will be located in WINDOW 2 when it is opened, but you can move the cursor to WINDOW 1 by pressing the F9 key.

Screen 11

As you can see, WINDOW 2 contains the file for DIRECT.DIR. The task required that we find the two files which begin with the letter P and type them into the letter where the ??s are. The two files are "Print.exe" and "Pascal.com". It is possible that you may have to scroll down the file to find the names which you will need. If you have to do this you may use the PageUp or PageDown Keys or the Arrow Keys. To move the cursor up to WINDOW 1, use the F9 Key. If you want to move the cursor back to WINDOW 2 press the F10 Key.

Screen 12

UTILITY FILES

WRITE.COM
PRINT.EXE
LOCATE.COM
DRAW.COM
PASCAL.COM
BASIC.EXE
FORMAT.EXE
TALK.COM

IMPORTANT: In this Window condition, you would move between WINDOWs 1 and 2, at this time, using the F9 and F10 keys to find and type the information needed. Thus, for example, after opening WINDOW 2, you would locate the names "Pascal.com" and "Print.exe" and use the F9 key to move to WINDOW 1 and type the information needed into the correct place. You do not need to commit the information to memory, as you can simply look at the information in WINDOW 2. Remember, however, you can only open WINDOW 2 once for any given task.

Screen 13

4100 Jee Street
Springfield, MA 38921

Mr. Paul D. Jones
Jones Inc.

Mr. Jones:

Enclosed are the computer files which you requested. They are: Spool.exe, Write.com, Print.exe, and Pas???.???. I am sure they will solve your computer problems.

Sincerely,

Screen 14

UTILITY FILES

WRITE.COM
PRINT.EXE
LOCATE.COM
DRAW.COM
PASCAL.COM

*** Notice that the two files are almost typed in where the XXs were. To finish the task, close WINDOW 2 with the F4 Key and then close WINDOW 1 with the F2 Key.

Screen 15

Now you will perform five tasks. The first one will walk you through the word processor and is designed to make you familiar with the system. After these five tasks, you will perform three timed tasks to make sure you understand how to use the word processor. Finally, you will perform six timed tasks for the experiment. If you have any questions, please ask an experimenter. Below is a summary of the F Keys.

Non-Window Instructions

Screen 1

This is considered to be WINDOW 1. The Status line at the top of the screen indicates what line and column the cursor is on, the name of the file in the window, and whether or not the Insert mode is on.

To access a file into this window, Press the F1 key. A question above the Status Line will ask what file you want to read into the window. Type the name of the file you want to edit and press the RETURN key. The cursor will remain BELOW the Status line when typing the name of this file. This process is demonstrated on the next screen.

Screen 2

In this screen, the F1 key was pressed. As you can see, the program asks for the name of the file you wish to edit. Here, the computer is typing in the name. Normally you would type the name in and the computer would access the file and place it in the window.

Once a file is read into WINDOW 1, you can use the PageUp and PageDown keys at the right of the keyboard to SCROLL through the file. The Arrow keys may also be used to move around in the file.

Screen 3

To open a SECOND WINDOW, press the F3 key. WINDOW 2 will completely cover WINDOW 1 and give you another status line. Again, you will be asked what file you want to read in to the window. Type the filename in as you did with WINDOW 1.

Screen 4

This is considered to be WINDOW 2. It was opened with the F3 Key. As you can see, it covers all of WINDOW 1. To access a file, type in the name of the file you want. You may again use the PageUp and PageDown keys as well as the Arrow keys to move around this file. To close WINDOW 2, press the F4 Key. When you close this window, WINDOW 1 will be uncovered.

Screen 5

Again, to open WINDOW 2, press the F3 key and to close this window, press the F4 key. The PAGEUP and DOWN keys as well as the ARROW keys may be used to move around the file. The INSERT key (located in the lower right of the keyboard) may also be used. If you try to open too many windows, you will get an error message located above the Window 1 Status Line. Clear the error statement with the ESCAPE key located in the upper left corner of the keyboard and begin again. When you are finished any task, make sure that you close Window 2 first with the F4 key and then close Window 1 with the F2 key. When you press the F2 key, you will be asked whether or not you want to save the file. You should answer with "Y". This question is asked in case you made a mistake and pressed F2 before you really wanted to.

Screen 6 - Note Condition

Supplementary information which you will need to complete any task will be found in WINDOW 2. You will ONLY BE ALLOWED TO OPEN WINDOW 2 ONCE for any task. YOU MAY TAKE NOTES on the information in WINDOW 2 in order to complete the task. A simple demonstration of a task will now be presented.

In this demonstration, two filenames which start with the letter "P" must be found in a file called DIRECT.DIR and typed into a file called JONES.DOC. JONES.DOC is the main file and DIRECT.DIR is the supplementary file. Therefore, JONES.DOC will be placed in WINDOW 1 and DIRECT.DIR will be placed in WINDOW 2. Spaces where information must be typed into the main file will be marked by Question Marks (???s). You will type over the question marks with the appropriate information.

Screen 6 - Switch Condition

Supplementary information which you will need to complete any task will be found in WINDOW 2. You will BE ALLOWED TO OPEN WINDOW 2 AS MANY TIMES AS YOU WISH for any task. YOU MAY NOT TAKE ANY NOTES. In order to complete the tasks, you do not need to memorize all of the information you will need at one time. Rather, you may have to open and close WINDOW 2 several times obtaining information from WINDOW 2. A simple demonstration of a task will now be presented.

In this demonstration, two filenames which start with the letter "P" must be found in a file called DIRECT.DIR and typed into a file called JONES.DOC. JONES.DOC is the main file and DIRECT.DIR is the supplementary file. Therefore, JONES.DOC will be placed in WINDOW 1 and DIRECT.DIR will be placed in WINDOW 2. Spaces where information must be typed into the main file will be marked by Question Marks (???s). You will type over the question marks with the appropriate information.

Screen 6 - Memorization Condition

Supplementary information which you will need to complete any task will be found in WINDOW 2. You will ONLY BE ALLOWED TO OPEN WINDOW 2 ONCE for any task. YOU MAY NOT ANY TAKE NOTES. In order to complete the tasks, you MUST commit the information which you will need from WINDOW 2 to memory. A simple demonstration of a task will now be presented.

In this demonstration, two filenames which start with the letter "P" must be found in a file called DIRECT.DIR and typed into a file called JONES.DOC. JONES.DOC is the main file and DIRECT.DIR is the supplementary file. Therefore, JONES.DOC will be placed in WINDOW 1 and DIRECT.DIR will be placed in WINDOW 2. Spaces where information must be typed into the main file will be marked by Question Marks (???s). You will type over the question marks with the appropriate information.

Screen 7

This screen shows that the F1 Key was pressed. As you can see, the "Read File" message is displayed. You would type the name "JONES.DOC" and then press the return key in order to access this file. The filename "JONES.DOC" may be typed in either upper, lower or mixed case. It is typed in upper case here for emphasis.

Screen 8

4100 Jee Street
Springfield, MA 38921

Mr. Paul D. Jones
Jones Inc.

Mr. Jones:

Enclosed are the computer files which you requested.
They are: Spool.exe, Write.com, ?????.???, and
?????.???. I am sure they will solve your computer
problems.

Sincerely,

Sara Drayer

This screen shows the text for JONES.DOC. The ??s show the position and number of letters for the words which must be typed into JONES.DOC. The next step is to open WINDOW 2 and access the file DIRECT.DIR.

Screen 9

In this screen, WINDOW 2 was opened with the F3 key. As you can see, this window covers WINDOW 1. Notice the "File Read" message at the top of the screen. Just type in the name of the supplementary file "DIRECT.DIR" and after pressing the RETURN key, the file will be placed in WINDOW 2.

Screen 10

UTILITY FILES

WRITE.COM
PRINT.EXE
LOCATE.COM
DRAW.COM
PASCAL.COM
BASIC.EXE
FORMAT.EXE
TALK.COM

As you can see, WINDOW 2 contains the file for DIRECT.DIR. The task required that we find the two files which begin with the letter P and type them into the letter where the ??s are. The two files are "Print.exe" and "Pascal.com". It is possible that you may have to scroll down the file to find the names which you will need. If you have to do this you may use the PageUp or PageDown Keys or the Arrow Keys.

Screen 11 - Notes Condition

UTILITY FILES

WRITE.COM
PRINT.EXE
LOCATE.COM
DRAW.COM
PASCAL.COM
BASIC.EXE
FORMAT.EXE
TALK.COM

IMPORTANT: In this Non-Window condition, you would make some notes, at this time, on the piece note paper given to you, as to the names and correct spelling of the files. Thus, for the example, on your note paper, you might write down the names "Pascal.com" and "Print.exe". Remember, once you close WINDOW 2 for this task, you will not be able to open it again. Therefore, it is imperative that accurately write the information down which you will need. After you have made your notes, close WINDOW 2 with the F4 Key and WINDOW 1 will again become visible.

Screen 11 - Switch Condition

UTILITY FILES

WRITE.COM
PRINT.EXE
LOCATE.COM
DRAW.COM
PASCAL.COM
BASIC.EXE
FORMAT.EXE
TALK.COM

IMPORTANT: In this Non-Window condition, you would find the information which you need from WINDOW 2 and remembering what you can, close WINDOW 2 with the F4 key and type the information into WINDOW 1. If you are unsure of spelling or can not remember all of the information, simply open WINDOW 2 again and obtain more information. Thus, for the example, you might try to remember the name "Pascal.com" , close WINDOW 2, type it into WINDOW 1, and then open WINDOW 2 again to obtain the name "Print.exe" to type into WINDOW 1. Remember, you can NOT take any notes.

Screen 11 - Memorization Condition

UTILITY FILES

WRITE.COM
PRINT.EXE
LOCATE.COM
DRAW.COM
PASCAL.COM
BASIC.EXE
FORMAT.EXE
TALK.COM

IMPORTANT: In this Non-Window condition, you would find the information which you will need from WINDOW 2 and memorize this information. Thus, for the example, you would have to memorize the correct spelling of the names "Pascal.com" and "Print.exe". Remember, once you close WINDOW 2 for this task, you will not be able to open it again. Therefore, it is imperative that accurately memorize the information which you will need. After you have memorized the information, close WINDOW 2 with the F4 Key and Window 1 will again be visible.

Screen 12

4100 Jee Street
Springfield, MA 38921

Mr. Paul D. Jones
Jones Inc.

Mr. Jones:

Enclosed are the computer files which you requested. They are: Spool.exe, Write.com, Print.exe, and Pas???.???. I am sure they will solve your computer problems.

Sincerely,

*** Notice that the two files are almost typed in where the ??s were. To finish the task close WINDOW 1 with the F2 Key.

Screen 13

Now you will perform five tasks. The first one will walk you through the word processor and is designed to make you familiar with the system. After these five tasks, you will perform three timed tasks to make sure you understand how to use the word processor. Finally, you will perform six timed tasks for the experiment. If you have any questions, please ask an experimenter. Below is a summary of the F Keys.

APPENDIX C

INITIAL WRITTEN INSTRUCTIONS

WELCOME TO TEXT EDITOR EXPERIMENT

Window Environment

This experiment is concerned with how people work with different types of computer environments in a word processing editor. This editor was designed specifically for the experiment. It is similar in operation to the Wordstar word processor and is relatively simple to use. You will only need a small number of commands which will be taught to you.

In this study, you will be asked to perform simple editing tasks such as finding a name in a list and typing it into a letter. This instruction sheet will give you general directions as to what is required. After you read this sheet, you will perform a 4 minute typing test to get an indication of how fast you type. Then you will go through a computer tutorial program which will give you specific details as to how to operate the word processor and complete the experiment. You will perform 5 tasks in the Tutorial to get used to the editor. Then you will perform three criteria tasks to see if you have learned how to use this editor, and finally, you will perform six tasks for the experiment.

Instructions for each of the tasks which you will perform are printed on the index cards in front of you. In order to complete the tasks, you will need to access information from two different files - a Main File such as a letter which you are editing, and a Supplementary File such as a phone list which contains information which you will need to type into the Main File. You will be working in a Windowed environment which means that the computer screen can be split into two areas which can contain different pieces of information. Thus, you can have the Main File in one window and the Supplementary File in another window. Initially, you will access a Main File in the editor. It will be placed in WINDOW 1. In order to access the Supplementary File, you will open a second window (WINDOW 2) which will cover the last 10 lines of the screen. In this way, you will be able to see and use the information in both the Main File and the Supplementary File at the same time. Throughout this document and the Tutorial Program, I will use the terminology of WINDOW 1 to refer to the Main File and WINDOW 2 to refer to the supplementary file.

The F keys (located to the left of the keyboard) control the opening, closing and moving between windows. The template over the F keys will help you remember their functions. In this study, you will always access main text files (such as a letter) in WINDOW 1. WINDOW 1 is opened by pressing the F1 key located to the left of the keyboard. In WINDOW 2, you will access other files which will contain lists of information (such as a phone list) needed to complete your assigned task. WINDOW 2 is opened by pressing the F3 key. F9 moves the cursor to WINDOW 1 and F10 moves the cursor to WINDOW 2. When you have completed your task, WINDOW 2 is closed with the F4 key and WINDOW 1 is closed with the F2 key. The tutorial will go over these instructions more carefully.

The word processor which you will be using will only allow you to open WINDOW 2 once for each task. You will also NOT be allowed to take notes.

In order to effectively accomplish the task given these constraints, you should first read the index instruction cards carefully before beginning any task. This is so that your reading time is not included in task time. Then, you should open WINDOW 1 and finally WINDOW 2. When both windows are open, you will be able to move the cursor back and forth between the two windows. In this way you can page through the information in WINDOW 2, move the cursor up to WINDOW 1 and type the information in. Some of the tasks will require only one or two words to be typed into WINDOW 1 while other tasks will require more. It is IMPERATIVE that you follow the above strategy as best you can.

NOTE: You are allowed to have, and will only need, a maximum of 2 windows opened for any task. Remember, you can only open WINDOW 2 once for each task and you are NOT allowed to take notes.

The TUTORIAL program will explain more about the specifics of the word processor and then will take you through five untimed tasks. After you have finished these tasks, you will perform three more tasks to make sure you have learned how to use this editor, and finally, you will perform six tasks for the experiment.

WELCOME TO TEXT EDITOR EXPERIMENT

Non-Window 1 Environment - Notes

This experiment is concerned with how people work with different types of computer environments in a word processing editor. This editor was designed specifically for the experiment. It is similar in operation to the Wordstar word processor and is relatively simple to use. You will only need a small number of commands which will be taught to you.

In this study, you will be asked to perform simple editing tasks such as finding a name in a list and typing it in a letter. This instruction sheet will give you general directions as to what is required. After you read this sheet, you will perform a 4 minute typing test to get an indication of how fast you type. Then you will go through a computer tutorial program which will give you specific details as to how to operate the word processor and complete the experiment. You will perform 5 tasks in the Tutorial to get used to the editor. Then you will perform three criteria tasks to see if you have learned how to use this editor, and finally, you will perform six tasks for the experiment.

Instructions for each of the tasks which you will perform are printed on the index cards in front of you. In order to complete the tasks, you will need to access information from two different files - a Main File such as a letter which you are editing, and a Supplementary File such as a phone list which contains information which you will need to type into the Main File. You will be working in a Non-Windowed environment which means that you can only look at one piece of information (one of these files) at a time. Initially, you will access a Main File in the editor. It will be placed in WINDOW 1. In order to access the Supplementary File, you will open a second window (WINDOW 2) which will cover the entire screen blocking your view of WINDOW 1. When you close WINDOW 2, WINDOW 1 will become visible again. In this way, you will be able to see only one of the files at a time (either the Main File or the Supplementary file). Throughout this document and the Tutorial Program, I will use the terminology of WINDOW 1 to refer to the Main File and WINDOW 2 to refer to the supplementary file.

The F keys (located to the left of the keyboard) control the opening and closing of windows. The template over the F keys will help you remember their functions. In this study, you will always access main text files (such as a letter) in WINDOW 1. WINDOW 1 is opened by pressing the F1 key located to the left of the keyboard. In WINDOW 2, you will access other files which will contain lists of information (such as a phone list) needed to complete your assigned task. WINDOW 2 is opened by pressing the F3 key. When you have completed your task, WINDOW 2 is closed with the F4 key and WINDOW 1 is closed with the F2 key. The tutorial will go over these instructions more carefully.

The word processor which you will be using will only allow you to open WINDOW 2 once for each task. You will, however, be allowed to take notes on any information in WINDOW 2 which you will need. In order to effectively accomplish the tasks given these constraints, it is important that you first read the index instruction cards carefully before beginning any task. This is so that your reading time is not included in task time. Then you should open WINDOW 1 and examine the main file to determine what type of information and where the information will be placed before opening WINDOW 2. It is important to examine the Main File before opening WINDOW 2. Otherwise you will end up writing down a lot of extraneous information. Some of the tasks will require only one or two words to be typed into WINDOW 1 while other tasks will require more. It is IMPERATIVE that you follow the above strategy as best you can.

NOTE: You are allowed to have, and will only need, a maximum of 2 windows opened for any task. Remember, you can only open open WINDOW 2 once for each task and you MAY take notes.

The TUTORIAL program will explain more about the specifics of the word processor and then will take you through five untimed tasks. After you have finished these tasks, you will perform three more tasks to make sure you have learned how to use this editor, and finally, you will perform six tasks for the experiment.

WELCOME TO TEXT EDITOR EXPERIMENT

Non-Window 2 Environment - Switch

This experiment is concerned with how people work with different types of computer environments in a word processing editor. This editor was designed specifically for this experiment. It is similar in operation to the Wordstar word processor and is relatively simple to use. You will only need a small number of commands which will be taught to you. In this study, you will be asked to perform simple editing tasks such as finding a name in a list and typing it in a letter. This instruction sheet will give you general directions as to what is required. After you read this sheet, you will perform a 4 minute typing test to get an indication of how fast you type. Then you will go through a computer tutorial program which will give you specific details as to how to operate the word processor and complete the experiment. You will perform 5 tasks in the Tutorial to get used to the editor. Then you will perform three criteria tasks to see if you have learned how to use this editor, and finally, you will perform six tasks for the experiment.

Instructions for each of the tasks which you will perform are printed on the index cards in front of you. In order to complete the tasks, you will need to access information from two different files - a Main File such as a letter which you are editing, and a Supplementary File such as a phone list which contains information which you will need to type into the Main File. You will be working in a Non-Windowed environment which means that you can only look at one piece of information (one of these files) at a time. Initially, you will access a Main File in the editor. It will be placed in WINDOW 1. In order to access the Supplementary File, you will open a second window (WINDOW 2) which will cover the entire screen blocking your view of WINDOW 1. When you close WINDOW 2, WINDOW 1 will become visible again. In this way, you will be able to see only one of the files at a time (either the Main File or the Supplementary file). Throughout this document and the Tutorial Program, I will use the terminology of WINDOW 1 to refer to the Main File and WINDOW 2 to refer to the supplementary file.

The F keys (located to the left of the keyboard) control the opening and closing of windows. The template over the F keys will help you remember their functions. In this study, you will always access main text files (such as a letter) in WINDOW 1. WINDOW 1 is opened by pressing the F1 key located to the left of the keyboard. In WINDOW 2, you will access other files which will contain lists of information (such as a phone list) needed to complete your assigned task. WINDOW 2 is opened by pressing the F3 key. When you have completed your task, WINDOW 2 is closed with the F4 key and WINDOW 1 is closed with the F2 key. The tutorial will go over these instructions more carefully.

The word processor which you will be using will allow you to open WINDOW 2 as many times as you wish. When you close WINDOW 2, WINDOW 1 will again be visible. You will NOT, however, be allowed to take any notes. In order to effectively accomplish the task given these constraints, it is important that you first read the index instruction cards carefully before beginning any task. This is so that your reading time is not included in task time. Then you should open WINDOW 1 and examine the main file to determine what type of information and where the information will be placed before opening WINDOW 2. Since you can open WINDOW 2 as many times as you wish, do not worry about memorizing all of the information at one time. After finding some information, you will close WINDOW 2, type the information in WINDOW 1 and again open WINDOW 2 if you need to obtain more information to be typed into WINDOW 1. Some of the tasks will require only one or two words to be typed into WINDOW 1 while other tasks will require more. It is IMPERATIVE that you follow the above strategy as best you can.

NOTE: You are allowed to have, and will only need, a maximum of 2 windows opened for any task. Remember, you may open WINDOW 2 as many times as you wish, but you may NOT take notes.

The TUTORIAL program will explain more about the specifics of the word processor and then will take you through five untimed tasks. After you have finished these tasks, you will perform three more tasks to make sure you have learned how to use this editor, and finally, you will perform six tasks for the experiment.

WELCOME TO TEXT EDITOR EXPERIMENT

Non-Window 3 Environment - Memorize

This experiment is concerned with how people work with different types of computer environments in a word processing editor. This editor was designed specifically for this experiment. It is similar in operation to the Wordstar word processor and is relatively simple to use. You will only need a small number of commands which will be taught to you. In this study, you will be asked to perform simple editing tasks such as finding a name in a list and typing it in a letter. This instruction sheet will give you general directions as to what is required. After you read this sheet, you will perform a 4 minute typing test to get an indication of how fast you type. Then you will go through a computer tutorial program which will give you specific details as to how to operate the word processor and complete the experiment. You will perform 5 tasks in the Tutorial to get used to the editor. Then you will perform three criteria tasks to see if you have learned how to use this editor, and finally, you will perform six tasks for the experiment.

Instructions for each of the tasks which you will perform are printed on the index cards in front of you. In order to complete the tasks, you will need to access information from two different files - a Main File such as a letter which you are editing, and a Supplementary File such as a phone list which contains information which you will need to type into the Main File. You will be working in a Non-Windowed environment which means that you can only look at one piece of information (one of these files) at a time. Initially, you will access a Main File in the editor. It will be placed in WINDOW 1. In order to access the Supplementary File, you will open a second window (WINDOW 2) which will cover the entire screen blocking your view of WINDOW 1. When you close WINDOW 2, WINDOW 1 will become visible again. In this way, you will be able to see only one of the files at a time (either the Main File or the Supplementary file). Throughout this document and the Tutorial Program, I will use the terminology of WINDOW 1 to refer to the Main File and WINDOW 2 to refer to the supplementary file.

The F keys (located to the left of the keyboard) control the opening and closing of windows. The template over the F keys will help you remember their functions. In this study, you will always access main text files (such as a letter) in WINDOW 1. WINDOW 1 is opened by pressing the F1 key located to the left of the keyboard. In WINDOW 2, you will access other files which will contain lists of information (such as a phone list) needed to complete your assigned task. WINDOW 2 is opened by pressing the F3 key. When you have completed your task, WINDOW 2 is closed with the F4 key and WINDOW 1 is closed with the F2 key. The tutorial will go over these instructions more carefully.

The word processor which you will be using will only allow you to open WINDOW 2 once for each task. You will also NOT be allowed to take notes on any information. In order to effectively accomplish the task given these constraints, it is important that you first read the index instruction cards carefully before beginning any task. This is so that your reading time is not included in task time. Then you should open WINDOW 1 and examine the main file to determine what type of information and where the information will be placed before opening WINDOW 2. Otherwise, you will not know precisely what you will need to remember. Since you will not be allowed to take notes and you will only be able to open WINDOW 2 once, you must commit the information which you will need to memory. You must try your best to memorize the information which you will need. Some of the tasks will require only one or two words to be typed into WINDOW 1 while other tasks will require more. It is IMPERATIVE that you follow the above strategy as best you can.

NOTE: You are allowed to have, and will only need, a maximum of 2 windows opened for any task. Remember, you can only open WINDOW 2 once for each task and you can NOT take notes.

The TUTORIAL program will explain more about the specifics of the word processor and then will take you through five untimed tasks. After you have finished these tasks, you will perform three more tasks to make sure you have learned how to use this editor, and finally, you will perform six tasks for the experiment.

APPENDIX D

EXPERIMENTAL TASK INSTRUCTIONS AND TEXT

HIGH MEMORY LOAD, HIGH COMPLEXITY

Instructions - HM, HC

On your disk there is a file named TASKE1.TXT. This file contains a letter to a psychologist with a list of 8 words which have been found to have a high "meaningfulness" rating. You can not remember the words and must refer to TASKE1.DAT, which is a file containing the words. You must find the 8 words which have a "meaningfulness" rating of 5.0 or higher in TASKE1.DAT and type them into the letter in TASKE1.TXT where the ???'s are.

Summary:

1. Open WINDOW 1 and read in TASKE1.TXT.
2. Open WINDOW 2 and read in TASKE1.DAT.
3. Replace the ??'s on line 17 of TASKE1.TXT with the 8 words that have a "meaningfulness" rating of 5.0 or higher found in TASKE1.DAT.
4. Close TASKE1.TXT.

Text - HM, HC

Department of Psychology
Virginia Tech
Blacksburg, VA 22314
February 18, 1987

E. Bruce Walker, Ph.D.
Department of Psychology
University of Richmond
Richmond, VA 23173

Dr. Walker:

As per your request of January 20th, enclosed are the top eight words from our study on the meaningfulness of words. I hope they serve their purpose well.

Words: ???, ???, ???, ???, ???, ???, ???, ???

Sincerely,

Data - HM, HC

MeaningFulness of Words

	Word	Meaningfulness	Specificity
1)	aid	7.66	3.04
2)	bee	8.33	4.28
3)	cap	8.55	5.04
4)	fur	9.47	4.96
5)	band	4.67	5.35
6)	key	8.77	5.70
7)	sand	4.22	6.52
8)	scale	4.39	3.78
9)	sport	3.89	7.22
10)	sum	7.88	5.22
11)	sword	4.61	2.33
12)	task	2.18	9.55
13)	teeth	3.67	8.18
14)	tone	3.85	3.33
15)	cat	9.10	6.48
16)	gas	8.98	5.04

MEDIUM MEMORY LOAD, HIGH COMPLEXITY

Instructions - MM, HC

There is a file on your disk called TASKE5.TXT. This file contains a letter from a heating contractor to a client who needs a heater fixed. In the letter, you want to list the four parts which will be needed to fix the heater. A parts list is contained in the file TASKE5.DAT. On line 16, of the letter, you need to type in the names of 4 parts found in the file TASKE5.DAT. The parts are found in the file TASKE5.DAT in 2 places under the name of Carpet Cleaning World. Two parts are found under Job 2 and 2 parts are found under Job 6. You must find all 4 parts and type them into TASKE5.TXT.

Summary:

1. Open WINDOW 1 and read in TASKE5.TXT.
2. Open WINDOW 2 and read in TASKE5.DAT.
3. Replace the ??'s on line 16 of TASKE5.TXT with the 4 parts for Carpet Cleaning World found in TASKE5.DAT under Jobs 2 and 6.
4. Close TASKE5.TXT.

Text - MM, HC

Chapman Heating & Cooling
Wilderness Road
St Louis, MO 26893

John Fosner
Carpet Cleaning World
4399 Jackson St.
St. Louis, MO 26875

Mr. Fosner:

As per your request of January 19, 1987, we have found a list of the parts which will be needed to repair your business. These are:

????, ?????, ?????, ??????

The total cost will be $85.00. If you have any questions, please give me a call.

Thank You,
Ken Laser

Data - MM, HC

List of Parts for January Jobs

Job	Parts
1) Norwest	Compressor, Hose, Fan Freon
2) Carpet Cleaning World	Pipe, Motor Continued as Job #6..
3) Jerry's Gas	T valve, Pressure Gauge, Copper tubing
4) Pizza Hut	Compressor, Terminator, Freon, seals
5) Jack Hall	Rewire heating Circuit
6) Carpet Cleaning	Metal, Bottle

LOW MEMORY LOAD, HIGH COMPLEXITY

Instructions - LM, HC

There is a file on your floppy disk called TASKE6.TXT. In this letter you wish to provide someone with information about an upcoming conference. Since you can not remember all the details about the conference, you must refer to another source to get them. Fortunately, there is a file on your floppy disk called TASKE6.DAT which is a memo that contains information about the conference. Find the date of the conference and the last date for submitting papers in the file TASKE6.DAT and type this information into TASKE6.TXT where the ???'s are on line 13.

Summary:

1. Open WINDOW 1 and read in TASKE6.TXT.
2. Open WINDOW 2 and read in TASKE6.DAT.
3. Replace the ??'s in TASKE6.TXT on line 13 with the date of the conference and the last date to submit papers found in TASKE6.DAT.
4. Close TASKE6.TXT.

Text - LM, HC

2840 Lee Street
St. Cloud, Minnesota 56301

Mr. Paul D. Nelson
Vice President, Public Relations
Televideo Studios

Mr. Nelson:

This is to inform you of the upcoming National Advertising
Media Conference of the American Advertising Association. The
conference will be held in Minneapolis, Minn. on
????????? 8. ???????? 1, 1987 is the last date to submit
papers. You will be staying
at the Minneapolis Hilton Hotel (Phone 612 783-3361).

We were very pleased to hear that you accepted our
invitation to speak at this important event for the
Association. My colleagues and I are looking forward to
seeing you again we wish you luck with your Cameron Project.

Sincerely,

John J. Karr
Coordinator, National

Data - LM, HC

National Advertising Media Conference
Information

Date of Conference: September 8, 1987

Place: Minneapolis Hilton Hotel

Location: 1432 Stinson Blvd.
Minneapolis, Minnesota

Phone: (612) 783-3361

Coordinator: John J. Karr (612) 321-8921

Scheduled Speakers:

Jack D. Fallway
Mary Confroy

HIGH MEMORY LOAD, LOW COMPLEXITY

Instructions - HM, LC

On your disk there is a file named TASKE2.TXT. This file contains a letter from a store owner ordering items from a supply company. You can not remember what it is you need to order, but you have them listed in the file, TASKE2.DAT. Find the 8 items with the code "08" which need to be ordered from the file TASKE2.DAT and type them on line 14 of TASKE2.TXT where where the ???'s are.

Summary:

1. Open WINDOW 1 and read in TASKE2.TXT.
2. Open WINDOW 2 and read in TASKE2.DAT.
3. Replace the ??'s on line 14 of TASKE2.TXT with the 8 items found in TASKE1.DAT with code "08" which need to be ordered.
4. Close TASKE2.TXT.

Text - HM, LC

General Store, Inc.
1722 Stalway Ave
Columbus, GA 43991
February 17, 1987

Mr. T. K. Jones
Gadway Grocery
1212 Custus Drive
Columbus, GA 43991

Mr. Jones:

Please enter our order for your line of the following products: ???, ???, ???, ????, ????, ????, ???, ???. We believe that your products are the best buy in town. Thank you.

Sincerely,

Data - HM, LC

Items Needed 2/1/87

Item	Code
Sugar	09
Tea	08
Pie	08
Pan	08
Corn	08
Flag	08
Wine	08
Gun	08
Map	08
Towels	10
Cups	10
Mugs	10
Coasters	11
Plates	11
Candles	12
Maps	12

MEDIUM MEMORY LOAD, LOW COMPLEXITY

Instructions

On your disk there is a file named TASKE3.TXT. This is a shopping list which you are writing so that you can make a recipe. You can not remember the 4 ingredients which you will need, so you must refer to the recipe found in TASKE3.DAT. Find the 4 ingredients from the recipe found in TASKE3.DAT and type them into the file TASKE3.TXT on line 4.

Summary:

1. Open WINDOW 1 and read in TASKE3.TXT.
2. Open WINDOW 2 and read in TASKE3.DAT.
3. Replace the ??'s on line 4 of TASKE3.TXT with the names of the 4 ingredients found in TASKE3.DAT.
4. Close TASKE3.TXT.

Text - MM, LC

```
               Shopping List for Recipe:
```

Data - MM, LC

```
                               Apple Pie

         Ingredients:
                6 cups Apples
                2 tablespoons Flour
                1 cup Cream
                7 slices Bread
```

LOW MEMORY LOAD, LOW COMPLEXITY

Instructions - LM, LC

There is a file on your disk called TASKE4.TXT. This is a memo from a school placement service to graduating seniors about possible job openings. You need to find the name of 1 company from a list in a file called TASKE4.DAT and place it on line 22 of the memo. The company name which you will need is the one hiring on June 15.

Summary:

1. Open WINDOW 1 and read in TASKE4.TXT.
2. Open WINDOW 2 and read in TASKE4.DAT.
3. Replace the ??'s on line 22 in TASKE4.TXT with the name of 1 company which is hiring on June 15 found in TASKE4.DAT.
4. Close TASKE4.TXT.

Text - LM, LC

Memo

To: Graduating Seniors
From: Placement Service
Subject: Companies Hiring This Spring
Date: January 30, 1987

For those seniors who are graduating this spring, we have compiled a list of all the companies in the Washington - Northern Virginia area which are hiring between June and July, 1987. Those of you who are interested should contact Suzie Jacobs of the Placement Service to set up an interview. Only those students who sign up by April 30th will be able to be interviewed. Sign up will take place in Henry Hall room 234. There will be a sign up sheet posted on the wall.

The Companies are:

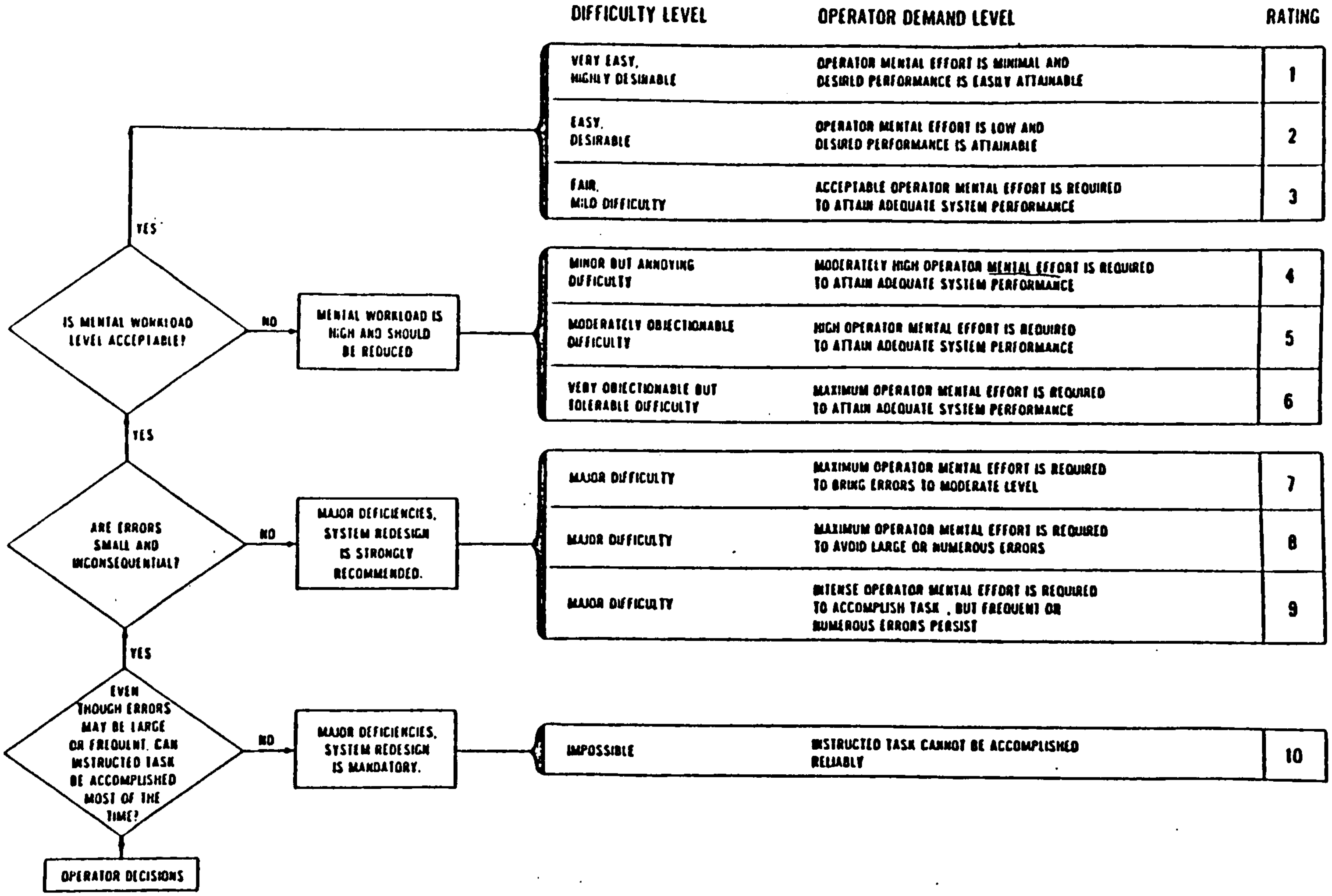
Jos. A. Bank Clothiers Inc.
Hahn Shoe Stores
7-Eleven Food Stores
?????????? ?????????

Data - LM, LC

Company	Date Hiring
Sears, Roebuck & Co.	Sept 30
Kay Jewelers Inc.	Sept 30
Dart-Kraft Inc.	Oct 10
Control Data Corp.	Oct 15
NCR Corporation	Oct 30
Geico Insurance	Nov 2
Hechinger Company	Nov 5
Production Foundation	June 15
Waxie Maxie	Dec 1
Price Waterhouse	Dec 5
Zayre Corporation	Jan 19
Saks Fifth Avenue	Jan 30
Camp Mfg Co.	May 5
Lord & Taylor	May 8
Union Camp Inc	Aug 10
Dryden Oil Co.	Aug 15

APPENDIX E

MODIFIED COOPER-HARPER RATING SCALE AND INSTRUCTIONS



DIFFICULTY LEVEL	OPERATOR DEMAND LEVEL	RATING
VERY EASY, HIGHLY DESIRABLE	OPERATOR MENTAL EFFORT IS MINIMAL AND DESIRED PERFORMANCE IS EASILY ATTAINABLE	1
EASY, DESIRABLE	OPERATOR MENTAL EFFORT IS LOW AND DESIRED PERFORMANCE IS ATTAINABLE	2
FAIR, MILD DIFFICULTY	ACCEPTABLE OPERATOR MENTAL EFFORT IS REQUIRED TO ATTAIN ADEQUATE SYSTEM PERFORMANCE	3
MINOR BUT ANNOYING DIFFICULTY	MODERATELY HIGH OPERATOR MENTAL EFFORT IS REQUIRED TO ATTAIN ADEQUATE SYSTEM PERFORMANCE	4
MODERATELY OBJECTIONABLE DIFFICULTY	HIGH OPERATOR MENTAL EFFORT IS REQUIRED TO ATTAIN ADEQUATE SYSTEM PERFORMANCE	5
VERY OBJECTIONABLE BUT TOLERABLE DIFFICULTY	MAXIMUM OPERATOR MENTAL EFFORT IS REQUIRED TO ATTAIN ADEQUATE SYSTEM PERFORMANCE	6
MAJOR DIFFICULTY	MAXIMUM OPERATOR MENTAL EFFORT IS REQUIRED TO BRING ERRORS TO MODERATE LEVEL	7
MAJOR DIFFICULTY	MAXIMUM OPERATOR MENTAL EFFORT IS REQUIRED TO AVOID LARGE OR NUMEROUS ERRORS	8
MAJOR DIFFICULTY	INTENSE OPERATOR MENTAL EFFORT IS REQUIRED TO ACCOMPLISH TASK , BUT FREQUENT OR NUMEROUS ERRORS PERSIST	9
IMPOSSIBLE	INSTRUCTED TASK CANNOT BE ACCOMPLISHED RELIABLY	10

YES
IS MENTAL WORKLOAD LEVEL ACCEPTABLE?
NO
MENTAL WORKLOAD IS HIGH AND SHOULD BE REDUCED
YES
ARE ERRORS SMALL AND INCONSEQUENTIAL?
NO
MAJOR DEFICIENCIES, SYSTEM REDESIGN IS STRONGLY RECOMMENDED.
YES
EVEN THOUGH ERRORS MAY BE LARGE OR FREQUENT, CAN INSTRUCTED TASK BE ACCOMPLISHED MOST OF THE TIME?
NO
MAJOR DEFICIENCIES, SYSTEM REDESIGN IS MANDATORY.
OPERATOR DECISIONS

Modified Cooper-Harper Rating Scale Instructions

After you complete each of the tasks, you will be asked to give a rating on the Modified Cooper-Harper Scale for workload. In order to use this scale, you must understand several terms.

First, "primary task" refers to the computer tasks which are printed on the index cards in front of you. It includes finding the information in the computer files, remembering the information and finally typing the information into the main file accurately.

Second, you are considered the operator for these tasks. You will be operating the software and then using the rating scale to quantify your experience.

Third, the system refers to the the complete group of equipment you will be using in performing the primary task. Together you and the system make up the operator/system. For the present experiment, the system is composed of the computer screen, keyboard, and template as well as the software system which you are using.

Fourth, errors include any of the following: mistakes in remembering, typing or writing information, omissions of information, incorrect responses opening or closing windows, or otherwise moving around on the computer screens. In other words, errors are any appreciable deviation from desired operator/system performance.

Finally, mental workload is the integrated mental effort required to perform the primary task. It includes such factors as level of attention, depth of thinking, and level of concentration required by the primary task.

Rating Scale Steps

On the Modified Cooper-Harper scale you will notice that there is a series of decisions which follow a predetermined logical sequence. This logic sequence is designed to help you make more consistent and accurate ratings. Thus, you should follow the logic sequence on the scale for each of your ratings in this experiment.

The steps which you will follow in using the rating scale logic are as follows.

1. First, you will decide if the primary task can be accomplished most of the time; if not, then your rating is a 10 and you should write 10 on the rating sheet.

2. Second, you will decide if adequate performance is attainable. Adequate performance means that the errors are small and inconsequential in performing the primary task. Adequate performance is defined as completing each task as called for on the index card while pressing No more than one Incorrect F key or incorrectly typing and having to correct No more than one word (or a partial word). If adequate performance is not attainable, then there are major deficiencies in the system and you should proceed to the right. By reading the descriptions associated with the numbers 7, 8, and 9, you should be able

to select the one that best describes the situation you have experienced. You should then write the most appropriate number on the rating sheet.

3. If adequate performance is attainable your next decision is whether or not your mental workload for the primary task is tolerable. If it is not tolerable, you should select a rating of 4, 5, or 6. One of these three ratings should describe the situation you have experienced, and you would circle the most appropriate number.

4. If mental workload is tolerable, you should then move to one of the top three descriptions on the scale. You would read and carefully select the rating 1, 2, or 3 based on the corresponding description that best describes the situation you have experienced. You would write the most appropriate number on the rating sheet.

Remember, you are to write only one number, and the number should be arrived at by following the logic of the scale. You should always begin at the lower level and follow the logic path until you have decided on a rating. In particular, do not skip any steps in the logic. Otherwise, your rating may not be valid and reliable.

How You Should Think of the Rating

Before you begin making the ratings, there are several points that need to be emphasized. First, be sure to try to perform the primary task as instructed and make all your evaluations within the context of the primary task.

Second, the rating scale is not a test of your personal skill. On all of your ratings, you will be evaluating the system for a general user population, not yourself. You may assume you are an experienced member of that population. You should make the assumption that problems you encounter are not problems you created. They are problems created by the system and the instructed primary task. In other words, don't blame yourself if the system is deficient, blame the system.

Third, try to avoid the problem of nit picking an especially good system, or on the other hand, of saying that a system which is difficult to use is not difficult to use at all. These problems can result in similar ratings for systems with quite different characteristics. Also, try not to overreact to small changes in the system. This can result in ratings which are extremely different when the systems themselves are quite similar. Thus, to avoid any problems, just always try to "tell it like it is" in making your ratings.

If you have any questions, please ask the experimenter at this time.

WorkLoad Ratings

Rate each task on this sheet. ONCE YOU MAKE A RATING PLEASE DO NOT CHANGE IT.

Tutor Tasks

1) _____

2) _____

3) _____

4) _____

5) _____

Criterion Tasks

1) _____

2) _____

3) _____

Experimental Tasks

1) _____

2) _____

3) _____

4) _____

5) _____

6) _____

Place vita Here

The 5 page vita has been removed from the scanned document