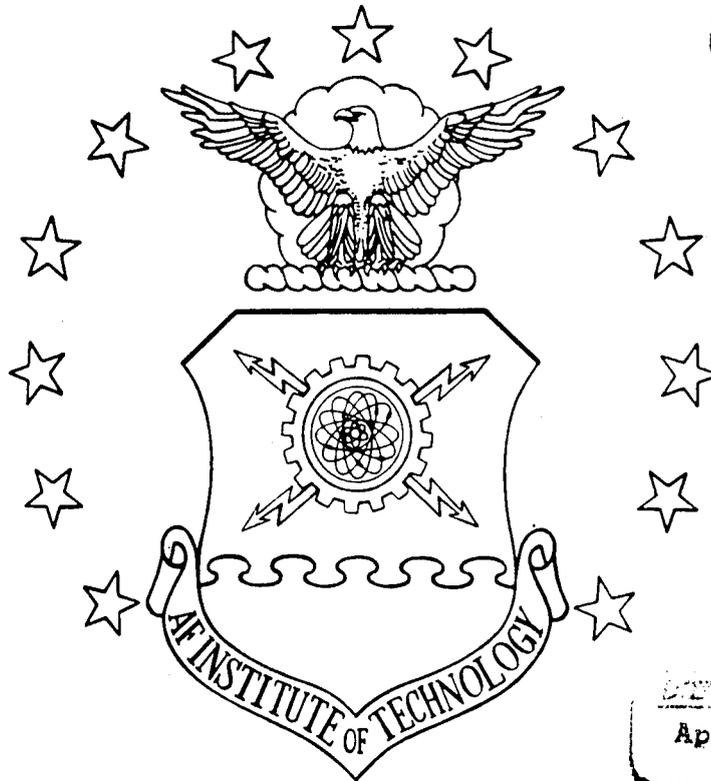


DTIC
SELECTE
DEC 27 1994
S C D



DISSEMINATION STATEMENT A
Approved for public release
Distribution Unlimited

**FACTORS IN HUMAN-COMPUTER
INTERFACE DESIGN
(A PILOT STUDY)**

THESIS

Susan Stewart, Captain, USAF

AFIT/GIR/LAR/94D-6

Accession

NTIS
DTIC
AFIT
GIR
LAR
94D-6

19941221 118

**DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY
AIR FORCE INSTITUTE OF TECHNOLOGY**

Wright-Patterson Air Force Base, Ohio **DTIC QUALITY INSPECTED 1**

AFIT/GIR/LAR/94D-6

FACTORS IN HUMAN-COMPUTER
INTERFACE DESIGN
(A PILOT STUDY)

THESIS

Susan Stewart, Captain, USAF

AFIT/GIR/LAR/94D-6

Accession For	
NTIS CRA&I	<input checked="" type="checkbox"/>
DTIC TAB	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification _____	
By _____	
Distribution /	
Availability Codes	
Dist	Avail and/or Special
A-1	

Approved for public release; distribution unlimited

The views expressed in this thesis are those of the authors and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

AFIT/GIR/LAR/94D-6

FACTORS IN HUMAN-COMPUTER INTERFACE DESIGN

(A PILOT STUDY)

THESIS

Presented to the Faculty of the School of
Logistics and Acquisition Management
of the Air Force Institute of Technology
Air Education and Training Command

In Partial Fulfillment of the
Requirements for the Degree of
Master of Science in Information Resource Management

Susan Stewart, B.A., M.S.

Captain, USAF

December 1994

Approved for public release; distribution unlimited

Acknowledgments

This study could not have been completed without the help of numerous individuals. I would like to thank my fellow classmates, Captain John Ellis, Captain Karen Cook, Captain Connie Hutchinson, Captain Alison McCoy, Captain Matt Pirko, Captain Brian Brown, Captain Susan Brown, Captain Brenda Forcht, Lieutenant Larry Cox, Lieutenant Dale Austin, and Mr. Pete Moseley for their divine wisdom and assistance. In addition, I would like to thank my thesis advisors Dr. Kim Sydow Campbell and Major Michael M. Shoukat for their guidance and support. I also received much needed assistance from Mr. Mike Rader, Major Mark Kraus, Dr. Guy S. Shane, Major Steven L. Teal, Dr. Robert P. Steel, Dr. David Barr, Dr. Steven K. Rogers, and Dr. Martin P. DeSimio. Sometimes you need to be hit over the head a few times before you have moments of epiphany. This group certainly knows which clubs are appropriate.

Finally, I want to thank my husband, E R, and sons, Scott, Keelan, and Niklaus, for their patience and understanding. They are the best.

Susan Stewart

Table of Contents

	Page
Acknowledgments	ii
List of Figures	v
List of Tables	vi
Abstract	vii
I. Introduction	1
Background	1
Importance of Research	1
Problem Statement	3
Research Objectives	3
Questions	4
Thesis Overview	4
II. Literature Review	5
Introduction	5
Expert System Overview	5
Importance of the Human-Computer Interface	9
Ravden and Johnson's Nine-Step Checklist	11
The Role of Consistency	22
Conclusion	25
III. Methodology	26
Introduction/Overview	26
Objectives	26
Prototype Design	26
Programming	27
Sections of the Prototypes	27
Versions of Sections	28
Consistency Features	29
Experimental Procedure	30
Subjects/Sample	30
Experimental Procedure	30

	Page
Data Collection	32
Data Analysis	33
Conclusion	36
 IV. Findings and Analysis	 37
Overview	37
Objective Data and Analysis	37
Complexity	46
Experimental Problems	48
Summary	49
 V. Conclusions	 50
Introduction	50
Overview	50
Conclusions	51
Recommendations	52
Future Research	53
Summary	53
 Appendix A: Prototype Version #1 - 4	 55
 Appendix B: Questionnaire	 186
 Appendix C: SAS Programs	 187
 Bibliography	 191
 Vita	 194

List of Figures

Figure	Page
1. Hardware and Software Trends	10
2. Good Versus Poor Visual Clarity	12
3. Good Consistency	13
4. Poor Consistency	14
5. Good Versus Poor Compatibility	15
6. Good Versus Poor Informative Feedback	16
7. Good Versus Poor Explicitness	17
8. Good Versus Poor Appropriate Functionality	18
9. Good Versus Poor Flexibility and Control	19
10. Good Versus Poor Error Prevention and Correction	20
11. Good Versus Poor User Guidance and Support	21
12. Prototype Layout	28
13. Design Characteristics	29

List of Tables

Table	Page
1. Timing Information	38
2. Readability Statistics	40
3. Testing Results	42
4. Dunnett's T Test	44

Abstract

The Department of Defense (DoD) has budgeted over \$9.8 billion for 1995 for information technology (Endoso, 1994:1), yet many workers let their existing systems sit idle. This thesis explores why these computers are sitting idle. This researcher's initial hypothesis was that certain features of the human-computer interface can positively or negatively affect efficiency, retention, and satisfaction level of workers. Although research is being done, interfaces continue to be of poor quality, especially in the DoD, where long procurement cycles, forced purchases, and limited budgets result in out-of-date software.

This thesis looks at four features and two interactions of the human-computer interface, color, white space, parallelism, and verb tense. While the researcher offers no significant statistical results, important advances are made in understanding the scientific method, experimentation, and the importance of these features. The results of the study discussed here suggest that features of the human-computer interface interact with one another, but this researcher's conclusion is that the extent to which any one or combination of features affect the overall performance of the subjects is impossible to ascertain using conventional linear methods. Instead, the sciences of complexity can account for the results of this study, since the entire effort made the erroneous assumption that the features independently affected the subjects' comprehension, satisfaction level, and speed. For managers and researchers, answering the question "Is this a linear relationship?" should be the first step in undertaking any research problem. If the answer is that the system is not linear, then more up-front consideration for modeling and analyzing the problem must be done for the project to yield any significant results.

FACTORS IN HUMAN-COMPUTER INTERFACE DESIGN

(A PILOT STUDY)

I. Introduction

Background

The DoD has budgeted over \$9.8 billion for 1995 for information technology (Endoso, 1994:1), yet many government office workers let their existing systems sit idle. This thesis explores why these computers are sitting idle. This researcher's initial hypothesis was that certain features of the human-computer interface can positively or negatively affect efficiency, retention, and satisfaction level of workers. Although some research is being done in this area, interfaces continue to be of poor quality, especially in the DoD, where long procurement cycles, forced purchases, and limited budgets result in out-of-date software.

Intuitively most programmers know the human-computer interface impacts on a person's ability to learn, but to what degree? This study is designed to test four basic human-computer interface features: color, white space, verb tense, and parallelism. In addition, it is designed to test the interaction between visual and linguistic features, color and verb tense, as well as between white space and parallelism.

Importance of Research

Money is always a key issue for both businesses, which are driven by a profit motive, and government, which is trying to get a handle on expenses. Putting software and computer interfaces into businesses or the government is costly, both because teams

of people are needed to develop even simple interfaces and because training for those interfaces is expensive. Once software is developed, people need to be able to use it. Training is expensive because finding people competent to operate new software packages is difficult and also because people must take valuable time away from their jobs in order to complete training.

Another reason why research in this area is important is that effective interface design can lead to a more productive workforce (Leibs, 1994). According to Bill Heflwy of the Software Engineering Institute at Carnegie Mellon (Leibs, 1994), one study revealed that an interface could trim as much as eight-tenths of a second off the time a user needs to do a repetitive task. Their study showed that one company would save almost \$2.5 million. Translated to the DoD, billions of dollars and millions of manhours could be saved in achieving higher productivity from fewer people.

Designing better interfaces is a major issue at many universities and businesses (Leibs, 1994). Because of the vast numbers of interface designers and trainers and the close-held nature of financial information, no one knows exactly how much is spent on interface design or training. Executrain, which claims to be the largest U.S. PC-training corporation, expects to earn \$90 million in 1994, an increase of more than one-third over last year (Leibs, 1994). In addition, over 700,000 people have been trained by Executrain since 1991.

In the Department of Defense, each branch of the service has its own teams of people who design software and conduct training. The DoD spends a great deal of its overall budget trying to manage information, but is still perceived as inefficient. Vice President Gore is trying to reverse this trend by streamlining the procurement system and by placing more emphasis on computer advancements. The Clinton administration is also continuing the Goals 2000 program, whereby educational and training reforms are to be in place and working by the year 2000. Heavy emphasis is placed, in this program, upon

computer literacy and training. In the Republican Agenda speech in November 1994, the next apparent Speaker of the House Gingrich claimed that the focus would be on Total Quality Management and streamlining government to make it more efficient. With these sweeping reforms, effective computer interface design will become critical. Training and support issues will need to be addressed in addition to the classical design issues.

It is the goal of this research to begin to lay the foundation for future human-computer interface design. If certain features or a combination of features of the human-computer interface can be positively shown to affect learning and retention, then they can be prescribed for use.

Problem Statement

Initially this thesis project was supposed to produce a computer-aided education course for the school. Once that scope was determined to be too wide, the project became to ascertain how much the human-computer interface affects learning, if at all. Are there certain features affecting speed and retention more than others?

To answer this question, the researcher developed one section of a course, IMGT 290, an Introduction to Computer Technologies, which incoming students of the Air Force Institute of Technology (AFIT) are required to take. Because of instructor manning, turnover problems, and inconsistent department expectations, this course had not been taught in two years. The prototype developed for this study instructs incoming students on one aspect of computer technologies--Internet's LISTSERV function--while testing their levels of satisfaction, speed, and retention of information.

Research Objectives

The main objective of this research is to determine the effect that the human-computer interface has on a user's: 1) accurate retention of information, 2) speed, and 3)

satisfaction level. When analyzing the results of this research, a secondary objective is to determine whether the results are translatable beyond the sample to government agencies as a whole.

Questions

Before undertaking this project, several investigative and management questions had to be answered. First, the researcher had to identify the primary interface features and then determine which of those features would be likely to affect the user's accurate retention of information, speed, and level of satisfaction. Once those features were identified, the researcher then had to determine how many features to test and how to test those that were chosen. Next, the researcher had to determine the population and sample size. Finally, the researcher had to determine if a model for feature selection and measurement was already in existence.

Thesis Overview

Chapter II presents a brief discussion of expert systems as educational aids and the importance of interface design. Chapter III discusses the research methodology, sample selection process, data collection steps, and data analysis in this project. Chapter IV examines the experimental problems and descriptive statistics of this study. Chapter V presents the conclusions and recommendations for future research.

II. Literature Review

Introduction

This literature review discusses the use of expert systems as educational aids and reviews the importance of the human-computer interface.

Expert System Overview

Just a decade or so ago people marveled at how well computers could manipulate numbers and formulas. Within the past few years, we came to realize that we have only uncovered a limited portion of the potential of computers. With the acceptance of the National Information Infrastructure, our government is spearheading the campaign for a global network. This network can have a dramatic affect on how the educational system works, particularly in the area of computer-aided education.

Because of the reliance this National Information Infrastructure will have on the existing Internet, this study explores using an expert system to teach the LISTSERV function of Internet (Levin & Baroudi, 1993; Snyder 1994; Cronin, 1994; Dern, 1994). This is just one example of how an expert system can be used to teach. There are many courses already in existence and many more to be developed (Garcia, 1990).

Because of the rapidly changing computer environment, this literature review is considered a snap-shot of today's technology with some insight as to how this technology will evolve in the years to come.

Expert systems provide a powerful educational tool for many reasons. According to Louis E. Frenzel, Jr., (1987) some advantages are;

- non-experts can do the work of experts,
- productivity can be heightened,
- time needed to accomplish tasks can be reduced,
- some operations can be simplified,
- repetitive tasks are easily performed,

- additional knowledge is permanently stored, and
- expert knowledge is available to a wider audience.

Expert systems have infinite patience. If a student requires hours of additional practice, the expert system will continue until the lesson has been learned. If, on the other hand, the student is already familiar with the material, that student can by-pass unnecessary steps, which makes the expert-system approach ideal for diverse groups of students. This is a great advantage that expert systems have over other more traditional computer aided education designs, which do not let the student control the flow of information.

Expert system designs can also diagnose weaknesses and suggest corrective actions. If a student is taking an electronic typing course and is good with typing all the letters, but often errs when typing numbers, the expert system can tally the mistakes and offer additional exercises involving more numbers. If properly programmed, the students can also use an explanation subsystem, which shows the sequence of rules that were used to reach a certain conclusion. Say the typing students were not sure why they were not passing the course; they could use the subsystem to see that they have exceeded the allowable number of mistakes when typing numbers. This additional resource leads to a more in-depth and personalized understanding of the material being presented.

In addition to more traditional expert systems, a new form of instructional aid, called intelligent computer assisted instruction, is also being used (Garcia, 1990). These systems use four modules which provide services such as:

- maintaining student information,
- tailoring instructions for individual students,
- organizing information for the students, and
- providing a friendly student-computer interface.

This program gives the student more control over how the information is presented. If the students have little prior knowledge of the subject, they can let the computer lead them through the material. If, on the other hand, they have a great deal of knowledge or are

preparing for a test, then the students can lead the sessions. In this way, the system adapts its performance to match those of its customers.

Of course, expert systems also have their faults. Developing a system is difficult and expensive. First, the developer must have adequate equipment. Most expert systems must still be run on minicomputers or mainframes. Sometimes, when microcomputers can not physically handle the memory required to run the system, the speed is drastically reduced.

Once the equipment is purchased, then the programmers must meet with content experts. Often content experts are hard to come by and getting information from them can be an arduous task, especially when the programmers are unfamiliar with the subject area. For example, several attempts at developing a system for medical diagnosis have been undertaken (Mockler & Dologite, 1987). Each of the attempts has met with commercial failure, because the programmers had to consult too many experts, each with conflicting opinions. Imagine going to different doctors with the same symptoms: fever and dizziness. A person is likely to get different opinions about what is wrong, based on each doctor's area of expertise. A general practitioner might tell the patient that they have the flu and that they should get plenty of fluids and rest. A surgeon might tell the patient that they have a cyst and that it is operable. A cancer specialist might tell the patient that they may have cancer and would recommend that patient have a series of tests. In programming expert systems, these inconsistencies would need to be resolved. In the case of the medical diagnosis expert system, the inconsistencies could not be resolved without the threat of lawsuits, so commercial applications have not become available.

Once all the proper equipment is purchased and the inconsistencies are eliminated, the programmer must ensure the system will work. According to Frenzel (1987), expert systems have been plagued with the problem of not being 100 percent reliable. Servers

can go down and experts can overlook critical details. Having a course scheduled when the computers are not working does not meet the customers' needs. Additionally, human judgment is still required, but the instructor does not need to be physically present to accomplish that task. That means that the instructors can be freed from the classroom and can then keep course materials current and administrative tasks up-to-date.

Many companies are realizing that computer aided education and training can be a viable alternative to the classroom (Maul & Spotts, 1993). In their study, the researchers found no significant difference between the effectiveness of classroom training and a computer aided education approach. They did find that 60 percent of their sample preferred the computer-run course, predominantly because they were allowed to progress at their own pace. The people completing the computer aided education course also finished much faster than the traditional classroom sample. The authors felt that computer aided education could be used when the number of students was high enough to compensate for the cost of implementing the course. They provided a break-even equation which employers and learning institutions could use to determine when computer-aided education courses could be used. Using this break-even equation could make computer-aided courses more feasible, especially in higher education (Solomon, 1994). To date most institutes of higher learning have not implemented computer-aided courses, with the most compelling reason why not being that multimedia and computer aided education have not been highly profitable (Solomon, 1994).

Solomon says the reason for this situation is that society needs time and motivation to make the transition from old to new technologies. He showed that society has undergone three revolutions in recent times. The first was the introduction of television in the classroom. Many thought television would revolutionize education. Although many millions of dollars were spent producing videos, not many classrooms incorporated them so the impact was small. The second phase was the introduction of microcomputers.

Once again universities spent billions of dollars and dramatically increased overhead costs, but the computers were drastically underused and as a result, not much has come from such effort and cost. Now the third revolution is developing multimedia computing. As usual, supporters feel this technology will dramatically enhance education.

Solomon says that several things will have to happen before computer aided education and multimedia production take hold in the classrooms. First he says that multimedia will have to be delivered inexpensively over a network. Multimedia will have to reduce operating costs, rather than increase them, if the colleges and universities are to remain competitive. Standards will have to be developed to allow mass marketing while commercial publishers will need to invest in producing high-quality products. The shift to a more computer-based education system seems rational as instructors are shifting their focus to publishing multimedia courses, rather than worrying exclusively about publishing textbooks and articles. Provided the same monetary incentives are offered for the courseware, this shift will continue.

Importance of the Human-Computer Interface

There are many aspects of the human-computer interface which must be considered before a designer codes the first line. According to Boehm (1981) Figure 1 depicts hardware and software trends.

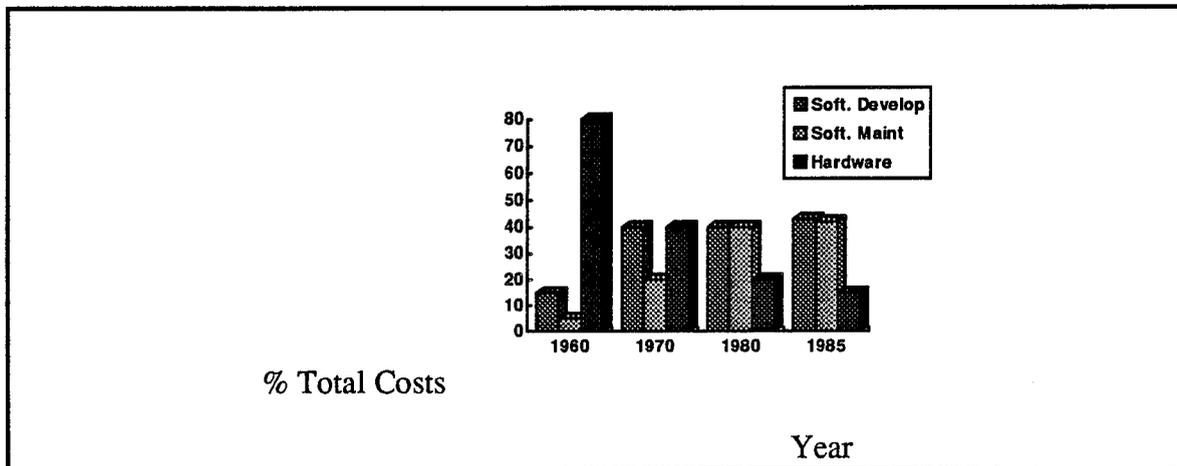


Figure 1. Hardware and Software Trends (Boehm, 1981: 6)

This graph shows how software development costs have been increasing relative to hardware costs over the years. The reason for the rise in software development costs is the trade-off between the interface and computing power (Leibs, 1994). For instance, in the early years of computing, most people using the systems were trained in both computing and programming. Most of the people did their own computing and were satisfied with the performance. As personal computers became more popular in the 1980's, average people with no previous computer experience began purchasing computers. Their complaints about the level of sophistication needed to do even simple operations on the computer were answered when Steve Jobs and Steve Wozniak developed the Apple computer and the first user-friendly interface. The price of this interface is high, with the tradeoff being that more memory is needed to run the applications. As the hardware and chip prices went lower, more memory was available for the interface and people began demanding better software. This trend continues today with windowing, the development of the Pentium chip, and the possible merger of Macintosh and IBM technologies.

The following section includes a discussion of a nine-step checklist for designing

interfaces and reviews current research in human factors and interface design which comes from *Evaluating Usability of Human-Computer Interfaces* by Ravden and Johnson (1989).

Ravden and Johnson's Nine-Step Checklist

In this book, the writers show that a delicate balance must be achieved between ensuring all users understand what they are doing and not being patronizing to those already skilled in computer operations. That balance is achieved through the development of an effective human-computer interface. The human-computer interface is the software that lets the user easily complete required tasks (Ravden & Johnson, 1989). In an educational setting, if the interface is poorly designed, then the students cannot be adequately judged on their ability to understand the material presented. Specifically, several things could affect their performance, such as misunderstanding the directions, confusion over what actions must be accomplished, inability of student to correct typographical errors, etc.

Ravden and Johnson developed a nine-step checklist for ensuring the human-computer interface is a good one. Their nine criteria are: 1) visual clarity, 2) consistency, 3) compatibility, 4) informative feedback, 5) explicitness, 6) appropriate functionality, 7) flexibility and control, 8) error prevention and correction, and 9) user guidance and support.

1. Visual clarity means that the information presented on the screen should be clear, well-organized, explicit, and easy-to-read. It encompasses features such as color, highlighting, fonts, and alignment.

Figure 2 shows an example of good visual clarity versus poor visual clarity.

Good visual clarity:

Visual clarity means that the information presented on the screen should be clear, well-organized, explicit, and easy-to-read. It encompasses features such as color, highlighting, fonts, and alignment.

Poor visual clarity:

Visual clarity means that the information presented on the screen should be clear, well-organized, explicit, and easy-to-read. It encompasses features such as color, highlighting, fonts, and alignment.

Figure 2. Good Versus Poor Visual Clarity

In Figure 2, one of the features which produces poor visual clarity is the combination of italicized writing with a script font.

2. Consistency calls for the system to look and act the same throughout the course. For example, if red is used to indicate errors in one section of the program, then it should be used to indicate an error everywhere. In addition, instructions should always appear in the same location, the cursor should always be placed in the same position for similar displays, and the menu should be consistent throughout the interface. Figures 3 and 4 show good versus poor consistency.

Good Consistency:

PAGE 1: Instructions: please read the following material.

Description: If a topic has additional information available, use the INFO? command to retrieve it. You can then use the INFO command again for specific topics.

Example: To: Internet'listserv@bitnic.bitnet'
Subj: Fuzzy Logic
info database, info refcard

PAGE 2: Instructions: please read the following material.

Description: Subscribing to a list is quite simple. In the body just type either the sub or signup command as shown above. If you want to try a Listserv, but don't feel like committing to the list today, try PRACTICE Listserv by following the example below.

Example: To: INTERNET'LISTSERV@ULKYVM.BITNET'
Subj: Your choice
SUB GSS-L Susan Stewart

Figure 3. Good Consistency

Poor Consistency:

PAGE 1: Instructions: please read the following material.

Description of topics: If a topic has additional information available, use the INFO? command to retrieve it. You can then use the INFO command again for specific topics.

Example-- To: Internet'listserv@bitnic.bitnet'
Subj: Fuzzy Logic
info database, info refcard

PAGE 2:

DESCRIBE: Subscribing to a list is quite simple. In the body just type either the sub or signup command as shown above. If you want to try a Listserv, but don't feel like committing to the list today, try PRACTICE Listserv by following the example below.

SUBSCRIBING EXAMPLE:

To: INTERNET'LISTSERV@ULKYVM.BITNET'
Subj: Your choice
SUB GSS-L Susan Stewart

Instructions: please read the above material.

Figure 4. Poor Consistency

In Figure 4, parallelism is inconsistent. For example, Page 1 entitles its description as "Description of topics," (i.e., a noun phrase) but Page 2 entitles its description as "DESCRIBE." (i.e., a verb)

3. Compatibility refers to the fact that the system should look and act the way people have become accustomed to systems looking and acting. The red used in this study to indicate an error would be appropriate since in our society red has traditionally been used to indicate errors. Graphical displays should be made horizontally or vertically, in line with what most people have come to expect. Deviations from what people are accustomed to can slow them. Since a benefit of computer-aided education courses is faster progression, deviations should be avoided. Figure 5 shows good versus poor compatibility.

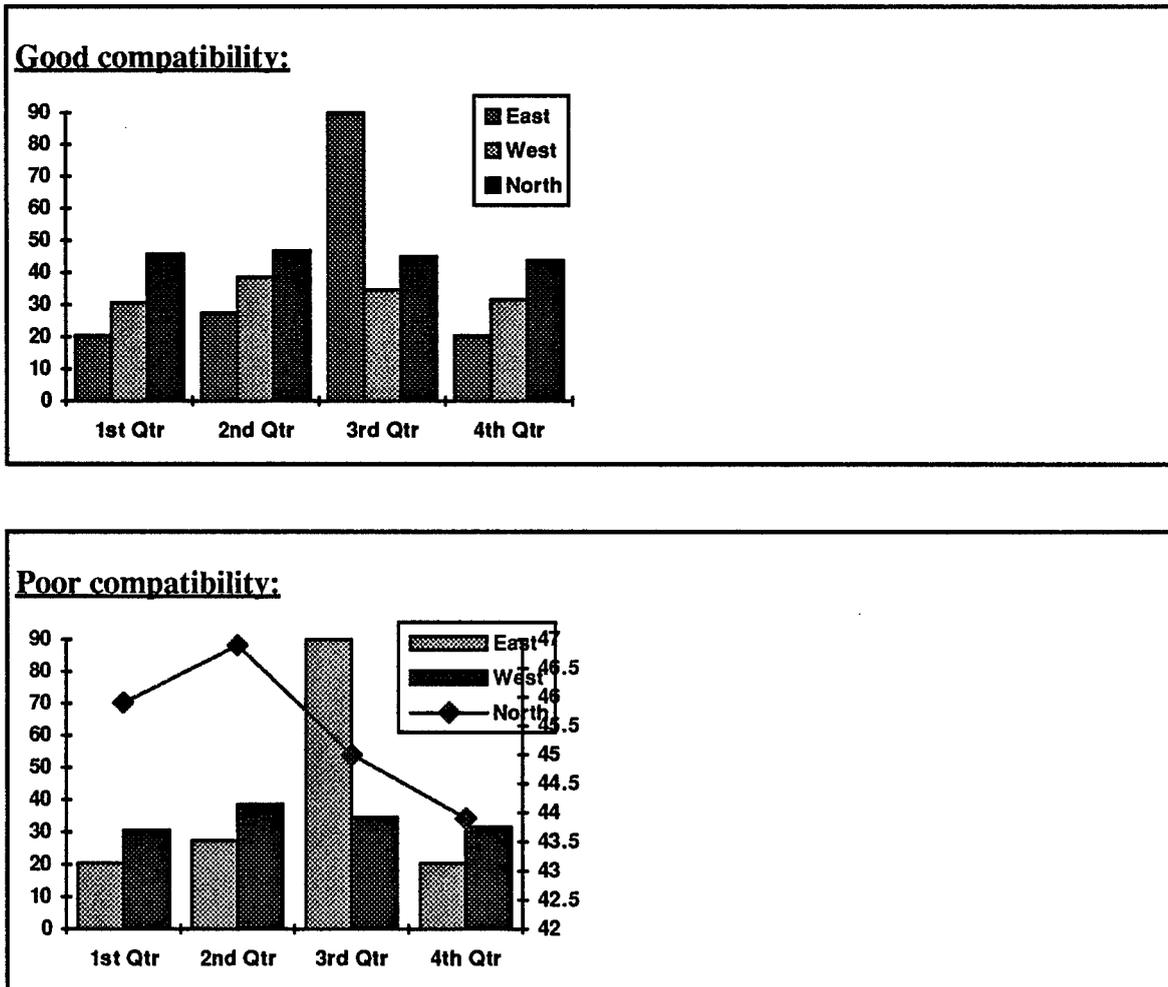


Figure 5. Good Versus Poor Compatibility

In Figure 5, one thing that makes the bottom chart poor is that traditionally East, West, and North have been considered as similar components. In the example of good compatibility, each of those features was represented by a bar. In the example of poor compatibility, East and West were represented by bars, but North was represented by points connected by a line.

4. Informative feedback are instructions and messages that let the students know they are on track. This feedback should be clear, precise, and relevant. Shortcuts should be presented for the computer-savvy folks and error messages should not only tell what the errors are, they should also tell where the errors are and why they occurred. Figure 6 depicts good versus poor informative feedback.

Good informative feedback:

You have answered question 1 incorrectly. The correct answer is "Programming." Please review page 3, paragraphs 4 and 5 of your course materials for further clarification. If you are still unsure why "Programming" is the correct response, please see your instructor.

Poor informative feedback:

One or more of your responses was incorrect. Please continue.

Figure 6. Good Versus Poor Informative Feedback

In Figure 6, no additional information was provided to the user and therefore the level of detail was not sufficient for meaningful analysis.

5. Explicitness involves structuring the system so that it is clear for the student. This differs from compatibility in that it looks at how the screens are organized and how the different parts of the system are presented. It differs from clarity in that clarity looks at the words and pictures themselves for a person's ability to see the screen. Explicitness looks at the screen layout to ensure a logical flow of information. Figure 7 shows good versus poor explicitness.

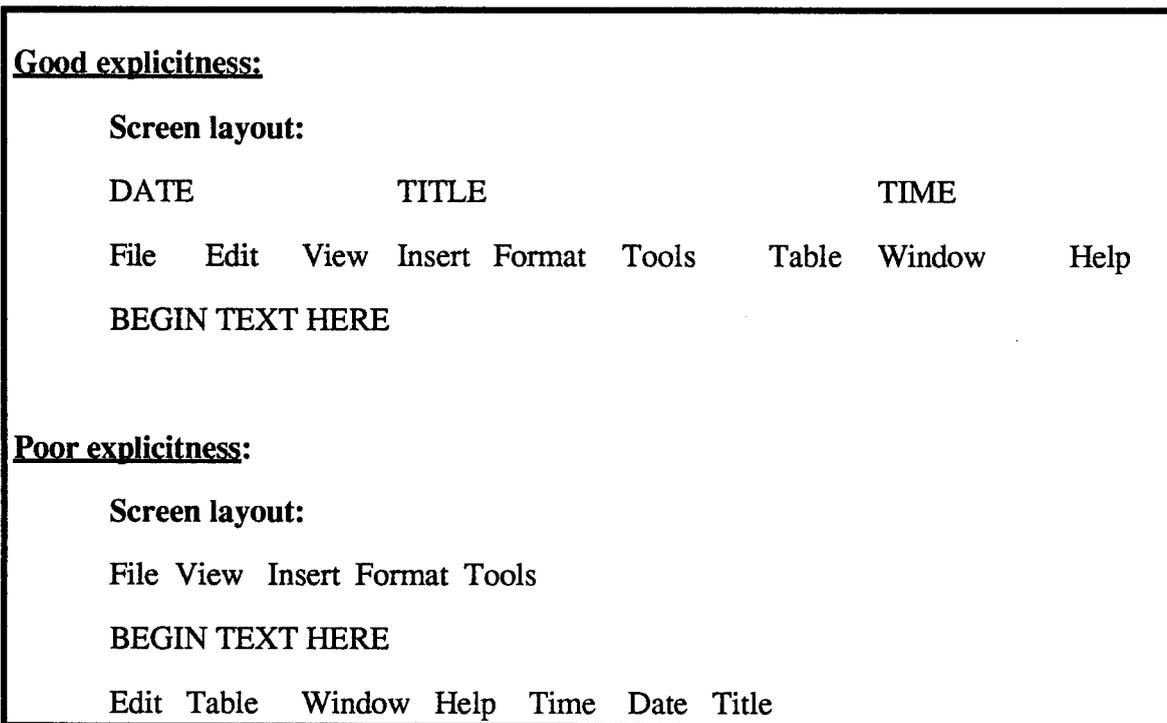


Figure 7. Good Versus Poor Explicitness

In Figure 7, the fact that some of the tools are above the text and some tools are below the text makes it more difficult for the user because of its cluttered and awkward appearance.

6. Appropriate functionality ensures the system is meeting the needs of the students as they carry out tasks. For example, the joystick or mouse should move analogously to the cursor, options should be clearly displayed at all stages of a given task, and jargon should be clearly defined. Figure 8 shows good versus poor appropriate functionality.

Good appropriate functionality:

Screen layout:

File Edit Options Help

Please read the instructions carefully. If at any time you are unsure of the instructions, please consult the Instructions section under the Help screen.

Poor appropriate functionality:

Screen layout:

File Edit

Please read the instructions carefully. You will not have the opportunity to review the instructions once you press the <Enter> key. Once you complete the instructions, continue with your next lesson.

Figure 8. Good Versus Poor Appropriate Functionality

In Figure 8, the feature which makes the first screen have good appropriate functionality is that it gives the user options if problems arise.

7. Flexibility and control empower students to a certain degree. By building flexibility and control into the human-computer interface, the designer can give the student control over the material being presented. If defaults are set, the student can override them easily. The user can determine how fast to progress through the lesson and can skip areas which are already understood. The students should also be able to tailor certain interface components, such as color, to their liking. This capability makes the end-user much more comfortable with the program, since undesirable aspects of the program can be easily altered. Figure 9 depicts good versus poor flexibility and control.

<p><u>Good flexibility and control:</u></p> <p>Below, please find a list of menu options which you can use to determine how your course is presented:</p> <p>COLOR: Press the Options, Color toolbar to alter both background and text colors</p> <p>SPEED: Press the Options, Speed tool to alter the pace of the material being presented.</p> <p>SKIP THIS SECTION: Press <Esc> to skip a section</p> <p><u>Poor flexibility and control:</u></p> <p>Please let the instructor know if you do not like the screen colors or speed of the material being presented. He/she will then assist you in altering the interface.</p>

Figure 9. Good Versus Poor Flexibility and Control

In Figure 9, the user has the option to control the screen's color, speed, and lesson order. This empowers the student and makes the first screen have good flexibility and control.

8. Error prevention and correction is another area of concern for the human-computer interface. The interface should minimize the possibility of errors and should be flexible enough for students to check and correct errors or potential errors before their input is processed. The system should be easy to use, perhaps even with an undo/redo feature. Figure 10 shows good versus poor error prevention and correction.

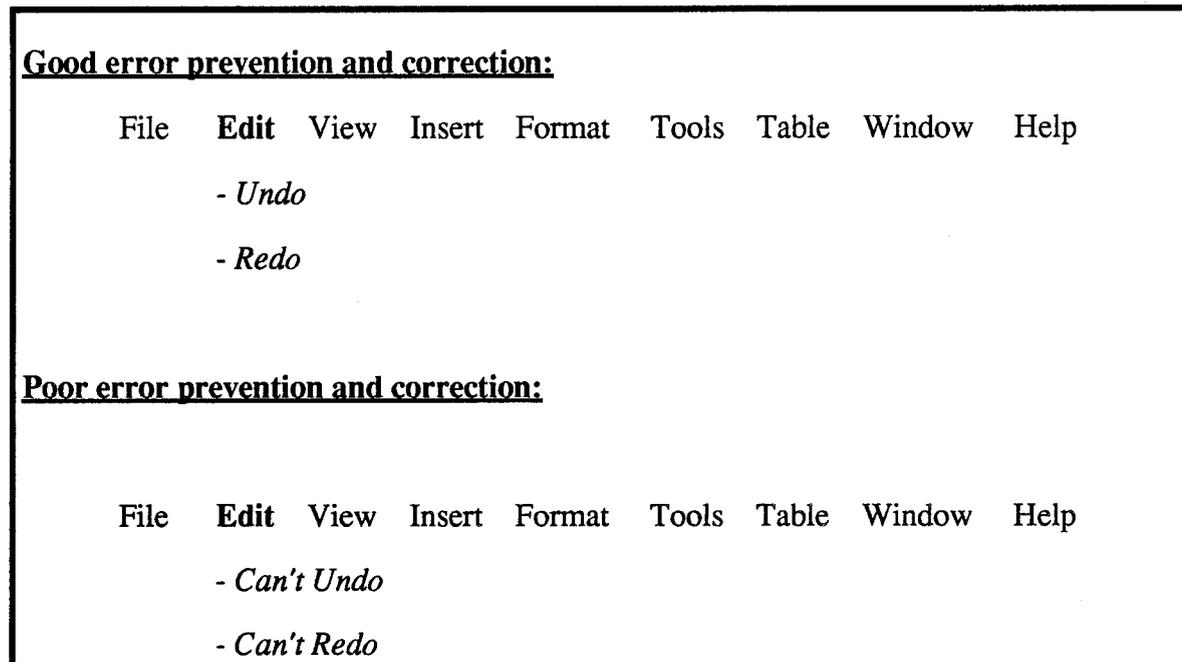


Figure 10. Good Versus Poor Error Prevention and Correction

In Figure 10, the users of the good screen can undo and redo typing and formats if they have made a mistake. In the bottom screen, the users must continue without making any changes or corrections.

9. User guidance and support is the final area explored by Ravden and Johnson. User guidance and support includes comprehensive on-line guidance as well as hard copy documentation. Once again, clarity of the guidance is the most important feature. Figure 11 shows good versus poor user guidance and support.

Good user guidance and support:

... Please see the help pull-down menu for any on-screen assistance or refer to your documentation package. If you still have questions about the material being presented, please consult your instructor.

Poor user guidance and support:

... Please consult your instructor if you have any questions about the material being presented.

Figure 11. Good Versus Poor User Guidance and Support

In Figure 11, the good user guidance and support screen gives sufficient detail for the user to answer questions. The bottom screen places all the responsibility on the instructor.

The Role of Consistency

The interface links the student, the machine, and the information to be learned. The easier the students can learn the mechanics of the system, the more time they can spend on-task. Although all nine features presented by Ravden and Johnson (1989) are important for an effective human-computer interface, this researcher chose to limit research to consistency because of time and resource constraints. Within consistency, this researcher looked at both visual and linguistic features.

In reviewing the literature on this subject, one of the major problems was that researchers come from different fields and therefore use different terminology. While several journals such as Human Factors, IEEE Transactions on Systems, Man, and Cybernetics, and IEEE Transactions on Professional Communication dedicate themselves to interface design there appear to be few specific articles for consistency. Nevertheless, consistency is widely recognized (Ramey, 1988 p 156; Campbell, 1995, p 97-1). First, Fisk and Jones (1992) showed the effect consistency had on performance and learning. They studied the effects of varying degrees of global or higher-order consistency using a visual search task. They used a small sample of graduate students who were first administered the Wechsler Adult Intelligence Scales to ascertain that the students were within normal population performance levels for digit-span, digit-symbol substitution, and vocabulary. The students were then given tasks which were either completely consistent wherein all inputs served as a target, not as a distractor; partially consistent, wherein some inputs served as a target for 25%, 50%, or 75% of the time and as a distractor for the rest of the time; or totally inconsistent, wherein inputs would randomly serve as targets or distractors. Their conclusion was that people can retain more information more rapidly when given consistent input.

Another article which shows the importance of consistency in interface design is "Compatibility and Consistency in Display-Control Systems: Implications for Aircraft

Decision Aid Design" by Andre and Wickens (1992). In this article, the researchers concluded that consistency possibly outweighs the importance of compatibility when constructing display controls. Their study involved having subjects perform an instrument monitoring task that required them to either vocally report or manually correct an abnormal status when prompted by either a status report or a command display on their screen. The researchers tested consistency by having some subjects receive both status and command displays and other subjects receive either status or command displays. The results showed that global consistency overrides local compatibility for both the status and command display groups. Performance of the manual correction tasks and on the vocal report tasks were better with the two consistent groups. The researchers also ran tests for response time and tracking error data which also indicated consistency was more important, but to a lesser degree.

In their article, "Display Formatting in Information Integration and Nonintegration Tasks," Boles and Wickens (1987) cite further research into visual displays by using multiple resource theory. They concluded that nonintegration (dual task) environments benefitted from a mixed-format display and integration environments benefitted from having a pure-format display. To clarify, nonintegration tasks are designed to handle dual-task situations where two stimuli are mapped to two responses. Integration tasks are when two or more stimuli are mapped to only one response. Boles and Wickens also suggest that if speed of performance is critical, the interface designer should use analog (such as bar charts) rather than verbal or digital displays. For this project, the three sections would be considered to be based on integration tasks, because multiple stimuli were mapped to one expected response. The difference between integration and nonintegration tasks was not tested, but consideration was given in how the material was being presented.

Overall, this study looked at consistency as its major feature. Within consistency,

several specific features were considered, such as color, white space, parallelism, and verb tense. The following literature represents some of the current research being done in this field. These articles are reviewed to highlight the background that shows why this researcher selected color, white space, parallelism, and verb tense as specific features to study. While many of the researchers tested color and white space, parallelism and verb tense were selected based on the recommendation of Campbell (1995).

In their article, "Emergent Features and Graphical Elements: Designing More Effective Configural Displays," Bennett, Toms, and Woods (1993) concluded that the graphical elements of a display may be improved by emphasizing scale, spatial separation (white space) and color-coding.

Wickens and Andre (1990) also experimented with color and space. In experiment #1, they tested spatial proximity between indicators and their distinctiveness in color. The results showed that spatial proximity had little effect on attention or integration, while distinct color code did improve attention and disrupted integration. In experiment #2, they tested color coding by presenting three indicators as a bar graph or as features of an object display. They concluded that the presence of color borders restored focused attention, but with a cost to response time.

In the article, "Legibility and Subjective Preference for Color Combinations in Text," Pastoor (1990) presented an experiment done to test for legibility performance and subjective preference for text/background color combinations displayed on a video monitor. His conclusions were that there was no evidence that different color combinations were influential. The only conclusion was that if light-on-dark text was presented, subjects seemed to prefer cool background colors such as blue and bluish cyan. If this is the case, then the consistency research done in this thesis would be more convincing, since individual color preferences were not believed to be important. Therefore, if any conclusive results were yielded from this study, they would not be

attributed to the particular colors chosen for any of the prototypes.

The final article considered for this study was "Ecological Interface Design: Theoretical Foundations" by Vicente and Rasmussen (1992). In this article, the researchers describe the ecological interface design (EID) as being based on the skills, rules, and knowledge of cognitive control. Their basic goals were to keep processing to what is needed to get the task accomplished and to support each of the cognitive control levels. They accomplished this by first looking at the novelty of the situation. Are the events similar to those done before? Are the operators experienced in this area? Was the situation anticipated? Once they established the situation along a continuum, they then looked at the structure of the interface design problem. They identified that the interface consists of content, structure, and form. For each of these areas, the interface must account for both lower level cognitive exercises, such as routine database entries, and for higher level exercises, such as unique project management tasks. To ascertain exactly how to design the interface, they concluded that a goal must be established based on which activities are most likely to be done. Apple is using this concept in its upcoming System 8 and 9 operating systems by breaking software into components, thereby simplifying operations and giving the customer exactly what is necessary for the individual company rather than generally serving the industry (Leibs, 1994).

Conclusion

This literature review discussed some of the pros and cons of developing expert systems for use in the classroom. It also discussed the importance of the human-computer interface, specifically in the importance of further research in consistency. As we understand more about which factors of the human computer interface influence learning the most, we can develop appropriate software and courseware.

III. Methodology

Introduction/Overview

This section will explore the specific methodology of this study by noting the objectives, prototype design, experimental protocol, and data analysis. The overall methodology was a pretest-posttest control group experimental design using different prototypes to test the effects of consistency features of the human-computer interface.

Objectives

The main objective of this research is to determine the impact that the human-computer interface has on a user's: 1) accurate retention of information, 2) speed, and 3) level of satisfaction. When analyzing the results of this research, a secondary objective is to determine whether the results are translatable beyond the sample to other users in government agencies.

Prototype Design

The design of the prototypes used in this study was simple because of time constraints and the necessity of developing multiple versions to test each treatment (Schriver, 1989). Using Ravden and Johnson's checklist for each of the nine factors, the programs were written and analyzed. See Appendix A for complete programming code. Following a short pretest, it was concluded that hypertexting and the potential for students to review sections at their own pace should be eliminated. Because experimental data is difficult to obtain and these students are kept to a strict schedule, there was a need to keep the program to a manageable size, with each section taking about five minutes to complete.

Programming

This experiment was done using an expert system shell, VP-Expert. The two primary books used to develop the expert system were *An Introduction to Expert Systems* by Robert J. Mockler and D.G. Dologite and *Illustrated VP-Expert* by Wally Wang and John Mueller. The reasons VP-Expert was selected for this prototype were: ease of use, familiarity, and cost. In addition to the two aforementioned books, this researcher also consulted *Artificial Intelligence: How Machines Think* by F. David Peat and *Crash Course in Artificial Intelligence and Expert Systems* by Louis E. Frenzel, Jr. Both these books provided specific background materials and examples to clarify and strengthen the expert system prototypes.

Sections of the Prototypes

There are three sections of course material in the prototype expert system. The first section is a background of LISTSERV. The second section is problems associated with LISTSERV. The final section is a How To section which outlines some common commands. Each section is followed by a five-question quiz over its content.

Versions of Sections

In order to test the impact of computer-interface design features, the first section was used as the control and the second and third sections were altered to provide three versions of each based on the consistency of two interface features. Figure 12 shows exactly how this was accomplished:

	Sections			
	Color	Verb Tense	White Space	Parallelism
Version 1	C	C	C	C
Version 2	X			X
Version 3	I	I	X	
Version 4		X	I	I

Figure 12. Prototype Layout

Key: X: Indicates the item was exclusively tested

C: Indicates the control group

I: Indicates the item was tested interactively with another item in that row

Although this is a pilot study for these prototypes and there are no other known models using this design, the prototypes were designed based upon the classic pretest-posttest-control group model (Campbell and Stanley, 1963). Figure 13 shows the design according to Campbell and Stanley:

CLASSIC PRETEST-POSTTEST CONTROL GROUP DESIGN:

O X O

O O

THIS STUDY:

O X1 O X2 O

O X3 O X4 O

O X5 O X6 O

O O O

Figure 13. Design Characteristics

WHERE O stands for Observation and X stands for the treatment. In this case, the observations were made by tests administered following each section and times which were taken throughout the experiment. The treatments were prototype sections, which were altered based on the consistency of interface features as described in Fig 12.

Consistency Features

To better understand this study, the following explanations are required:

1) Color: In this study, colors were used either consistently (red means mistake, yellow means the passage is highlighted) or inconsistently (red means correct or mistake, highlights are green or blue or white, etc).

2) Verb Tense: Verb tenses were also used consistently (all past tense) or inconsistently (some present, some past, and some future).

3) White Space: White space is when a passage uses lists to reinforce learning. In this study, one prototype gave lists and another presented the material in paragraph form and lists together, but with no significance to the different forms being together.

4) Parallelism: Parallelism is when items are presented in the same manner and format each time they are presented. In this study, parallelism was either consistent or

inconsistent. See the consistency example in the previous chapter for an example of parallelism.

Experimental Procedure

Subjects/Sample

Due to the limitations of time, cost, and availability of the sampling population, a convenience sampling was used in this research. The convenience sample used in this study consisted of incoming logistics graduate students at the Air Force Institute of Technology who are required to take the course Introduction to Computer Technologies. One portion of this course was dedicated to teaching how to use the Internet. Since the instructor wanted to teach a block on the LISTSERV function, the prototype for this study was developed which would serve both as instructional tool for the course. Since sampling people takes time and money, this solution offered both value for the subjects and a controlled experimental situation for the researcher.

Experimental Procedure

Once the prototypes were developed and beta tested, then the researcher coordinated how they would be administered to the subjects during their introductory course. Several steps were necessary for the administration of this experiment. They included determining when and where the prototypes would be given, how the experiment would be introduced, and how the data would be collected. In addition to the actual experiment, data was also collected about the readability of the tutorial.

First, it was determined that the Internet coursework would be taught on the second day of a two-day course. Since the researcher could not guarantee how long the experiment would take each subject to complete, it was further determined that the

prototypes would be administered during the first hour of the second day. Since the subjects would need access to computer terminals, the experiments were conducted in the computer lab of the Air Force Institute of Technology. This room has 17 terminals which limited our class size for any one group.

Once when and where issues were resolved, the next task was to outline how each of the sections would be run. Because of the small size of the groups, it was determined that one or two people could administer the experiments. This helped the researcher maintain reliability because the groups were given the same information each time.

More specifically, each session was conducted by this researcher with some assistance by the advisor during the first few sessions. The sessions were begun with an introduction to the experiment. Since the subjects were not familiar with the computer terminals at AFIT, they were given assistance in logging in and bringing up the program. The program itself was stored on a common network drive, allowing each subject to access the program without having to copy the program on each of the 17 terminals.

Once the subjects had accessed the VP-Expert shell, they were instructed to stop and to listen how each section of the prototypes worked. Since a separate Norton Utilities shell was used for timing information, there were two steps one needed to complete between each of the sections of the prototypes. Appendix A gives complete programming requirements for the Norton Utilities shell. Basically the shell took a clock reading as the sample subject began and ended each section of their tutorial. The shell also added all the times and gave an overall reading at the end. In keeping with the recommendations given in the Literature Review section, both on-screen and physical instructions were available. The subjects were instructed how to access this program element and were shown the steps on a white board in the front of the room as well as advised that the instructions would appear on their screens. The steps consisted of hitting the <enter> key three times following the end of each of the first two sections and one time at the end of the last

section of the course material. In addition, they were told that they would be completing a questionnaire following the coursework. They were also shown how to complete the necessary forms. After all questions were answered, the subjects counted-off (1-4). They were then given instructions how to access each version based on their number. These instructions were to type k:\vpctest(1-4). Following any additional questions, they began the experiment.

After each of the subjects completed their three sections of each prototype, they were given their questionnaires and an electronic scan sheet. Once each person had completed a questionnaire, it was checked for completeness and accuracy. In addition to the specific features, the researcher also collected demographic data. A sample of the questionnaire can be found in Appendix B. The researcher and advisor collected the questionnaires and the subjects were placed on break until everyone had finished. During this time, the researcher collected data and ensured each terminal was returned to its original program. The subjects were then brought back into the room, thanked for their participation, and given the remainder of their Internet briefing.

Data Collection

The final step in the process was actually to gather the data. In this experiment, three separate methods were used to collect the data. First, the subjects were timed using the Norton Utilities package. Timing information was programmed to go into an individual account for each subject based on the subject's log-in name. This information was then captured on the c:\ drive under the person's user name. Second, the subjects were tested following each section of their tutorial. These data were collected using the VP-Expert shell. The data included which version of the tutorial was being used, which specific answers were given, and whether that answer was correct or incorrect. These data were also captured on the c:\ drive under the person's user name. Finally, the

subjects were given a questionnaire. They were asked to give their user name on the electronic scan form so that all the data could be captured on that scan form following the testing. The subjects then bubbled in each of their responses to the questionnaire and turned the questionnaire and scan form in to the researcher or advisor. Once the person was placed on break, the researcher copied that person's data file from the c:\ drive onto a disk, checking to ensure that the user names were the same both on the file and on the scan form. Once all the subjects were done and all the forms and files were collected, the researcher went to a separate computer room and printed the files. The files were then matched with the scan forms and the results of the tests were bubbled in. When all the forms had been completed, the researcher used a scanning device to create a data file which was subsequently used as the data file for statistical analysis. Since the timing information could not be entered on the scan form without giving ranges rather than specific times, those data were handled separately.

The timing information was captured on the printed file and was then transferred to a spreadsheet. Realizing that sometimes errors can occur in the transfer process, the numbers for each of the files were added and compared to the totals on the spreadsheet to ensure accuracy. This spreadsheet was then used to run statistical analysis of the timing information, whereas SAS was used for the scan sheet data.

In an attempt to understand the results of the experiment, another data collection effort was done to ascertain the readability of the tutorial. The programs were converted to a Microsoft Word document to permit further analysis.

Data Analysis

The hypothesis tested was as follows:

H_a : The four sampled populations do not have identical features

Once all the raw data were gathered, several analytical tools were available to the

researcher for subsequent statistical analysis. The two major computer software packages used were Microsoft's Excel and SAS. There were three general types of tests done, ANOVA, regression analysis, and the Dunnett's test.

This researcher chose to perform an analysis of variance because of its ability to compare more than two means at the same time (Freeman & Godfrey, 1991). To give some background on the procedures used in this thesis, several definitions are provided. By looking at the F-test in the ANOVA results, this researcher could see if the means were significantly different from each other, but it did not tell her which means differ from which other means. In other words, the F-test used in this study could tell the researcher if the results of one sample group were different than the results for a different sample group on any one of the tests.

Second, this researcher used regression analysis to determine the relationship between one dependent variable and another set of independent variables (Freeman & Godfrey, 1991). In this case, regression analysis would tell the researcher if color would influence a person's ability to answer a question in the second section of the tutorial.

Finally, this researcher used Dunnett's test which is a special case of means comparison that can compare a set of new treatments and a single control (Freeman & Godfrey, 1991). This differs from more standard General Linear Model (GLM) statistics because of its ability to compensate for uneven sample sizes.

As was stated in the previous section, this researcher had four different types of data to consider--timing, test results, questionnaire responses, and readability.

Microsoft's Excel was used to analyze the timing information. Each subject's times were collected for each of the sections. The times were then added for each version (1-4) of the prototypes. Aggregate times for each section of each version were then compared. For example, all times from the people who used the first version were added together and then divided by the number of subjects for each of the three sections of the tutorial. Using

a one-way analysis of variance, this average time was then compared to the average times of groups 2-4 to see if there was any difference in the samples. The first section was used as a pretest. In this section, the groups were compared to see if they were different. If the groups were statistically different in this pretest, then the results of other tests would be skewed because the sample would not be representative of the population.

In addition to timing information, data was also collected on test results for each of the sections. Adding and dividing the raw data was not necessary for test results. The raw data were transferred from the scan sheets into a datafile and that datafile was analyzed using ANOVA and Dunnett's test.

The third type of data collected during this effort was questionnaire data. The data were again captured on the scan form and were directly used by the SAS program. In this analysis, the raw data were analyzed based on different groupings of items which were designed to assess subjects' perceptions of these qualities of the prototype: Accuracy, Clarity, Coherence, Completeness, Continuity, and Distinctiveness. The ANOVA and Dunnett's test were done in addition to a paired-comparison test, which told the researcher at a glance whether the results were statistically significant. One mathematical tool which was extensively used in these programming efforts was based on Fisher's test for comparing features. This tool is extensively used to analyze models similar to the one in this study; the tool's usefulness will be further discussed in the next chapter (McClave, & Benson, 1991). Examples of specific data analysis programs for these tests are provided in Appendix C.

One final test was done on the VP-Expert programs themselves. The programs were tested for readability. Once the files were converted to Microsoft Word, they were analyzed for readability using the grammar checker. The rationale for testing for readability was that if the versions of prototype sections differed in the amount of time it took for subjects to get through the material, it would not be because the material

presented was of a different reading level.

Conclusion

Chapter III discussed the methodology that was followed in completing this research project. Topics included the objectives, prototype design issues, experimental procedure, and data analysis. The overall methodology used for this study was a pretest-posttest control group experimental design using different prototypes to test for features of the human-computer interface. This formal methodology approach provided useful data necessary to answer the investigative questions. Specific data collection results and analysis of the collected data are reported in the next section, Chapter IV.

IV. Findings and Analysis

Overview

This research focused on the effect of certain features of the human-computer interface on a user's accurate retention of information, satisfaction level, and speed. This chapter describes the findings and analysis of the results of experimentation by providing a discussion of the objective data and analysis, subjective questionnaire data and analysis, complexity, and experimental problems encountered during this study.

Objective Data and Analysis

The experiments were run during the first three weeks of the summer short term at AFIT. A total of 79 people were administered the prototypes on Tuesdays and Thursdays of each week. The subjects were seated in rows of 5 or 6 across and were briefly introduced to the mechanics of the experiment, without knowing what was being tested. Since there were six separate groups, the subjects were not told the nature of the experiment until all groups had been tested.

As was stated in the previous chapter, this researcher had four different types of data to consider--timing, readability, test results, and questionnaire responses. There were also three general types of tests done for this thesis, ANOVA, regression analysis, and the Dunnett's test. The following series of tables represent the results of the data collection efforts. Each table is followed by a brief explanation of the meaning of the specific results.

Table 1 in this section shows timing information. This information was gathered to show if one group of individuals differed from another in terms of the amount of time it took them to complete a section of the prototype course material. The assumption was that if the control group (Sample 1) did not differ from the other groups during Section A (or the control section) of the tutorial, then that control group could be considered a good

sample of the population. In addition, if the control group was representative and if any other group differed during any of the other sections of the tutorial, then presumably that difference would be because of the specific treatment applied (color, verb tense, white space, or parallelism) in the version of that section that they were given and not because they are different from the control group.

Table 1
Timing Information

	Tutorial Section A (in seconds)	Tutorial Section B (in seconds)	Tutorial Section C (in seconds)	Dunnett's (mean comparison)	Readability Statistics (grade)
Sample 1	315.095	297.333	326.381	Control	10.1
Sample 2	286.737	271.842	297.632	1.47	10.1
Sample 3	286.789	284.947	311.579	0.96	10.1
Sample 4	316.211	298.632	334.158	0.91	10.1
SUM (1+4)/2 MINUS SUM (2+3)/2	28.89	19.588	25.664 sec		

Key:**Sample 1:** Control group**Sample 2:** Group given interfaces with inconsistent color and parallelism**Sample 3:** Group given interfaces with inconsistent color/tense & white space**Sample 4:** Group given interfaces with inconsistent tense & white space/parallelism**Tutorial Section A:** Control section**Tutorial Section B:** Inconsistent color and tense interface**Tutorial Section C:** Inconsistent white space and parallelism interface

Discussion: Table 1 depicts several things. The first three columns show the average time it took each sample group to complete each of the three sections of the tutorial. The last row in these columns shows the difference between groups one/four (control groups) and groups two/three (treatment groups) to be roughly 30 seconds for the same section of the tutorial. The results of the Dunnett's test in column four shows the comparison between each group and the control group. Finally, the readability level results show that all versions of the tutorial were written at the Flesch-Kincaid reading level of grade 10.1. While there is no statistically significant difference--indicated by the high numbers in the Dunnett's test--among the four groups, there is some practical difference. As was mentioned, control groups 1 and 4 differed from treatment groups 2 and 3 by about 30 seconds when tested with the exact same material. In fact, groups 1 and 4 consistently took longer to complete each section of the tutorial than groups 2 and 3. This could be attributed to the fact that this research used convenience samples, rather than a random selection process, because a convenience sample does not typically represent a larger population with the statistical strength that a random sample does.

With so great a difference between the two sets, a possible future measure would

be to analyze each individual's reading comprehension level first by giving a standardized intelligence test (Bennett & Toms, 1993) and then separate the groups based on similar scores. For this study, the readability level of the prototype tutorial is listed in Table 2.

Table 2
Readability Statistics

Counts	
Words	1562
Characters	7535
Paragraphs	61
Sentences	92
Averages	
Sentences within the Paragraph	1.5
Words within the Sentence	17.0
Characters within the Word	4.8
Readability	
Passive	27%
Flesch Reading Ease	59.7
Flesch Grade	10.1
Flesch-Kincaid	8.7
Gunning Fog Index	10.5

Discussion: These figures are provided to show that the material presented was written at a sufficiently accessible level. The numbers give a rough estimate of the level at which the material was written, but they are not truly accurate since the software used is limited by its programming. For instance, if a complex sentence structure was used or if a word was used which was not in the program's dictionary, the program would categorize those as being of a higher readability level. Since many words used for describing LISTSERV and the Internet are not common dictionary words, this could have skewed the readability index somewhat. This would not affect the overall conclusions however, since the numbers would be the same for each of the sample groups.

Once timing and readability results were analyzed, the researcher looked for significance in the test results. Several statistical tests were run including regression and ANOVA. SAS's PROC REG command was used because it (Freeman & Godfrey, 1991):

- handles multiple regression models;
- provides nine model-selection methods;
- allows interactive changes both in the model and in the data;
- allows linear inequality restrictions on parameters;
- tests linear hypotheses and multivariate hypotheses;
- saves estimates, predicted values, residuals, confidence limits, and other diagnostic statistics in output SAS data sets; and
- generates scatter plots of data and of various statistics.

Dunnett's test was used to compare each of the results with the control group.

Table 3 shows the results of testing.

Table 3
Testing Results

	Tutorial Section A (Dunnett-F)	Tutorial Section B (Dunnett-F)	Tutorial Section C (Dunnett-F)
Sample 1	Control	Control	Control
Sample 2	0.4010	0.6742	0.2030
Sample 3	0.1905	0.4637	0.0451
Sample 4	0.1378	0.2005	-0.1654
Regression R ²	0.016606	0.055351	0.013746

Dunnett Critical Value 2.403

Key:

Sample 1: Control group

Sample 2: Group given interfaces with inconsistent color and parallelism

Sample 3: Group given interfaces with inconsistent color/tense & white space

Sample 4: Group given interfaces with inconsistent tense & white space/parallelism

Tutorial Section A: Control section

Tutorial Section B: Test for color and tense

Tutorial Section C: Test for white space and parallelism

Discussion: Initially, the regression analysis was done, primarily because it gives a good rough estimate about how groups compare and is much easier to program than the

Dunnett's test (Freeman & Godfrey, 1991). Once the low R^2 results were seen, the Dunnett's test--which is much more powerful than the regression analysis--was programmed. Once again, the high Critical Dunnett value offered no insight, but further testing did show that five-question tests for sections B and C were significantly different. This difference could mean that the model used was inadequate for distinguishing feature effects because the tests were not representative. Care should be exercised in any future testing to ensure each test does not differ to a significant level.

The final data collection efforts were using questionnaires. These questionnaires were used to analyze specific design considerations. Once the questionnaire results were electronically scanned and entered into a data file, the researcher developed a statistical check to show the Fisher's test results and Dunnett's T test results. See Appendix C for a copy of the SAS program which was used.

Table 4
Dunnett's T Test
(Comparison Against a Control Group)

$\alpha = 0.05$ Conf = 0.95 df = 73

	Sample 1	Sample 2	Sample 3	Sample 4	Critical T	F	PR>F
Accuracy	Control	.6349	.4361	.1203	2.404	.6684	.5742
Clarity	Control	1.825	1.088	1.088	2.403	5.7412	.0005
Coherence	Control	1.5714	1.5188	.9925	2.403	3.0385	.0540
Completeness	Control	.7419	.3734	.1629	2.403	16.8200	.0001
Continuity	Control	1.0317	.8446	.2607	2.404	10.1517	.0001
Distinctiveness	Control	1.3058	.9373	.7268	2.403	1.5048	.2289

Discussion: One of the positive statistical results is that the questionnaire used did show that it was valid (i.e., that the groups of items designed to assess these different qualities of the design did in fact measure those qualities) because the critical t score was nearly the same for each feature. On the negative side, the raw data did not show a significant difference among the participants on the basis of these design considerations. As the next paragraph shows, the whole experimental process is to probably the reason for these results

John H. Holland, one of the founders of complexity research, explained that Fisher's math, originally designed to look at natural selection, considered one gene at a time, as if each gene's contribution to the organism's survival remained totally independent of all the other genes (Waldrop, 1993). But in real life, each gene must work as part of a larger code. In essence, then, in isolation one gene does not predict much. Likewise, in

this study, in isolation one feature does not predict much. Color alone is nothing without the words. Parallelism and white space alone do not comprise the entire prototype. The interaction among the features is what makes the system complex, not linear. But Fisher's math is designed for linear models. Although both Fisher and this study were originally designed to seek the perfect combination (of genes or features), with complex systems, this is not done through straight linear analysis. To effectively analyze these statistics, a model would have to be developed on a neural network or similar device which can use non-linear analysis to see exactly how much each feature contributes to the overall whole. The more features which are identified, the stronger the relationships and the better the overall analysis, although with too many features (more than about 20), the program would be too cumbersome.

That is not to say that this study was done in vain. This researcher would conclude several things based on the numbers presented in the previous tables. Three possible explanations for the insignificant numbers are: 1) the sample was too small to yield significant results, 2) the sample was inappropriate for the analysis, or 3) the statistics used were not powerful enough to capture the subtle differences among the groups. This is an important point, because much of management science's research is done using linear models, with the t-test and Fisher's test being two of the most widely used analysis tools. If, in fact, Fisher's test can only be translated to linear models, then many management studies would be in error by using those statistics to analyze the complex interactions which are present when trying to model human behavior. The following section gives more detail about how this complexity might have affected this and possibly other management studies.

Complexity

The sciences of complexity are emerging as some of the most important global research being done today (Casti, 1994). Premiere Nobel laureates and highly acclaimed scientists have been tackling complexity since the mid-80s, yet little is really known about the field. Basically, complexity is the edge between where systems are orderly and chaotic. Scientists are discovering that complex systems have different properties than more simple systems. According to John Holland, one of the founders of these theories, complex adaptive systems have different properties (Waldrop, 1993). They:

- 1) are "networks of many 'agents' acting in parallel,"
- 2) have many levels of organization, with agents at any one level serving as the building blocks for agents at a higher level,
- 3) anticipate the future, and
- 4) have many niches, each one of which can be exploited by an agent adapted to fill that niche.

Agents might be anything from nerve cells in a brain, households in an economy, or whole nations in international trade. Each agent must interrelate with other agents in the environment. Therefore, the environment is constantly changing. The control of these complex adaptive systems is highly dispersed. Since there is no central control, if there is coherent behavior in the system it is due to competition and cooperation among the agents.

According to Mitchell Waldrop (1992: 232), "Life is based to an incredible degree on its ability to process information." Complex adaptive systems are hard to analyze with standard mathematics. Most conventional techniques, like calculus or linear analysis are very well suited to linear models. Complex systems are nonlinear. Researchers must therefore develop math and computer simulation techniques that analyze internal models and the web of interactions between multiple agents.

Another of complexity's founding fathers, William Arthur, says that there are three basic approaches to problem solving: cost-benefit, full institutional-political analysis, and equilibrium versus complexity (Waldrop, 1993). Cost-benefit analysis has proven useful in many situations. In this case, the costs of doing business are high in the information resource management field, with billions of dollars spent each year. The benefits of a thorough study would be substantial. If people are able to learn more rapidly and with higher retention, then each effort would be worthwhile depending on the size of the market.

Full institutional-political analysis means that all the players are considered, for example the government heads, the employees, the instructors and students, the software developers and maintainers, and the government contractors would each have their own slant on the situation. Once all the considerations were on the table, then an analysis could be done to see which perspectives to eliminate and which to exploit.

With equilibrium versus complexity, equilibrium would ascribe to the notion that things are dichotomous. Either a feature affects learning or it does not. Color consistency does affect learning while parallelism does not, etc. Complexity says that different features are complexly intertwined with other features and that to understand how these features affect learning and retention; the whole is greater than the sum of its parts.

Complex adaptive systems, such as people trying to learn about LISTSERV using an expert-system prototype, need appropriate models to better understand their behavior. The fact that people are complex adaptive systems makes researching them much more difficult than merely studying their behavior (Wenger & Spyridakis, 1989). The following section highlights some of the difficulties this researcher encountered during the administration of this experiment.

Experimental Problems

Several problems emerged when administering these prototypes:

- The expert system shell failed to fire on four occasions. Three of the failed attempts resulted in the test administrator moving the participants to a new location. One of the failures resulted in the participant's results being eliminated from consideration.
- The number of participants varied in each group testing on each prototype, forcing the researcher to use less powerful statistics when making the comparisons.
- The room was not soundproofed -- and therefore extraneous noises were present -- causing some participants to lose concentration, thus affecting their timing.
- Groups of participants were tested simultaneously and then questionnaires were passed out. Once the first participants began receiving their questionnaires, the others seemed to finish their sections thinking that they were holding up the process. It would have been more scientific if each participant had been screened separately, but due to time constraints, this was not feasible.
- Sample selection for this study was to use a convenience sample. an insufficient number to achieve significant statistical results. Ideally, follow-on studies would need at least 30, and probably a much higher number, to be able to ascribe characteristics to any one of the features.
- This model has outstanding internal validity and the following features are rated as good: history, maturation, testing, instrumentation, regression, selection, mortality, interaction of selection and maturation, etc. External validity was not rated as good, however. Because of the scope of this study, this was not determined to be significant (Wenger & Spyridakis, 1989). Future studies could compensate for this external validity deficiency by using a random sample rather than a convenience sample.
- A total of six groups were used, each containing 12-15 subjects, which allowed for

several free terminals to be available in case one of the computers or programs malfunctioned. This was beneficial on three occasions when subjects were unable to access the program and were subsequently moved to different terminals and were able to complete the course.

It is this researcher's conclusion that the sample used was somewhat poor because of the subjects' ability to compensate for the confounds presented. It is not determined from this study whether a different age group or educational level would be more appropriate; however, it is reasonable to expect different results from different levels of reading ability.

Summary

Although this model was originally supposed to measure the linear relationship among four features which were extracted from the human-computer interface, as in most studies the problem became complex. The sciences of complexity are teaching us about the nature of interactions, showing that they cannot be conventionally analyzed and measured. The failure of this model to test whether the four selected features have an impact on a user's accurate retention of information, satisfaction level, and speed is not a problem with the model so much as with the tools presently available to analyze the raw data. Without advanced nonlinear procedures, this lack could be a major setback in researching complex problems such as this one.

V. Conclusions

Introduction

In this final chapter, the researcher provides an encapsulated overview, specific conclusions, recommendations, and suggested future research areas. Although this pilot study was not conclusive, several key observations were noted and should be considered when making similar studies.

Overview

This study used a pretest-posttest control group experimental design to test the effect of consistency on speed, retention, and user satisfaction. Four prototype tutorial interfaces were altered for color, white space, verb tense, and parallelism and the researcher tested four groups of people using timing, tests, and questionnaires. The literature review showed the interest which is being shown in the interface design area. Billions of dollars are spent each year in interface design and modelling, and yet many people still complain of cumbersome software. The goal of this study was to find a reason why the interfaces are still cumbersome. Are there specific features of the interface which can be improved for heightened productivity? The method used provided a sound pilot study for this research area, but several problems did arise which must be addressed in future undertakings. The results of these tests were statistically insignificant, but the researcher uncovered several possible reasons for this situation and addressed how future studies could be improved.

The remainder of this chapter will be devoted to looking at specific conclusions drawn from this research with an eye toward improving any future studies.

Conclusions

The first conclusion is that the complexity of the project should not be underestimated. When originally conceived, this thesis project appeared to be a straightforward experiment in which specific features of a human-computer interface could be manipulated to alter people's ability to perform. What was forgotten was that it was actually a management study in which people were the key ingredient, not the machines. In experimentation, the researcher tries to eliminate as many confounds as possible. This researcher thought that the pretest-posttest control group experimental design would overcome the other confounds in such a way that specific hard evidence would result. All along, each professor had been emphasizing the importance of sound methodology. Specific design considerations were weighed and the pretest-posttest control group design offered the most power. The statistics used were the most powerful available for this type of study. What this researcher did not realize is that people are the confounds. People read at different levels, people compensate for things, people look around them and hurry up when others are already finished. People are complex adaptive systems which are nonlinear and cannot be properly analyzed by traditional engineering and scientific methods.

The second conclusion is that the human-computer interface is not entirely understood and that future studies should concentrate on one feature at a time exhaustively and then look at the overall picture. Although all the confounds and complexity could not be eliminated, more accurate testing would most likely produce more significant results. This methodology tried to capture four different features with two interactions. Other similar studies used hours of reading (Wenger & Spyridakis, 1989; Schriver, 1989; and Fisk & Jones 1992). If color consistency were to be tested,

then the researcher could use already existing systems, such as computer aided courses, which have hours of material. The 15-minute presentation in this thesis probably was not sufficient for testing retention and speed. The subjects should have been tested in separate rooms so they could not see what others were doing. The instructor should not have been present to answer questions, because then the timing is altered and concentration is broken. The best way to do that would be to videotape the performance. All the other observations (i.e. test results and timing) could be captured in the programming.

The final conclusion is that better modeling techniques and statistical analysis tools are needed to understand this type of research (Wenger & Spyridakis, 1989). The low R^2 and Dunnett test results are indicative that there was some confound in this study. It is this researcher's opinion that the whole approach is inappropriate. Neural networks could provide the nonlinear look at the features. Entire courses are taught in pattern recognition and the like, but few management people attend them. Without a good understanding about why people are complex systems, management studies can fall short of proper analysis. It is no wonder that no feature other than goal-setting ever achieved statistical significance in keeping people happy on the job. The analyses done have all been using linear tools. Looking at human motivations and satisfaction levels is complex and needs better models and more powerful analytical tools if anything meaningful is to result. Those frameworks do not exist today, but much research is being done at places like the Santa Fe Institute. Managers and researchers in management must realize the importance of using proper techniques when developing studies and experiments, or they are likely to jump to false conclusions or, as this study did, have no significant results.

Recommendations

1. Develop better models for testing complex problems.
2. Develop better statistical tools for analyzing these problems.

3. Start small and then build on the research.
4. Do not get too bogged down on those small pieces that the researcher loses sight of the overall, sometimes complex, picture.
5. Broaden field of research to include science, mathematics, computer science, and linguistics, to name a few. Without cross-pollination of ideas and research, people tend to repeat efforts, make the same mistakes, and lose sight of the big picture.

Future Research

Future research is needed to ascertain which features play critical roles in accurate retention of information, user level of satisfaction, and speed of processing. This researcher feels that color would be the easiest measurable area, since it is the easiest to manipulate on the screen. Based on observations during the testing of this thesis, this researcher would also recommend that white space and parallelism be tested, but using much longer material. If time had been sufficient to allow this thesis to fully develop a three- or four-day course on an Introduction to Computer Technology, then more accurate results would have been yielded. Testing of the person's reading level should have been done first, however. One researcher felt that perhaps elementary school children should be tested. Age and reading levels would appear to matter because graduate students should have better faculty for adjusting to poor interfaces than those with little experience. Finally, to do an effective study, the recommendations presented in the previous section must be considered. People are confounds in and of themselves and must be considered as such when developing other models in the future.

Summary

This thesis looked at four features and two interactions of the human-computer interface. While no significant statistical results resulted, important advances were made in understanding the scientific method, experimentation, and the effect of these features.

The differences among the four groups cannot be discounted when analyzing the figures and the complexity of the problem cannot be underestimated.

The features of the human-computer interface interact with one another, but to what extent any one or combination of features affect the overall performance of the subjects is impossible to ascertain using conventional linear methods. The sciences of complexity can account for the poor results of this study, since the entire effort made the erroneous assumption that the features independently affected the subjects' comprehension, satisfaction level, and speed. For managers and researcher, answering the question "Is this a linear relationship?" should be the first step in undertaking a research problem. If the answer is that the system is not linear, then more up-front consideration for modeling and analyzing the problem must be done for the project to yield any significant results.

Appendix A: Prototype Version #1 - 4

STARTING FILE: VIA.KBS

!EXECUTE;

BKCOLOR = 0;

RUNTIME;

ACTIONS

WOPEN 1,1,1,20,77,0

ACTIVE 1

COLOR = 15

DISPLAY"

LISTSERV: A Powerful Mail Manager

This article describes the LISTSERV program, which is used by BITNET and other systems to add and delete customers from mailing lists, among other functions.

The article discusses several areas:

- Background and history of LISTSERV
- Problems associated with LISTSERV
- How To section which outlines some common commands

You will be tested on each of the sections.

Make sure you pay particular attention as you are reading.

You do not have the option to reread sections once you have gone to the next screen."

COLOR = 12

DISPLAY" Press any key to continue~"

WCLOSE 1

WOPEN 1,1,1,3,77,3

ACTIVE 1

COLOR = 1

DISPLAY" BACKGROUND AND HISTORY (THIS IS TESTABLE)"

WOPEN 2,5,1,16,77,0

ACTIVE 2

COLOR = 15

DISPLAY"

When people are first interested in exploring the Internet, they often do not know other people already on the net. One way to meet people is to put your name on mailing lists based on an area of interest. If you are interested in science, for example, many groups are active on Internet. Each group is different, and often the group's name has nothing to do with its conversations. It is, therefore, advisable to subscribe to a group and read the messages for a week or so before writing your own commentary."

COLOR = 12

DISPLAY" Press any key to continue~"

COLOR = 15

CLS

DISPLAY"

To subscribe to a mailing list, two options can be used (Levine & Baroudi, 1993). The first way is for someone to send a message requesting to be put on or taken off a particular list to a list maintainer. This list maintainer, in turn, updates the list manually.

This method is ineffective because the list maintainers usually hold real jobs somewhere and often do not update the lists in a timely manner.

The second way is for a computer to automatically update the requests. LISTSERV is an abbreviation for List Server, a family of programs developed by IBM which use automatic mail managers."

```
COLOR = 12
```

```
DISPLAY" Press any key to continue~"
```

```
COLOR = 15
```

```
CLS
```

```
DISPLAY"
```

According to Crossing the Internet Threshold, LISTSERV is a 'BITNET service providing distributed messages that form conferences and allow the archiving of files and messages which can be searched and retrieved.' In other words, it is a software program which runs on a BITNET computer and handles all the administrative functions of a list, such as subscribing people or listing people on the net (LaQuey, 1992)."

```
COLOR = 12
```

```
DISPLAY" Press any key to continue~"
```

```
COLOR = 15
```

```
CLS
```

```
DISPLAY"
```

LISTSERV came about when people using BITNET

realized just how much the size and number of groups were growing. At first only those who were directly connected to BITNET and its IBM mainframe computers were able to use LISTSERV (Levine & Baroudi, 1993). Today, all Internet subscribers have access to this service.

LISTSERV is a somewhat user-unfriendly, but it is worth the extra effort to learn because of its outstanding ability to handle large mailing lists quickly. For instance, LISTSERV can send mail to 1000 addresses in about 5 minutes. The Internet sendmail program would take over 1 hour to perform the same task (Levine & Baroudi, 1993)."

```
COLOR = 12
```

```
DISPLAY" Press any key to continue~"
```

```
CLS
```

```
COLOR = 12
```

```
DISPLAY"
```

```
READY FOR YOUR FIRST TEST? Press any key to begin.~"
```

```
CLS
```

```
CHAIN i:\firstt1;
```

```
NEW FILE: FIRSTT1
```

```
!EXECUTE;
```

```
BKCOLOR = 0;
```

```
RUNTIME;
```

```
ENDOFF;
```

```
ACTIONS
```

```
WOPEN 1,1,1,9,77,0
```

ACTIVE 1
COLOR = 15
DISPLAY"

INSTRUCTIONS:

For multiple choice: Use arrow keys to highlight answer,
then <Enter>. For short answer: Type answer at cursor,
use UNDERSCORE between words, and WATCH YOUR SPELLING!
<Backspace> to fix typos. Case doesn't matter. Use
acronyms if you can."

WOPEN 2,11,1,9,77,7

ACTIVE 2

COLOR = 1

FIND correct1

FIND correct2

FIND correct3

FIND correct4

FIND correct5

SAVEFACTS i:\temp1

CHAIN i:\b11.kbs

;

!=====RULES BLOCK=====

RULE 1

IF answer1 = b

THEN correct1 = yes

ELSE correct1 = no;

RULE 2

IF answer2 = BITNET

THEN correct2 = yes

ELSE correct2 = no;

RULE 3

IF answer3 = c

THEN correct3 = yes

ELSE correct3 = no;

RULE 4

IF answer4 = d

THEN correct4 = yes

ELSE correct4 = no;

RULE 5

IF answer5 = a

THEN correct5 = yes

ELSE correct5 = no;

!=====QUESTIONS BLOCK=====

ASK answer1: "LISTSERV is a BITNET service which provides distributed messages
that

- a. Allow_archiving_of_files_and_system_implementation,
- b. Form_conferences_and_allow_file_archiving,
- c. List_people_on_the_net,
- d. None_of_the_above";

CHOICES answer1: a, b, c, d;

ASK answer2: "Initially only those who were directly connected to < ? >
were able to use LISTSERV?";

ASK answer3: "Who can use LISTSERV today?

- a. Anyone_with_a_computer,
- b. Only_those_who_own_IBM_compatible_machines,
- c. Any_Internet_subscriber,
- d. Only_AFIT_students";

CHOICES answer3: a, b, c, d;

ASK answer4: "About how long does it take Internet to send mail to 1000 addresses?

- a. 2_days,
- b. 1000_minutes,
- c. 3_minutes,
- d. 1_hour";

CHOICES answer4: a, b, c, d;

ASK answer5: "About how long does it take LISTSERV to send mail to 1000 addresses?

- a. 5_minutes,
- b. 3_minutes,
- c. 2_days,
- d. 1_hour";

CHOICES answer5: a, b, c, d;

NEW FILE: BL1.KBS

!EXECUTE;

BKCOLOR = 1;

RUNTIME;

ACTIONS

WOPEN 1,3,10,13,60,5

COLOR = 1

ACTIVE 1

LOADFACTS i:\temp1

PRINTON

FIND Q1

SHIP Q1, k:\%USID%.res

FIND Q2

SHIP Q2, k:\%USID%.res

FIND Q3

SHIP Q3, k:\%USID%.res

FIND Q4

SHIP Q4, k:\%USID%.res

FIND Q5

SHIP Q5, k:\%USID%.res

PRINTOFF

RULE 1

IF correct1 = yes

THEN Q1 = yes

DISPLAY "B11.kbs: You have answered question 1 correctly."

ELSE Q1 = no

DISPLAY "B11.kbs: You have answered question 1 incorrectly.";

RULE 2

IF correct2 = yes

THEN Q2 = yes

DISPLAY "You have answered question 2 correctly."

ELSE Q2 = no

DISPLAY "You have answered question 2 incorrectly.";

RULE 3

IF correct3 = yes

THEN Q3 = yes

DISPLAY "You have answered question 3 correctly."

ELSE Q3 = no

DISPLAY "You have answered question 3 incorrectly.";

RULE 4

IF correct4 = yes

THEN Q4 = yes

DISPLAY "You have answered question 4 correctly."

ELSE Q4 = no

DISPLAY "You have answered question 4 incorrectly.";

RULE 5

IF correct5 = yes

THEN Q5 = yes

DISPLAY "You have answered question 5 correctly.

Please wait for instructions to continue.~"

ELSE Q5 = no

DISPLAY "You have answered question 5 incorrectly.

Please wait for instructions to continue.~";

NEW FILE: NORTON UTILITY PACKAGE

!This package captures timing and test results of each machine by the person's login

@echo off

verify on

cls

ncc /n /start:1

ncc /n /start:2

rem ...Insert call to VPX program, part #1

r:\vpexpert\vpk k:\vpctest\v1a.kbs

echo. >>c:\%UID%.dat

echo. >>c:\%UID%.dat

echo Part #1 of the test was finished in: >>c:\%UID%.dat

ncc /n /1 /stop:2 >>c:\%UID%.dat

ncc /n /start:3

rem ...Insert call to VPX program, part #2

r:\vpexpert\vpk k:\vpctest\v1c.kbs

echo. >>c:\%UID%.dat

echo. >>c:\%UID%.dat

echo Part #2 of the test was finished in: >>c:\%UID%.dat

ncc /n /1 /stop:3 >>c:\%UID%.dat

ncc /n /start:4

rem ...Insert call to VPX program, part #3

r:\vpexpert\vpk k:\vpctest\v1d.kbs

echo. >>c:\%UID%.dat

echo. >>c:\%UID%.dat

echo Part #3 of the test was finished in: >>c:\%UID%.dat

ncc /n /1 /stop:4 >>c:\%UID%.dat

echo. >>c:\%UID%.dat

echo. >>c:\%UID%.dat

echo The entire test was finished in: >>c:\%UID%.dat

ncc /n /1 /stop:1 >>c:\%UID%.dat

echo. >>c:\%UID%.dat

echo. >>c:\%UID%.dat

echo The test results from part #1: >>c:\%UID%.dat

type i:\t1res.kbs >>c:\%UID%.dat

echo. >>c:\%UID%.dat

echo. >>c:\%UID%.dat

echo The test results from part #2: >>c:\%UID%.dat

type i:\t2res.kbs >>c:\%UID%.dat

echo. >>c:\%UID%.dat

echo. >>c:\%UID%.dat

echo The test results from part #3: >>c:\%UID%.dat

type i:\t3res.kbs >>c:\%UID%.dat

echo. >>c:\%UID%.dat

echo. >>c:\%UID%.dat

NEW FILE: VIC.KBS

!EXECUTE;

RUNTIME;

BKCOLOR = 0;

ACTIONS

WOPEN 1,1,1,3,77,5

ACTIVE 1

COLOR = 1

DISPLAY" PROBLEMS (THIS IS TESTABLE)"

WOPEN 2,5,1,16,77,0

ACTIVE 2

COLOR = 15

DISPLAY"

As most A.F.I.T. students know, mailing lists sometimes cause your computer to be loaded with many unwanted messages. Because of its incredible power, LISTSERV can also contribute to this problem. Some people like to receive junk mail and others do not. If you are the type of person who gets depressed when 3000 messages are not queued and waiting after a 3-day weekend, then subscribe to multiple groups. Otherwise, it is probably advisable to join one group at a time to see if they are in line with your interests."

COLOR = 12

DISPLAY" Press any key to continue~"

COLOR = 15

CLS

DISPLAY"

Another problem LISTSERV has is processing requests for changes of address. Many people change jobs. When this happens, often people's e-mail addresses also change. Because most people are too caught up in moving, they do not consider that LISTSERV does not know about their e-mail address change. Once settled, they try to send something to their group and realize their address is wrong.

It should be an easy thing to change addresses, but because of the way LISTSERV operates, it does not recognize the changes automatically. The people must write to the keeper of the list with an address change and explanation of the situation."

COLOR = 12

DISPLAY" Press any key to continue~"

COLOR = 15

CLS

DISPLAY"

Because of this problem, sometimes messages are also rejected by LISTSERV and error messages are sent to other subscribers.

This can become irritating when several people move at once."

COLOR = 14

DISPLAY"

LISTSERV changes names easily, however. Just send another subscribe message with the correct spelling"

COLOR = 15

DISPLAY"

Something more of an embarrassment to the originator, rather than a problem inherent in the LISTSERV, is that beginners often send their intention to join a group directly to the group, rather than to the LISTSERV. Using LISTSERV also requires that you know the addresses, which differ from regular Internet addresses (LaQuey, 1992)."

COLOR = 12

DISPLAY" Press any key to continue~"

COLOR = 15

CLS

DISPLAY"

Care should be exercised when responding to someone on the list. Some lists have automatic defaults so your response only goes to the original author, whereas others send your reply to the entire group."

COLOR = 14

DISPLAY"

Make sure to change the address line if you do not want your message to go to the default (LaQuey, 1992)."

COLOR = 12

DISPLAY" Press any key to continue~"

CLS

COLOR = 15

DISPLAY"

One of the biggest obstacles to overcome when trying to use the LISTSERV application is determining what lists to explore.

Volumes have been written about the different lists, but groups come and go so quickly that it is tough to find that perfect list."

COLOR = 14

DISPLAY"

You can get the current list-of-lists by sending an e-mail message to `LISTSERV@BITNIC.bitnet` with the command `LIST GLOBAL` in the body (Tennant, Ober, & Lipow, 1993)."

COLOR = 15

DISPLAY"

WARNING: It is a large list."

COLOR = 12

DISPLAY" Press any key to continue~"

COLOR = 15

CLS

DISPLAY"

The public Internet mailing lists are also regularly updated and posted to Usenet newsgroups such as `news.lists`, `news.announce`, `newusers`, etc. (Hahn & Stout, 1994). You can also download the list by FTPing `rtfm.mit.edu` with Anonymous. The files are `part1`, `part2`, etc. in the directory `/pub/usenet/news.answers/mail/mailling-lists`.

Another good list-of-lists is managed by Kent State University. It lists scholarly and academic subject lists. Once again, use the Anonymous FTP from `ksuvxa.kent.edu`. Change directory (CD) to `library` and download the

file acadlist.readme."

COLOR = 12

DISPLAY" Press any key to continue~"

COLOR = 14

CLS

DISPLAY"

You can usually download lists by signing in as Anonymous."

COLOR = 15

DISPLAY"

Once you sign in, use your e-mail password to identify yourself.

Often the groups monitor who is permitted to join the group by asking questions. Sometimes the groups are moderated to keep the volume of traffic at a manageable level (Levine & Baroudi, 1993)."

COLOR = 14

DISPLAY"

The best advice is to get on the system and surf or look around at different groups."

COLOR = 12

DISPLAY" Press any key to continue~"

CLS

COLOR = 12

DISPLAY"

Ready for a test? Press any key to start.~"

CHAIN i:\tone1.kbs;

NEW FILE: TONE1.KBS

!EXECUTE;
BKCOLOR = 0;
RUNTIME;
ENDOFF;
ACTIONS

WOPEN 1,1,1,9,77,0
ACTIVE 1
COLOR = 15
DISPLAY"

INSTRUCTIONS:

For multiple choice: Use arrow keys to highlight answer,
then <Enter>. For short answer: Type answer at cursor,
use UNDERSCORE between words, and WATCH YOUR SPELLING!
<Backspace> to fix typos. Case doesn't matter. Use
acronyms if you can."

WOPEN 2,11,1,9,77,7
ACTIVE 2
COLOR = 1
FIND correct1
FIND correct2
FIND correct3
FIND correct4
FIND correct5
SAVEFACTS i:\tempdata1
CHAIN i:\resone1.kbs;

!=====RULES BLOCK=====

RULE 1

IF answer1 = c

THEN correct1 = yes

ELSE correct1 = no;

RULE 2

IF answer2 = b

THEN correct2 = yes

ELSE correct2 = no;

RULE 3

IF answer3 = d

THEN correct3 = yes

ELSE correct3 = no;

RULE 4

IF answer4 = c

THEN correct4 = yes

ELSE correct4 = no;

RULE 5

IF answer5 = a

THEN correct5 = yes

ELSE correct5 = no;

!=====QUESTIONS BLOCK=====

ASK answer1: "How do you change you name in LISTSERV?"

- a. Change_name_with_correct_spelling,
- b. Delete_original_and_reregister,
- c. Subscribe_with_correct_spelling,
- d. Tell_list_keeper_to_change_it";

CHOICES answer1: a, b, c, d;

ASK answer2: "What do you do if you do not want your message to go to the default?"

- a. Change_the_automatic_default,
- b. Change_the_address_line,
- c. Change_the_query_line,
- d. Use_LISTSERV_function_key";

CHOICES answer2: a, b, c, d;

ASK answer3: "How can you get the current list-of-lists?"

- a. Query_BITNET @_LISTSERV &_use_GET_LIST,
- b. Query_LISTSERV @_BITNET &_use_LIST_GLOBAL,
- c. Query_LISTSERV @_BITNET.bitnic &_use_GET_LIST,
- d. Query_LISTSERV @_BITNIC.bitnet &_use_LIST_GLOBAL";

CHOICES answer3: a, b, c, d;

ASK answer4: "What login can you use to download lists?"

- a. Guest,
- b. Your_e-mail_password,
- c. Anonymous,
- d. Readme";

CHOICES answer4: a, b, c, d;

ASK answer5: "What is it called when you look around at different groups?"

- a. Surfing,
- b. Skating,

- c. Jumping,
- d. Moderating";

CHOICES answer5: a, b, c, d;

NEW FILE: RESONE1.KBS

```
!EXECUTE;
BKCOLOR = 1;
RUNTIME;
ACTIONS
  WOPEN 1,3,10,13,60,5
  COLOR = 1
  ACTIVE 1
! PRINTON
LOADFACTS i:\tempdata1
SAVEFACTS c:\%UID%.dbt
  FIND Q1
  FIND Q2
  FIND Q3
  FIND Q4
  FIND Q5
! PRINTOFF
RULE 1
IF correct1 = yes
THEN Q1 = yes
  DISPLAY "Resone1.kbs: You have answered question 1 correctly."
ELSE Q1 = no
```

DISPLAY "Resone1.kbs: You have answered question 1 incorrectly.";

RULE 2

IF correct2 = yes

THEN Q2 = yes

DISPLAY "You have answered question 2 correctly."

ELSE Q2 = no

DISPLAY "You have answered question 2 incorrectly.";

RULE 3

IF correct3 = yes

THEN Q3 = yes

DISPLAY "You have answered question 3 correctly."

ELSE Q3 = no

DISPLAY "You have answered question 3 incorrectly.";

RULE 4

IF correct4 = yes

THEN Q4 = yes

DISPLAY "You have answered question 4 correctly."

ELSE Q4 = no

DISPLAY "You have answered question 4 incorrectly.";

RULE 5

IF correct5 = yes

THEN Q5 = yes

DISPLAY "You have answered question 5 correctly."

Please wait for instructions to continue.~"

ELSE Q5 = no

DISPLAY "You have answered question 5 incorrectly.

Please wait for instructions to continue.~";

NEW FILE: VID.KBS

!EXECUTE;

RUNTIME;

BKCOLOR = 0;

ACTIONS

WOPEN 1,1,1,3,77,5

ACTIVE 1

COLOR = 1

DISPLAY" HOW TO (THIS IS TESTABLE)"

WOPEN 2,5,1,19,77,0

ACTIVE 2

COLOR = 14

DISPLAY"

SUBSCRIBING TO A LIST:"

COLOR = 15

DISPLAY"

SUB <LIST NAME <YOUR NAME> or

SIGNUP <LIST NAME> <Your Name>"

COLOR = 15

DISPLAY"

Description: Subscribing to a list is quite simple. In the body

just type either the sub or signup command as

shown above. If you want to try a Listserv, but

don't feel like committing to the list today, try

PRACTICE Listserv by following the example below.

Example: To: INTERNET'LISTSERV@ULKYVM.BITNET'

Subj: Your choice

SUB GSS-L Susan Stewart

Ctrl/Z (Use your real name, not email address)"

COLOR = 12

DISPLAY" Press any key to continue~"

CLS

COLOR = 14

DISPLAY"

UNSUBSCRIBING TO A LIST:"

COLOR = 15

DISPLAY"

UNSUB <LIST NAME> or

SIGNOFF <LIST NAME>

Description: The above two commands can be used to unsubscribe

you to a list. Some groups will send you a welcome message, which reminds you how to unsubscribe. Since each group has a different level of involvement by its members, you may have to join several lists before you get the right mix for you. If you decide you've had enough and want to cancel all your groups, use

UNSUB * (you don't need your name or the groups)

Example: UNSUB GSS-L"

COLOR = 12

DISPLAY" Press any key to continue~"

CLS

COLOR = 14

DISPLAY"

SUSPENDING MAIL:"

COLOR = 15

DISPLAY"

SET <LIST NAME> NOMAIL

Description: When you want to temporarily suspend your messages, use the nomail command. This will allow you to go on vacation and not have a huge backlog when you return. You can also send NOFiles if you don't want any non-email files sent.

Example: SET GSS-L NOMAIL

(The computer automatically reads your email address from the message you are sending)"

COLOR = 12

DISPLAY" Press any key to continue~"

CLS

COLOR = 14

DISPLAY"

RESUMING MAIL:"

COLOR = 15

DISPLAY"

SET <LIST NAME> MAIL"

COLOR = 15

DISPLAY"

Description: This is the opposite of the NOMAIL feature. By sending this message, you cancel your previous NOMAIL message and resume mail. Again, the computer will automatically read your email address.

Example: SET GSS-L MAIL"

COLOR = 12

DISPLAY" Press any key to continue~"

CLS

COLOR = 14

DISPLAY"

VIEWING THE SUBSCRIBER LIST:"

COLOR = 15

DISPLAY"

REV <LIST NAME>"

COLOR = 15

DISPLAY"

Description: This command returns a complete list of subscribers and their email addresses. You have several options you can use such as how the list is sorted, etc. Send the command 'info refcard' for information about the current options. You can also use the command 'rev help-net' for information.

Example: REV GSS-L"

COLOR = 12

DISPLAY" Press any key to continue~"

CLS

COLOR = 14

DISPLAY"

RETRIEVING FILES:"

COLOR = 15

DISPLAY"

INDEX <LIST NAME> then GET <FILE NAME> <FILE TYPE>

Description: Use this command when you want to retrieve one of the

files you identified with INDEX. You may want to log

on manually right after you request a file. Files are

usually sent to you within five minutes or so and

sometimes they are rather large. Since you probably

can't read 400K as it scrolls by, extract the message

to a file and download it from your workspace.

Example: INDEX GSS-L

GET FUZZY-LOGIC LOG4312"

COLOR = 12

DISPLAY" Press any key to continue~"

CLS

COLOR = 14

DISPLAY"

GETTING LIST OF TOPICS:"

COLOR = 15

DISPLAY"

INFO?"

COLOR = 15

DISPLAY"

Description: If a topic has additional information available, use the INFO? command to retrieve it. You can then use the INFO command again for specific topics. One handy feature in LISTSERV is that you can send multiple commands in one message. Make sure to use a separate line for each command, however.

Example: To: Internet'listserv@bitnic.bitnet'
Subj: Fuzzy Logic
info database
info refcard
Ctrl/Z"

COLOR = 12

DISPLAY" Press any key to continue~"

CLS

COLOR = 14

DISPLAY"

SEARCHING AN ARCHIVE:"

COLOR = 15

DISPLAY"

INFO DATABASE"

COLOR = 15

DISPLAY"

Description: For more information about doing detailed searches of archived lists, send the message INFO DATABASE to any host which uses LISTSERV software. The

archived files may contain thousands of messages,
so it is best to send keywords. Usually your search
will be done in batch mode, but there are many options
available. INFO DATABASE will give you the most
up-to-date info about performing these searches.

Example: To: Internet'listserv@frenk.bitnet'

Subj: Book

INFO DATABASE"

COLOR = 12

DISPLAY" Press any key to continue~"

CLS

COLOR = 12

DISPLAY"

Ready for another test? Press any key to start.~"

CHAIN i:\ttwo1.kbs;

NEW FILE: TTWO1.KBS

!EXECUTE

BKCOLOR = 0;

RUNTIME;

ENDOFF;

ACTIONS

WOPEN 1,1,1,9,77,0

ACTIVE 1

COLOR = 15

DISPLAY"

INSTRUCTIONS:

For multiple choice: Use arrow keys to highlight answer,
then <Enter>. For short answer: Type answer at cursor,
then use UNDERSCORE between words, and WATCH YOUR SPELLING!
<Backspace> to fix typos. Case doesn't matter. Use acronyms
if you can."

WOPEN 2,11,1,10,77,7

ACTIVE 2

COLOR = 1

FIND correct6

FIND correct7

FIND correct8

FIND correct9

FIND correct10

SAVEFACTS i:\tempdata2

CHAIN i:\restwo1.kbs;

!=====RULES BLOCK=====

RULE 1

IF answer6 = Set_LIST_NAME_MAIL

THEN correct6 = yes

ELSE correct6 = no;

RULE 2

IF answer7 = INFO?

THEN correct7 = yes

ELSE correct7 = no;

RULE 3

IF answer8 = INFO_DATABASE

THEN correct8 = yes

ELSE correct8 = no;

RULE 4

IF answer9 = SUB_LIST_NAME_YOUR_NAME

THEN correct9 = yes

ELSE correct9 = no;

RULE 5

IF answer10 = REV_LIST_NAME

THEN correct10 = yes

ELSE correct10 = no;

!=====QUESTIONS BLOCK=====

ASK answer6: "When you have gone on vacation for a while and return,
how do you resume your mail?";

ASK answer7: "How can you get a list of topics from LISTSERV?";

ASK answer8: "If you want to do a detailed search of archived lists, what
message do you send to a LISTSERV host?

- a. SEARCH_LIST,
- b. INFO_DATABASE,
- c. GET_INFO,
- d. INFO_LIST";

CHOICES answer8: a, b, c, d;

ASK answer9: "How do you subscribe to a list?

- a. SUB_LIST_NAME_YOUR_NAME,

- b. SIGNIN_YOUR_NAME,
- c. SUB_YOUR_NAME,
- d. ENTER_YOUR_NAME_LIST_NAME";

CHOICES answer9: a, b, c, d;

ASK answer10: "How can you view the subscriber list?"

- a. INDEX_LIST_GET_SUBLIST,
- b. FTP_VIEW_LIST_NAME,
- c. REV_LIST_NAME,
- d. VIEW_HELP-NET";

CHOICES answer10: a, b, c, d;

NEW FILE: RESTWO1.KBS

!EXECUTE;

BKCOLOR = 1;

RUNTIME;

ACTIONS

WOPEN 1,3,10,13,60,5

COLOR = 1

ACTIVE 1

! PRINTON

LOADFACTS i:\tempdata2

FIND Q1

FIND Q2

FIND Q3

FIND Q4

FIND Q5

! PRINTOFF

RULE 1

IF correct1 = yes

THEN Q1 = yes

DISPLAY "Restwo1.kbs: You have answered question 1 correctly."

ELSE Q1 = no

DISPLAY "Restwo1.kbs: You have answered question 1 incorrectly.";

RULE 2

IF correct2 = yes

THEN Q2 = yes

DISPLAY "You have answered question 2 correctly."

ELSE Q2 = no

DISPLAY "You have answered question 2 incorrectly.";

RULE 3

IF correct3 = yes

THEN Q3 = yes

DISPLAY "You have answered question 3 correctly."

ELSE Q3 = no

DISPLAY "You have answered question 3 incorrectly.";

RULE 4

IF correct4 = yes

THEN Q4 = yes

DISPLAY "You have answered question 4 correctly."

ELSE Q4 = no

DISPLAY "You have answered question 4 incorrectly.";

RULE 5

IF correct5 = yes

THEN Q5 = yes

DISPLAY "You have answered question 5 correctly.

Please wait for instructions to continue.~"

ELSE Q5 = no

DISPLAY "You have answered question 5 incorrectly.

Please wait for instructions to continue.~";

NEW FILE: V1E.KBS

!EXECUTE;

RUNTIME;

BKCOLOR = 0;

ACTIONS

WOPEN 1,1,1,3,77,5

ACTIVE 1

COLOR = 15

DISPLAY"

Thank you for participating in this thesis project.~";

Appendix A: Prototype Version #2

STARTING FILE: V2A

!EXECUTE;

BKCOLOR = 0;

RUNTIME;

ACTIONS

WOPEN 1,1,1,20,77,0

ACTIVE 1

COLOR = 15

DISPLAY"

LISTSERV: A Powerful Mail Manager

This article describes the LISTSERV program, which is used by BITNET and other systems to add and delete customers from mailing lists, among other functions.

The article discusses several areas:

- Background and history of LISTSERV
- Problems associated with LISTSERV
- How To section which outlines some common commands

You will be tested on each of the sections.

Make sure you pay particular attention as you are reading.

You do not have the option to reread sections once you have gone to the next screen."

COLOR = 12

DISPLAY" Press any key to continue~"

WCLOSE 1

WOPEN 1,1,1,3,77,3

ACTIVE 1

COLOR = 1

DISPLAY" BACKGROUND AND HISTORY (THIS IS TESTABLE)"

WOPEN 2,5,1,16,77,0

ACTIVE 2

COLOR = 15

DISPLAY"

When people are first interested in exploring the Internet, they often do not know other people already on the net. One way to meet people is to put your name on mailing lists based on an area of interest. If you are interested in science, for example, many groups are active on Internet. Each group is different, and often the group's name has nothing to do with its conversations. It is, therefore, advisable to subscribe to a group and read the messages for a week or so before writing your own commentary."

COLOR = 12

DISPLAY" Press any key to continue~"

COLOR = 15

CLS

DISPLAY"

To subscribe to a mailing list, two options can be used (Levine & Baroudi, 1993). The first way is for someone to send a message requesting to be put on or

taken off a particular list to a list maintainer.

This list maintainer, in turn, updates the list manually.

This method is ineffective because the list maintainers usually hold real jobs somewhere and often do not update the lists in a timely manner.

The second way is for a computer to automatically update the requests. LISTSERV is an abbreviation for List Server, a family of programs developed by IBM which use automatic mail managers."

```
COLOR = 12
```

```
DISPLAY" Press any key to continue~"
```

```
COLOR = 15
```

```
CLS
```

```
DISPLAY"
```

According to Crossing the Internet Threshold, LISTSERV is a 'BITNET service providing distributed messages that form conferences and allow the archiving of files and messages which can be searched and retrieved.' In other words, it is a software program which runs on a BITNET computer and handles all the administrative functions of a list, such as subscribing people or listing people on the net (LaQuey, 1992)."

```
COLOR = 12
```

```
DISPLAY" Press any key to continue~"
```

```
COLOR = 15
```

```
CLS
```

DISPLAY"

LISTSERV came about when people using BITNET realized just how much the size and number of groups were growing. At first only those who were directly connected to BITNET and its IBM mainframe computers were able to use LISTSERV (Levine & Baroudi, 1993). Today, all Internet subscribers have access to this service.

LISTSERV is a somewhat user-unfriendly, but it is worth the extra effort to learn because of its outstanding ability to handle large mailing lists quickly. For instance, LISTSERV can send mail to 1000 addresses in about 5 minutes. The Internet sendmail program would take over 1 hour to perform the same task (Levine & Baroudi, 1993)."

COLOR = 12

DISPLAY" Press any key to continue~"

CLS

COLOR = 12

DISPLAY"

READY FOR YOUR FIRST TEST? Press any key to begin.~"

CLS

CHAIN i:\firstt2;

NEW FILE: FIRSTT2.KBS

!EXECUTE;

BKCOLOR = 0;

RUNTIME;

ENDOFF;

ACTIONS

WOPEN 1,1,1,9,77,0

ACTIVE 1

COLOR = 15

DISPLAY"

INSTRUCTIONS:

For multiple choice: Use arrow keys to highlight answer,
then <Enter>. For short answer: Type answer at cursor,
use UNDERSCORE between words, and WATCH YOUR SPELLING!
<Backspace> to fix typos. Case doesn't matter. Use
acronyms if you can."

WOPEN 2,11,1,9,77,7

ACTIVE 2

COLOR = 1

FIND correct1

FIND correct2

FIND correct3

FIND correct4

FIND correct5

SAVEFACTS i:\temp1

CHAIN i:\bl2.kbs;

!=====RULES BLOCK=====

RULE 1

IF answer1 = b

THEN correct1 = yes

ELSE correct1 = no;

RULE 2

IF answer2 = BITNET
THEN correct2 = yes
ELSE correct2 = no;

RULE 3

IF answer3 = c
THEN correct3 = yes
ELSE correct3 = no;

RULE 4

IF answer4 = d
THEN correct4 = yes
ELSE correct4 = no;

RULE 5

IF answer5 = a
THEN correct5 = yes
ELSE correct5 = no;

!=====QUESTIONS BLOCK=====

ASK answer1: "LISTSERV is a BITNET service which provides distributed messages
that

- a. Allow_archiving_of_files_and_system_implementation,
- b. Form_conferences_and_allow_file_archiving,
- c. List_people_on_the_net,
- d. None_of_the_above";

CHOICES answer1: a, b, c, d;

ASK answer2: "Initially only those who were directly connected to < ? >

were able to use LISTSERV?";

ASK answer3: "Who can use LISTSERV today?

- a. Anyone_with_a_computer,
- b. Only_those_who_own_IBM_compatible_machines,
- c. Any_Internet_subscriber,
- d. Only_AFIT_students";

CHOICES answer3: a, b, c, d;

ASK answer4: "About how long does it take Internet to send mail to 1000 addresses?

- a. 2_days,
- b. 1000_minutes,
- c. 3_minutes,
- d. 1_hour";

CHOICES answer4: a, b, c, d;

ASK answer5: "About how long does it take LISTSERV to send mail to 1000 addresses?

- a. 5_minutes,
- b. 3_minutes,
- c. 2_days,
- d. 1_hour";

CHOICES answer5: a, b, c, d;

NEW FILE: BL2.KBS

!EXECUTE;

BKCOLOR = 1;

RUNTIME;

ACTIONS

WOPEN 1,3,10,13,60,5

COLOR = 1

ACTIVE 1

LOADFACTS i:\temp2

PRINTON

FIND Q1

SHIP Q1, k:\%USID%.res

FIND Q2

SHIP Q2, k:\%USID%.res

FIND Q3

SHIP Q3, k:\%USID%.res

FIND Q4

SHIP Q4, k:\%USID%.res

FIND Q5

SHIP Q5, k:\%USID%.res

PRINTOFF

RULE 1

IF correct1 = yes

THEN Q1 = yes

DISPLAY "B12.kbs: You have answered question 1 correctly."

ELSE Q1 = no

DISPLAY "B12.kbs: You have answered question 1 incorrectly.";

RULE 2

IF correct2 = yes

THEN Q2 = yes

DISPLAY "You have answered question 2 correctly."

ELSE Q2 = no

DISPLAY "You have answered question 2 incorrectly.";

RULE 3

IF correct3 = yes

THEN Q3 = yes

DISPLAY "You have answered question 3 correctly."

ELSE Q3 = no

DISPLAY "You have answered question 3 incorrectly.";

RULE 4

IF correct4 = yes

THEN Q4 = yes

DISPLAY "You have answered question 4 correctly."

ELSE Q4 = no

DISPLAY "You have answered question 4 incorrectly.";

RULE 5

IF correct5 = yes

THEN Q5 = yes

DISPLAY "You have answered question 5 correctly."

Please wait for instructions to continue.~"

ELSE Q5 = no

DISPLAY "You have answered question 5 incorrectly."

Please wait for instructions to continue.~";

NEW FILE: NORTON UTILITY PACKAGE

!This package captures timing and test results of each machine by the person's login

@echo off

verify on

cls

ncc /n /start:1

ncc /n /start:2

rem ...Insert call to VPX program, part #1

r:\vpexpert\vpk k:\vpctest\v2a.kbs

echo. >>c:\%UID%.dat

echo. >>c:\%UID%.dat

echo Part #1 of the test was finished in: >>c:\%UID%.dat

ncc /n /1 /stop:2 >>c:\%UID%.dat

ncc /n /start:3

rem ...Insert call to VPX program, part #2

r:\vpexpert\vpk k:\vpctest\v2c.kbs

echo. >>c:\%UID%.dat

echo. >>c:\%UID%.dat

echo Part #2 of the test was finished in: >>c:\%UID%.dat

ncc /n /1 /stop:3 >>c:\%UID%.dat

ncc /n /start:4

rem ...Insert call to VPX program, part #3

r:\vpexpert\vpk k:\vpctest\v2d.kbs

echo. >>c:\%UID%.dat

echo. >>c:\%UID%.dat

echo Part #3 of the test was finished in: >>c:\%UID%.dat

ncc /n /1 /stop:4 >>c:\%UID%.dat

echo. >>c:\%UID%.dat

echo. >>c:\%UID%.dat

echo The entire test was finished in: >>c:\%UID%.dat

ncc /n /1 /stop:1 >>c:\%UID%.dat

echo. >>c:\%UID%.dat

echo. >>c:\%UID%.dat

echo The test results from part #1: >>c:\%UID%.dat

type i:\t1res.kbs >>c:\%UID%.dat

echo. >>c:\%UID%.dat

echo. >>c:\%UID%.dat

echo The test results from part #2: >>c:\%UID%.dat

type i:\t2res.kbs >>c:\%UID%.dat

echo. >>c:\%UID%.dat

echo. >>c:\%UID%.dat

echo The test results from part #3: >>c:\%UID%.dat

type i:\t3res.kbs >>c:\%UID%.dat

echo. >>c:\%UID%.dat

echo. >>c:\%UID%.dat

NEW FILE: V2C.KBS

!EXECUTE;

RUNTIME;

BKCOLOR = 0;

ACTIONS

WOPEN 1,1,1,3,77,5

ACTIVE 1

COLOR = 1

DISPLAY" PROBLEMS (THIS IS TESTABLE)"

WOPEN 2,5,1,16,77,0

ACTIVE 2

COLOR = 13

DISPLAY"

As most A.F.I.T. students know, mailing lists sometimes cause your computer to be loaded with many unwanted messages. Because of its incredible power, LISTSERV can also contribute to this problem. Some people like to receive junk mail and others do not. If you are the type of person who gets depressed when 3000 messages are not queued and waiting after a 3-day weekend, then subscribe to multiple groups. Otherwise, it is probably advisable to join one group at a time to see if they are in line with your interests."

COLOR = 28

DISPLAY " Press any key to continue~"

CLS

COLOR = 10

DISPLAY"

Another problem LISTSERV has is processing requests for changes of address. Many people change jobs. When this happens, often people's e-mail addresses also change. Because most people are too caught up in moving, they do not consider that LISTSERV does not know about their e-mail address change. Once settled, they try to send something to their group and realize their address is wrong."

COLOR = 9

DISPLAY"

It should be an easy thing to change addresses, but because of the way LISTSERV operates, it does not recognize the changes automatically. The people must write to the keeper of the list with an address change and explanation of the situation."

COLOR = 17

DISPLAY" Press any key to continue~"

CLS

COLOR = 14

DISPLAY"

Because of this problem, sometimes messages are also rejected by LISTSERV and error messages are sent to other subscribers.

This can become irritating when several people move at once."

COLOR = 2

DISPLAY"

LISTSERV changes names easily, however. Just send another subscribe message with the correct spelling"

COLOR = 11

DISPLAY"

Something more of an embarrassment to the originator, rather than a problem inherent in the LISTSERV, is that beginners often send their intention to join a group directly to the group, rather than to the LISTSERV. Using LISTSERV also requires that you know the addresses, which differ from regular Internet addresses (LaQuey, 1992)."

COLOR = 3

DISPLAY" Press any key to continue~"

COLOR = 12

CLS

DISPLAY"

Care should be exercised when responding to someone on the list. Some lists have automatic defaults so your response only goes to the original author, whereas others send your reply to the entire group."

COLOR = 13

DISPLAY"

Make sure to change the address line if you do not want your message to go to the default (LaQuey, 1992)."

COLOR = 31

DISPLAY" Press any key to continue~"

CLS

COLOR = 4

DISPLAY"

One of the biggest obstacles to overcome when trying to use

the LISTSERV application is determining what lists to explore.

Volumes have been written about the different lists, but groups

come and go so quickly that it is tough to find that

perfect list."

COLOR = 13

DISPLAY"

You can get the current list-of-lists by sending

an e-mail message to LISTSERV@BITNIC.bitnet with the command

LIST GLOBAL in the body (Tennant, Ober, & Lipow, 1993)."

COLOR = 14

DISPLAY"

WARNING: It is a large list."

COLOR = 10

DISPLAY" Press any key to continue~"

CLS

COLOR = 5

DISPLAY"

The public Internet mailing lists are also regularly updated and posted to Usenet newsgroups such as news.lists, news.announce, newusers, etc. (Hahn & Stout, 1994). You can also download the list by FTPing rtfm.mit.edu with Anonymous. The files are part1, part2, etc. in the directory /pub/usenet/news.answers/mail/mailling-lists.

Another good list-of-lists is managed by Kent State University. It lists scholarly and academic subject lists. Once again, use the Anonymous FTP from ksuvxa.kent.edu.

Change directory (CD) to library and download the
file acadlist.readme."

COLOR = 12

DISPLAY" Press any key to continue~"

COLOR = 15

CLS

DISPLAY"

You can usually download lists by signing in as Anonymous."

COLOR = 14

DISPLAY"

Once you sign in, use your e-mail password to identify yourself.

"COLOR = 15

DISPLAY"

Often the groups monitor who is permitted to join the
group by asking questions. Sometimes the groups are moderated
to keep the volume of traffic at a manageable level (Levine &
Baroudi, 1993)."

COLOR = 15

DISPLAY"

The best advice is to get on the system and surf
or look around at different groups."

COLOR = 30

DISPLAY" Press any key to continue~"

CLS

COLOR = 29

DISPLAY"

Ready for a test? Press any key to start.~"

CHAIN i:\tone2.kbs;

NEW FILE TONE2.KBS

!EXECUTE;

BKCOLOR = 0;

RUNTIME;

ENDOFF;

ACTIONS

WOPEN 1,1,1,9,77,0

ACTIVE 1

COLOR = 15

DISPLAY"

INSTRUCTIONS:

For multiple choice: Use arrow keys to highlight answer, then <Enter>. For short answer: Type answer at cursor, use UNDERSCORE between words, and **WATCH YOUR SPELLING!** <Backspace> to fix typos. Case doesn't matter. Use acronyms if you can."

WOPEN 2,11,1,9,77,7

ACTIVE 2

COLOR = 1

FIND correct1

WCLOSE 1

WCLOSE 2

WOPEN 1,1,1,9,77,1

ACTIVE 1

COLOR = 14

DISPLAY"

INSTRUCTIONS:

For multiple choice: Use arrow keys to highlight answer, then <Enter>. For short answer: Type answer at cursor, use UNDERSCORE between words, and WATCH YOUR SPELLING! <Backspace> to fix typos. Case doesn't matter. Use acronyms if you can."

WOPEN 2,11,1,9,77,7

ACTIVE 2

COLOR = 1

FIND correct2

WCLOSE 1

WCLOSE 2

WOPEN 1,1,1,9,77,5

ACTIVE 1

COLOR = 14

DISPLAY"

INSTRUCTIONS:

For multiple choice: Use arrow keys to highlight answer, then <Enter>. For short answer: Type answer at cursor, use UNDERSCORE between words, and WATCH YOUR SPELLING! <Backspace> to fix typos. Case doesn't matter. Use acronyms if you can."

WOPEN 2,11,1,9,77,7

ACTIVE 2

COLOR = 1
FIND correct3
WCLOSE 1
WCLOSE 2
WOPEN 1,1,1,9,77,3
ACTIVE 1
COLOR = 1
DISPLAY"

INSTRUCTIONS:

For multiple choice: Use arrow keys to highlight answer,
then <Enter>. For short answer: Type answer at cursor,
use UNDERSCORE between words, and WATCH YOUR SPELLING!
<Backspace> to fix typos. Case doesn't matter. Use
acronyms if you can."

WOPEN 2,11,1,9,77,7
ACTIVE 2
COLOR = 1
FIND correct4
WCLOSE 1
WCLOSE 2
WOPEN 1,1,1,9,77,7
ACTIVE 1
COLOR = 0
DISPLAY"

INSTRUCTIONS:

For multiple choice: Use arrow keys to highlight answer,

then <Enter>. For short answer: Type answer at cursor,
use UNDERSCORE between words, and WATCH YOUR SPELLING!
<Backspace> to fix typos. Case doesn't matter. Use
acronyms if you can."

WOPEN 2,11,1,9,77,7

ACTIVE 2

COLOR = 1

FIND correct5

SAVEFACTS i:\tempdata1

CHAIN i:\resone2.kbs;

!=====RULES BLOCK=====

RULE 1

IF answer1 = c

THEN correct1 = yes

ELSE correct1 = no;

RULE 2

IF answer2 = b

THEN correct2 = yes

ELSE correct2 = no;

RULE 3

IF answer3 = d

THEN correct3 = yes

ELSE correct3 = no;

RULE 4

IF answer4 = c

THEN correct4 = yes

ELSE correct4 = no;

RULE 5

IF answer5 = a

THEN correct5 = yes

ELSE correct5 = no;

!=====QUESTIONS BLOCK=====

ASK answer1: "How do you change you name in LISTSERV?"

- a. Change_name_with_correct_spelling,
- b. Delete_original_and_reregister,
- c. Subscribe_with_correct_spelling,
- d. Tell_list_keeper_to_change_it";

CHOICES answer1: a, b, c, d;

ASK answer2: "What do you do if you do not want your message to go to the default?"

- a. Change_the_automatic_default,
- b. Change_the_address_line,
- c. Change_the_query_line,
- d. Use_LISTSERV_function_key";

CHOICES answer2: a, b, c, d;

ASK answer3: "How can you get the current list-of-lists?"

- a. Query_BITNET @_LISTSERV_&_use_GET_LIST,
- b. Query_LISTSERV @_BITNET_&_use_LIST_GLOBAL,
- c. Query_LISTSERV @_BITNET.bitnic_&_use_GET_LIST,
- d. Query_LISTSERV @_BITNIC.bitnet_&_use_LIST_GLOBAL";

CHOICES answer3: a, b, c, d;

ASK answer4: "What login can you use to download lists?"

- a. Guest,
- b. Your_e-mail_password,
- c. Anonymous,
- d. Readme";

CHOICES answer4: a, b, c, d;

ASK answer5: "What is it called when you look around at different groups?"

- a. Surfing,
- b. Skating,
- c. Jumping,
- d. Moderatin";

CHOICES answer5: a, b, c, d;

NEW FILE: RESONE2.KBS

!EXECUTE;

BKCOLOR = 1;

RUNTIME;

ACTIONS

WOPEN 1,3,10,13,60,5

COLOR = 1

ACTIVE 1

! PRINTON

LOADFACTS i:\tempdata1

SAVEFACTS c:\%UID%.dbt

FIND Q1

```
FIND Q2
FIND Q3
FIND Q4
FIND Q5
! PRINTOFF
RULE 1
IF correct1 = yes
THEN Q1 = yes
    DISPLAY "Resone2.kbs: You have answered question 1 correctly."
ELSE Q1 = no
    DISPLAY "Resone2.kbs: You have answered question 1 incorrectly.";
RULE 2
IF correct2 = yes
THEN Q2 = yes
    DISPLAY "You have answered question 2 correctly."
ELSE Q2 = no
    DISPLAY "You have answered question 2 incorrectly.";
RULE 3
IF correct3 = yes
THEN Q3 = yes
    DISPLAY "You have answered question 3 correctly."
ELSE Q3 = no
    DISPLAY "You have answered question 3 incorrectly.";
RULE 4
IF correct4 = yes
THEN Q4 = yes
```

```
DISPLAY "You have answered question 4 correctly."
ELSE Q4 = no
DISPLAY "You have answered question 4 incorrectly.";
RULE 5
IF correct5 = yes
THEN Q5 = yes
DISPLAY "You have answered question 5 correctly.
```

```
    Please wait for instructions to continue.~"
ELSE Q5 = no
DISPLAY "You have answered question 5 incorrectly.
```

```
    Please wait for instructions to continue.~";
```

NEW FILE: V2D.KBS

```
!EXECUTE;
RUNTIME;
BKCOLOR = 0;
ACTIONS
    WOPEN 1,1,1,3,77,5
    ACTIVE 1
    COLOR = 1
    DISPLAY"          HOW TO (THIS IS TESTABLE)"
    WOPEN 2,5,1,19,77,0
    ACTIVE 2
    COLOR = 14
DISPLAY"
```

SUBSCRIBING TO A LIST:"

COLOR = 15

DISPLAY"

SUB <LIST NAME <YOUR NAME> or

SIGNUP <LIST NAME> <Your Name>"

COLOR = 15

DISPLAY"

DESCRIBE: Subscribing to a list is quite simple. In the body

just type either the sub or signup command as

shown above. If you want to try a Listserv, but

don't feel like committing to the list today, try

PRACTICE Listserv by following the example below.

SUBSCRIBING EXAMPLE:

To: . INTERNET'LISTSERV@ULKYVM.BITNET'

Subj: Your choice

SUB GSS-L Susan Stewart

Ctrl/Z (Use your real name, not email address)"

COLOR = 12

DISPLAY" Press any key to continue~"

CLS

COLOR = 14

DISPLAY"

UNSUBSCRIBE:"

COLOR = 15

DISPLAY"

TWO WAYS TO UNSUBSCRIBE ARE: UNSUB <LIST NAME> or

SIGNOFF <LIST NAME>

DESCRIPTION OF UNSUBSCRIBING:

The above two commands can be used to unsubscribe you to a list. Some groups will send you a welcome message, which reminds you how to unsubscribe. Since each group has a different level of involvement by its members, you may have to join several lists before you get the right mix for you. If you decide you've had enough and want to cancel all your groups, use UNSUB * (you don't need your name or the groups)

EXAMPLES OF UNSUBSCRIBING:

UNSUB GSS-L, SIGNOFF GSS-L"

COLOR = 12

DISPLAY" Press any key to continue~"

CLS

COLOR = 14

DISPLAY"

HOW TO SUSPEND YOUR MAIL:"

COLOR = 15

DISPLAY"

Suspending Mail: SET <LIST NAME> NOMAIL

WHEN DO YOU SUSPEND MAIL?:

When you want to temporarily suspend your messages, use the nomail command. This will allow you to go on vacation and not have a huge backlog when you return. You can also send NOFiles if you don't want

any non-email files sent.

EXAMPLE: SET GSS-L NOMAIL

(The computer automatically reads your email address
from the message you are sending)"

COLOR = 12

DISPLAY" Press any key to continue~"

CLS

COLOR = 14

DISPLAY"

WHAT TO DO TO RESUME MAIL:"

COLOR = 15

DISPLAY"

SET <LIST NAME> MAIL"

COLOR = 15

DISPLAY"

WHAT TO DO TO RESUME MAIL:

This is the opposite of the NOMAIL feature. By sending
this message, you cancel your previous NOMAIL message
and resume mail. Again, the computer will automatically
read your email address.

One example might be: SET GSS-L MAIL"

COLOR = 12

DISPLAY" Press any key to continue~"

CLS

COLOR = 14

DISPLAY"

IF YOU WANT TO VIEW THE SUBSCRIBER LIST:"

COLOR = 15

DISPLAY"

REV <LIST NAME>"

COLOR = 15

DISPLAY"

Viewing Description: This command returns a complete list of subscribers and their email addresses. You have several options you can use such as how the list is sorted, etc. Send the command 'info refcard' for information about the current options. You can also use the command 'rev help-net' for information.

For instance: REV GSS-L"

COLOR = 12

DISPLAY" Press any key to continue~"

CLS

COLOR = 14

DISPLAY"

RETRIEVING FILES:"

COLOR = 15

DISPLAY"

INDEX <LIST NAME> then GET <FILE NAME> <FILE TYPE>

WHAT DOES IT MEAN TO RETRIEVE A FILE:

Use this command when you want to retrieve one of the files you identified with INDEX. You may want to log on manually right after you request a file. Files are

usually sent to you within five minutes or so and sometimes they are rather large. Since you probably can't read 400K as it scrolls by, extract the message to a file and download it from your workspace.

Here's an Example: INDEX GSS-L

```
GET FUZZY-LOGIC LOG4312"
```

```
COLOR = 12
```

```
DISPLAY" Press any key to continue~"
```

```
CLS
```

```
COLOR = 14
```

```
DISPLAY"
```

```
LIST OF TOPICS:"
```

```
COLOR = 15
```

```
DISPLAY"
```

```
COMMAND: INFO?"
```

```
COLOR = 15
```

```
DISPLAY"
```

Description of topics: If a topic has additional information available, use the INFO? command to retrieve it. You can then use the INFO command again for specific topics. One handy feature in LISTSERV is that you can send multiple commands in one message. Make sure to use a separate line for each command.

Example--To: Internet'listserv@bitnic.bitnet'

Subj: Fuzzy Logic

info database

info refcard

Ctrl/Z"

COLOR = 12

DISPLAY" Press any key to continue~"

CLS

COLOR = 14

DISPLAY"

WHEN YOU NEED TO SEARCH AN ARCHIVE:"

COLOR = 15

DISPLAY"

SYNTAX OF COMMAND: INFO DATABASE"

COLOR = 15

DISPLAY"

ARCHIVE INFO: For more information about doing detailed searches

of archived lists, send the message INFO DATABASE

to any host which uses LISTSERV software. The

archived files may contain thousands of messages,

so it is best to send keywords. Usually your search

will be done in batch mode, but there are many options

available. INFO DATABASE will give you the most

up-to-date info about performing these searches.

LAST EXAMPLE: To: Internet'listserv@frenk.bitnet'

Subj: Book

INFO DATABASE"

COLOR = 12

DISPLAY" Press any key to continue~"

CLS

COLOR = 12

DISPLAY"

Ready for another test? Press any key to start.~"

CHAIN i:\ttwo2.kbs;

NEW FILE: TTWO2.KBS

!EXECUTE

BKCOLOR = 0;

RUNTIME;

ENDOFF;

ACTIONS

WOPEN 1,1,1,9,77,0

ACTIVE 1

COLOR = 15

DISPLAY"

INSTRUCTIONS:

For multiple choice: Use arrow keys to highlight answer,
then <Enter>. For short answer: Type answer at cursor,
then use UNDERSCORE between words, and WATCH YOUR SPELLING!
<Backspace> to fix typos. Case doesn't matter. Use acronyms
if you can."

WOPEN 2,11,1,10,77,7

ACTIVE 2

COLOR = 1

FIND correct6

FIND correct7

FIND correct8

FIND correct9

FIND correct10

SAVEFACTS i:\tempdata2

CHAIN i:\restwo2.kbs;

!=====RULES BLOCK=====

RULE 1

IF answer6 = Set_LIST_NAME_MAIL

THEN correct6 = yes

ELSE correct6 = no;

RULE 2

IF answer7 = INFO?

THEN correct7 = yes

ELSE correct7 = no;

RULE 3

IF answer8 = INFO_DATABASE

THEN correct8 = yes

ELSE correct8 = no;

RULE 4

IF answer9 = SUB_LIST_NAME_YOUR_NAME

THEN correct9 = yes

ELSE correct9 = no;

RULE 5

IF answer10 = REV_LIST_NAME

THEN correct10 = yes

ELSE correct10 = no;

!=====QUESTIONS BLOCK=====

ASK answer6: "When you have gone on vacation for a while and return,
how do you resume your mail?";

ASK answer7: "How can you get a list of topics from LISTSERV?";

ASK answer8: "If you want to do a detailed search of archived lists, what
message do you send to a LISTSERV host?

- a. SEARCH_LIST,
- b. INFO_DATABASE,
- c. GET_INFO,
- d. INFO_LIST";

CHOICES answer8: a, b, c, d;

ASK answer9: "How do you subscribe to a list?

- a. SUB_LIST_NAME_YOUR_NAME,
- b. SIGNIN_YOUR_NAME,
- c. SUB_YOUR_NAME,
- d. ENTER_YOUR_NAME_LIST_NAME";

CHOICES answer9: a, b, c, d;

ASK answer10: "How can you view the subscriber list?

- a. INDEX_LIST_GET_SUBLIST,
- b. FTP_VIEW_LIST_NAME,
- c. REV_LIST_NAME,

d. VIEW_HELP-NET";
CHOICES answer10: a, b, c, d;

NEW FILE: RESTWO2.KBS

!EXECUTE;

BKCOLOR = 1;

RUNTIME;

ACTIONS

WOPEN 1,3,10,13,60,5

COLOR = 1

ACTIVE 1

! PRINTON

LOADFACTS i:\tempdata2

FIND Q1

FIND Q2

FIND Q3

FIND Q4

FIND Q5

! PRINTOFF

CHAIN i:\V2E.kbs;

RULE 1

IF correct1 = yes

THEN Q1 = yes

DISPLAY "Restwo2.kbs: You have answered question 1 correctly."

ELSE Q1 = no

DISPLAY "Restwo2.kbs: You have answered question 1 incorrectly.";

RULE 2

IF correct2 = yes

THEN Q2 = yes

DISPLAY "You have answered question 2 correctly."

ELSE Q2 = no

DISPLAY "You have answered question 2 incorrectly.";

RULE 3

IF correct3 = yes

THEN Q3 = yes

DISPLAY "You have answered question 3 correctly."

ELSE Q3 = no

DISPLAY "You have answered question 3 incorrectly.";

RULE 4

IF correct4 = yes

THEN Q4 = yes

DISPLAY "You have answered question 4 correctly."

ELSE Q4 = no

DISPLAY "You have answered question 4 incorrectly.";

RULE 5

IF correct5 = yes

THEN Q5 = yes

DISPLAY "You have answered question 5 correctly.

Please wait for instructions to continue.~"

ELSE Q5 = no

DISPLAY "You have answered question 5 incorrectly.

Please wait for instructions to continue.~";

NEW FILE: V2E.KBS

!EXECUTE;

RUNTIME;

BKCOLOR = 0;

ACTIONS

WOPEN 1,1,1,3,77,5

ACTIVE 1

COLOR = 15

DISPLAY"

Thank you for participating in this thesis project.~";

Appendix A: Prototype Version #3

STARTING FILE: V3A

!EXECUTE;

BKCOLOR = 0;

RUNTIME;

ACTIONS

WOPEN 1,1,1,20,77,0

ACTIVE 1

COLOR = 15

DISPLAY"

LISTSERV: A Powerful Mail Manager

This article describes the LISTSERV program,
which is used by BITNET and other systems to add and delete
customers from mailing lists, among other functions.

The article discusses several areas:

- Background and history of LISTSERV
- Problems associated with LISTSERV
- How To section which outlines some common commands

You will be tested on each of the sections.

Make sure you pay particular attention as you are reading.

You do not have the option to reread sections once you have
gone to the next screen."

COLOR = 12

DISPLAY" Press any key to continue~"

WCLOSE 1

WOPEN 1,1,1,3,77,3

ACTIVE 1

COLOR = 1

DISPLAY" BACKGROUND AND HISTORY (THIS IS TESTABLE)"

WOPEN 2,5,1,16,77,0

ACTIVE 2

COLOR = 15

DISPLAY"

When people are first interested in exploring the Internet, they often do not know other people already on the net. One way to meet people is to put your name on mailing lists based on an area of interest. If you are interested in science, for example, many groups are active on Internet. Each group is different, and often the group's name has nothing to do with its conversations. It is, therefore, advisable to subscribe to a group and read the messages for a week or so before writing your own commentary."

COLOR = 12

DISPLAY" Press any key to continue~"

COLOR = 15

CLS

DISPLAY"

To subscribe to a mailing list, two options can be used (Levine & Baroudi, 1993). The first way is for someone to send a message requesting to be put on or taken off a particular list to a list maintainer.

This list maintainer, in turn, updates the list manually.

This method is ineffective because the list maintainers usually hold real jobs somewhere and often do not update the lists in a timely manner.

The second way is for a computer to automatically update the requests. LISTSERV is an abbreviation for List Server, a family of programs developed by IBM which use automatic mail managers."

```
COLOR = 12
```

```
DISPLAY" Press any key to continue~"
```

```
COLOR = 15
```

```
CLS
```

```
DISPLAY"
```

According to Crossing the Internet Threshold, LISTSERV is a 'BITNET service providing distributed messages that form conferences and allow the archiving of files and messages which can be searched and retrieved.' In other words, it is a software program which runs on a BITNET computer and handles all the administrative functions of a list, such as subscribing people or listing people on the net (LaQuey, 1992)."

```
COLOR = 12
```

```
DISPLAY" Press any key to continue~"
```

```
COLOR = 15
```

```
CLS
```

```
DISPLAY"
```

LISTSERV came about when people using BITNET

realized just how much the size and number of groups were growing. At first only those who were directly connected to BITNET and its IBM mainframe computers were able to use LISTSERV (Levine & Baroudi, 1993). Today, all Internet subscribers have access to this service.

LISTSERV is a somewhat user-unfriendly, but it is worth the extra effort to learn because of its outstanding ability to handle large mailing lists quickly. For instance, LISTSERV can send mail to 1000 addresses in about 5 minutes. The Internet sendmail program would take over 1 hour to perform the same task (Levine & Baroudi, 1993)."

```
COLOR = 12
```

```
DISPLAY" Press any key to continue~"
```

```
CLS
```

```
COLOR = 12
```

```
DISPLAY"
```

```
READY FOR YOUR FIRST TEST? Press any key to begin.~"
```

```
CLS
```

```
CHAIN i:\firstt3;
```

```
NEW FILE: FIRSTT3.KBS
```

```
!EXECUTE;
```

```
BKCOLOR = 0;
```

```
RUNTIME;
```

```
ENDOFF;
```

```
ACTIONS
```

```
WOPEN 1,1,1,9,77,0
```

ACTIVE 1
COLOR = 15
DISPLAY"

INSTRUCTIONS:

For multiple choice: Use arrow keys to highlight answer,
then <Enter>. For short answer: Type answer at cursor,
use UNDERSCORE between words, and WATCH YOUR SPELLING!
<Backspace> to fix typos. Case doesn't matter. Use
acronyms if you can."

WOPEN 2,11,1,9,77,7

ACTIVE 2

COLOR = 1

FIND correct1

FIND correct2

FIND correct3

FIND correct4

FIND correct5

SAVEFACTS i:\temp1

CHAIN i:\bl3.kbs;

!=====RULES BLOCK=====

RULE 1

IF answer1 = b

THEN correct1 = yes

ELSE correct1 = no;

RULE 2

IF answer2 = BITNET

THEN correct2 = yes

ELSE correct2 = no;

RULE 3

IF answer3 = c

THEN correct3 = yes

ELSE correct3 = no;

RULE 4

IF answer4 = d

THEN correct4 = yes

ELSE correct4 = no;

RULE 5

IF answer5 = a

THEN correct5 = yes

ELSE correct5 = no;

!=====QUESTIONS BLOCK=====

ASK answer1: "LISTSERV is a BITNET service which provides distributed messages
that

- a. Allow_archiving_of_files_and_system_implementation,
- b. Form_conferences_and_allow_file_archiving,
- c. List_people_on_the_net,
- d. None_of_the_above";

CHOICES answer1: a, b, c, d;

ASK answer2: "Initially only those who were directly connected to < ? >
were able to use LISYSERV?";

ASK answer3: "Who can use LISTSERV today?"

- a. Anyone_with_a_computer,
- b. Only_those_who_own_IBM_compatible_machines,
- c. Any_Internet_subscriber,
- d. Only_AFIT_students";

CHOICES answer3: a, b, c, d;

ASK answer4: "About how long does it take Internet to send mail to 1000 addresses?"

- a. 2_days,
- b. 1000_minutes,
- c. 3_minutes,
- d. 1_hour";

CHOICES answer4: a, b, c, d;

ASK answer5: "About how long does it take LISTSERV to send mail to 1000 addresses?"

- a. 5_minutes,
- b. 3_minutes,
- c. 2_days,
- d. 1_hour";

CHOICES answer5: a, b, c, d;

NEW FILE: BL3.KBS

!EXECUTE;

BKCOLOR = 1;

RUNTIME;

ACTIONS

WOPEN 1,3,10,13,60,5

```
COLOR = 1
ACTIVE 1
LOADFACTS i:\temp3
PRINTON
  FIND Q1
  SHIP Q1, k:\%USID%.res
  FIND Q2
  SHIP Q2, k:\%USID%.res
  FIND Q3
  SHIP Q3, k:\%USID%.res
  FIND Q4
  SHIP Q4, k:\%USID%.res
  FIND Q5
  SHIP Q5, k:\%USID%.res
PRINTOFF
RULE 1
IF correct1 = yes
THEN Q1 = yes
  DISPLAY "B13.kbs: You have answered question 1 correctly."
ELSE Q1 = no
  DISPLAY "B13.kbs: You have answered question 1 incorrectly.";
RULE 2
IF correct2 = yes
THEN Q2 = yes
  DISPLAY "You have answered question 2 correctly."
ELSE Q2 = no
```

DISPLAY "You have answered question 2 incorrectly.";

RULE 3

IF correct3 = yes

THEN Q3 = yes

DISPLAY "You have answered question 3 correctly."

ELSE Q3 = no

DISPLAY "You have answered question 3 incorrectly.";

RULE 4

IF correct4 = yes

THEN Q4 = yes

DISPLAY "You have answered question 4 correctly."

ELSE Q4 = no

DISPLAY "You have answered question 4 incorrectly.";

RULE 5

IF correct5 = yes

THEN Q5 = yes

DISPLAY "You have answered question 5 correctly.

Please wait for instructions to continue.~"

ELSE Q5 = no

DISPLAY "You have answered question 5 incorrectly.

Please wait for instructions to continue.~";

NEW FILE: NORTON UTILITY PACKAGE

!This package captures timing and test results of each machine by the person's login

@echo off

verify on

cls

```
ncc /n /start:1
ncc /n /start:2
rem ...Insert call to VPX program, part #1
r:\vpexpert\vpk k:\vpctest\v3a.kbs
echo. >>c:\%UID%.dat
echo. >>c:\%UID%.dat

echo Part #1 of the test was finished in: >>c:\%UID%.dat
ncc /n /1 /stop:2 >>c:\%UID%.dat
ncc /n /start:3
rem ...Insert call to VPX program, part #2
r:\vpexpert\vpk k:\vpctest\v3c.kbs
echo. >>c:\%UID%.dat
echo. >>c:\%UID%.dat

echo Part #2 of the test was finished in: >>c:\%UID%.dat
ncc /n /1 /stop:3 >>c:\%UID%.dat
ncc /n /start:4
rem ...Insert call to VPX program, part #3
r:\vpexpert\vpk k:\vpctest\v3d.kbs
echo. >>c:\%UID%.dat
echo. >>c:\%UID%.dat

echo Part #3 of the test was finished in: >>c:\%UID%.dat
ncc /n /1 /stop:4 >>c:\%UID%.dat
echo. >>c:\%UID%.dat
```

echo. >>c:\%UID%.dat

echo The entire test was finished in: >>c:\%UID%.dat

ncc /n /1 /stop:1 >>c:\%UID%.dat

echo. >>c:\%UID%.dat

echo. >>c:\%UID%.dat

echo The test results from part #1: >>c:\%UID%.dat

type i:\t1res.kbs >>c:\%UID%.dat

echo. >>c:\%UID%.dat

echo. >>c:\%UID%.dat

echo The test results from part #2: >>c:\%UID%.dat

type i:\t2res.kbs >>c:\%UID%.dat

echo. >>c:\%UID%.dat

echo. >>c:\%UID%.dat

echo The test results from part #3: >>c:\%UID%.dat

type i:\t3res.kbs >>c:\%UID%.dat

echo. >>c:\%UID%.dat

echo. >>c:\%UID%.dat

NEW FILE: V3C.KBS

!EXECUTE;

RUNTIME;

BKCOLOR = 0;

ACTIONS

WOPEN 1,1,1,3,77,5

ACTIVE 1

COLOR = 1

DISPLAY" PROBLEMS (THIS IS TESTABLE)"

WOPEN 2,5,1,16,77,0

ACTIVE 2

COLOR = 15

DISPLAY"

As most A.F.I.T. students know, mailing lists sometimes cause your computer to be loaded with many unwanted messages. Because of its incredible power, LISTSERV can also contribute to this problem. Some people liked to receive junk mail and others did not. If you are the type of person who gets depressed when 3000 messages are not queued and waiting after a 3-day weekend, then subscribe to multiple groups. Otherwise, it is probably advisable to join one group at a time to see if they are in line with your interests.~"

CLS

DISPLAY"

Another problem LISTSERV had was processing requests for changes of address. Many people change jobs. When this happens, often people's e-mail addresses also change. Because most people were too caught up in moving, they did not consider that LISTSERV did not know about their e-mail address change. Once settled, they try to send something to their group and realize their address is wrong.

It should be an easy thing to change addresses, but because of the way LISTSERV operates, it does not recognize the changes automatically. The people wrote to the keeper of the list with an address change and explanation of the situation.~"

CLS

DISPLAY"

Because of this problem, sometimes messages are also rejected by LISTSERV and error messages are sent to other subscribers.

This can become irritating when several people move at once."

COLOR = 15

DISPLAY"

LISTSERV will change names easily, however. Just send another subscribe message with the correct spelling"

COLOR = 15

DISPLAY"

• Something more of an embarrassment to the originator, rather than a problem inherent in the LISTSERV, is that beginners often send their intention to join a group directly to the group, rather than to the LISTSERV. Using LISTSERV will also require that you know the addresses, which will differ from regular Internet addresses (LaQuey, 1992).~"

CLS

DISPLAY"

Care should be exercised when responding to someone on the list. Some lists have automatic defaults so your response only goes to the original author, whereas others send your reply

to the entire group."

COLOR = 15

DISPLAY"

You will have to change the address line if you did not want your message to go to the default (LaQuey, 1992).~"

CLS

COLOR = 15

DISPLAY"

One of the biggest obstacles to overcome when trying to use the LISTSERV application is determining what lists to explore. Volumes have been written about the different lists, but groups come and go so quickly that it is tough to find that perfect list."

COLOR = 15

DISPLAY"

You could have received the current list-of-lists by sending an e-mail message to LISTSERV@BITNIC.bitnet with the command LIST GLOBAL in the body (Tennant, Ober, & Lipow, 1993)."

COLOR = 14

DISPLAY"

WARNING: It is a large list.~"

CLS

COLOR = 15

DISPLAY"

The public Internet mailing lists are also regularly updated and posted to Usenet newsgroups such as news.lists, news.announce,

newusers, etc. (Hahn & Stout, 1994). You could also have download the list by FTPing rtfm.mit.edu with Anonymous. The files are part1, part2, etc. in the directory /pub/usenet/news.answers/mail/mailling-lists.

Another good list-of-lists was managed by Kent State University. It lists scholarly and academic subject lists. Once again, use the Anonymous FTP from ksuvxa.kent.edu. Change directory (CD) to library and download the file acadlist.readme.~"

COLOR = 15

CLS

DISPLAY"

You will usually download lists by signing in as Anonymous."

COLOR = 14

DISPLAY"

Once you sign in, use your e-mail password to identify yourself.

"COLOR = 15

DISPLAY"

Often the groups monitor who is permitted to join the group by asking questions. Sometimes groups were moderated to keep the volume of traffic at a manageable level (Levine & Baroudi, 1993)."

COLOR = 15

DISPLAY"

The best advice is to get on the system and surf or look around at different groups.~"

CLS

COLOR = 14

DISPLAY"

Ready for a test? Press any key to start.~"

CHAIN i:\tone3.kbs;

NEW FILE: TONE3.KBS

!EXECUTE;

BKCOLOR = 0;

RUNTIME;

ENDOFF;

ACTIONS

WOPEN 1,1,1,9,77,0

ACTIVE 1

COLOR = 15

DISPLAY"

INSTRUCTIONS:

For multiple choice: Use arrow keys to highlight answer,
then <Enter>. For short answer: Type answer at cursor,
use UNDERSCORE between words, and WATCH YOUR SPELLING!
<Backspace> to fix typos. Case didn't matter. Use
acronyms if you can."

WOPEN 2,11,1,9,77,7

ACTIVE 2

COLOR = 1

FIND correct1

FIND correct2

FIND correct3

FIND correct4

FIND correct5

SAVEFACTS i:\tempdata1

CHAIN i:\resone3.kbs;

!=====RULES BLOCK=====

RULE 1

IF answer1 = c

THEN correct1 = yes

ELSE correct1 = no;

RULE 2

IF answer2 = b

THEN correct2 = yes

ELSE correct2 = no;

RULE 3

IF answer3 = d

THEN correct3 = yes

ELSE correct3 = no;

RULE 4

IF answer4 = c

THEN correct4 = yes

ELSE correct4 = no;

RULE 5

IF answer5 = a

THEN correct5 = yes

ELSE correct5 = no;

!=====QUESTIONS BLOCK=====

ASK answer1: "How will you change you name in LISTSERV?"

- a. Change_name_with_correct_spelling,
- b. Delete_original_and_reregister,
- c. Subscribe_with_correct_spelling,
- d. Tell_list_keeper_to_change_it";

CHOICES answer1: a, b, c, d;

ASK answer2: "What do you do if you do not want your message to go to the default?"

- a. Change_the_automatic_default,
- b. Change_the_address_line,
- c. Change_the_query_line,
- d. Use_LISTSERV_function_key";

CHOICES answer2: a, b, c, d;

ASK answer3: "How can you get the current list-of-lists?"

- a. Query_BITNET @_LISTSERV &_use_GET_LIST,
- b. Query_LISTSERV @_BITNET &_use_LIST_GLOBAL,
- c. Query_LISTSERV @_BITNET.bitnic &_use_GET_LIST,
- d. Query_LISTSERV @_BITNIC.bitnet &_use_LIST_GLOBAL";

CHOICES answer3: a, b, c, d;

ASK answer4: "What login did you use to download lists?"

- a. Guest,
- b. Your_e-mail_password,
- c. Anonymous,

d. Readme";

CHOICES answer4: a, b, c, d;

ASK answer5: "What is it called when you look around at different groups?

a. Surfing,

b. Skating,

c. Jumping,

d. Moderating";

CHOICES answer5: a, b, c, d;

NEW FILE: RESONE3.KBS

!EXECUTE;

BKCOLOR = 1;

RUNTIME;

ACTIONS

WOPEN 1,3,10,13,60,5

COLOR = 1

ACTIVE 1

! PRINTON

LOADFACTS i:\tempdata1

FIND Q1

FIND Q2

FIND Q3

FIND Q4

FIND Q5

! PRINTOFF

CHAIN i:\V3D.kbs;

RULE 1

IF correct1 = yes

THEN Q1 = yes

DISPLAY "Resone3.kbs: You have answered question 1 correctly."

ELSE Q1 = no

DISPLAY "Resone3.kbs: You have answered question 1 incorrectly.";

RULE 2

IF correct2 = yes

THEN Q2 = yes

DISPLAY "You have answered question 2 correctly."

ELSE Q2 = no

DISPLAY "You have answered question 2 incorrectly.";

RULE 3

IF correct3 = yes

THEN Q3 = yes

DISPLAY "You have answered question 3 correctly."

ELSE Q3 = no

DISPLAY "You have answered question 3 incorrectly.";

RULE 4

IF correct4 = yes

THEN Q4 = yes

DISPLAY "You have answered question 4 correctly."

ELSE Q4 = no

DISPLAY "You have answered question 4 incorrectly.";

RULE 5

IF correct5 = yes

THEN Q5 = yes

DISPLAY "You have answered question 5 correctly.

Please wait for instructions to continue.~"

ELSE Q5 = no

DISPLAY "You have answered question 5 incorrectly.

Please wait for instructions to continue.~";

NEW FILE: V3D.KBS

!EXECUTE;

RUNTIME;

BKCOLOR = 0;

ACTIONS

WOPEN 1,1,1,3,77,5

ACTIVE 1

COLOR = 1

DISPLAY" HOW TO (THIS IS TESTABLE)"

WOPEN 2,5,1,19,77,0

ACTIVE 2

COLOR = 14

DISPLAY"

SUBSCRIBING TO A LIST:"

COLOR = 15

DISPLAY"

To subscribe to a list you can use the following two commands:

SUB <LIST NAME <YOUR NAME> or SIGNUP <LIST NAME> <Your Name>

Subscribing to a list is quite simple. In the body

just type either the sub or signup command as shown above. If you

want to try a Listserv, but don't feel like committing to the list today, try PRACTICE Listserv as shown in the following example.

To: INTERNET'LISTSERV@ULKYVM.BITNET' then Subj: Your choice then
SUB GSS-L Susan Stewart and Ctrl/Z (Use your real name)"

COLOR = 12

DISPLAY" Press any key to continue~"

CLS

COLOR = 14

DISPLAY"

UNSUBSCRIBING TO A LIST:"

COLOR = 15

DISPLAY"

To unsubscribe to a list, use UNSUB <LIST NAME> or SIGNOFF <LIST NAME>

The above two commands can be used to unsubscribe you to a list.

Some groups will send you a welcome message, which reminds you how to unsubscribe. Since each group has a different level of involvement by its members, you may have to join several lists before you get the right mix for you. If you decide you've had enough and want to cancel all your groups, use UNSUB * (you don't need your name or the groups). For example: UNSUB GSS-L"

COLOR = 12

DISPLAY" Press any key to continue~"

CLS

COLOR = 14

DISPLAY"

SUSPENDING MAIL:"

COLOR = 15

DISPLAY"

When you want to temporarily suspend your messages, use the command:
SET <LIST NAME> NOMAIL. This will allow you to go on vacation and not have a huge backlog when you return. You can also send NOFiles if you don't want any non-email files sent. For example: SET GSS-L NOMAIL. (The computer automatically reads your email address from the message you are sending)"

COLOR = 12

DISPLAY" Press any key to continue~"

CLS

COLOR = 14

DISPLAY"

RESUMING MAIL:"

COLOR = 15

DISPLAY"

When you want to resume mail, use the command: SET <LIST NAME> MAIL
This is the opposite of the NOMAIL feature. By sending this message, you cancel your previous NOMAIL message and resume mail. Again, the computer will automatically read your email address. For example: SET GSS-L MAIL"

COLOR = 12

DISPLAY" Press any key to continue~"

CLS

COLOR = 14

DISPLAY"

VIEWING THE SUBSCRIBER LIST:"

COLOR = 15

DISPLAY"

The following command returns a complete list of subscribers and their email addresses: REV <LIST NAME>. You have several options you can use such as how the list is sorted, etc. Send the command 'info refcard' for information about the current options. You can also use the command 'rev help-net' for information. For example: REV GSS-L"

COLOR = 12

DISPLAY" Press any key to continue~"

CLS

COLOR = 14

DISPLAY"

RETRIEVING FILES:"

COLOR = 15

DISPLAY"

Use INDEX <LIST NAME> then GET <FILE NAME> <FILE TYPE> when you want to retrieve one of the files you identified with INDEX. You may want to log on manually right after you request a file. Files are usually sent to you within five minutes or so and sometimes they are rather large. Since you probably can't read 400K as it scrolls by, extract the message to a file and download it from your workspace. For example: INDEX GSS-L then GET FUZZY-LOGIC LOG4312."

COLOR = 12

DISPLAY" Press any key to continue~"

CLS

COLOR = 14

DISPLAY"

GETTING LIST OF TOPICS:"

COLOR = 15

DISPLAY"

If a topic has additional information available, use the INFO? command to retrieve it. You can then use the INFO command again for specific topics. One handy feature in LISTSERV is that you can send multiple commands in one message. Make sure to use a separate line for each command, however. For example: To: Internet'listserv@bitnic.bitnet'; Subj: Fuzzy Logic; info database; info refcard; Ctrl/Z"

COLOR = 12

DISPLAY" Press any key to continue~"

CLS

COLOR = 14

DISPLAY"

SEARCHING AN ARCHIVE:"

COLOR = 15

DISPLAY"

For more information about doing detailed searches of archived lists, send the message INFO DATABASE to any host which uses LISTSERV software. The archived files may contain thousands of messages, so it is best to send keywords. Usually your search will be done in batch mode, but there are many options available. INFO DATABASE will give you the most up-to-date info about performing these searches. For example: To: Internet'listserv@frenk.bitnet'; Subj: Book; INFO DATABASE"

COLOR = 12

DISPLAY" Press any key to continue~"

CLS

COLOR = 12

DISPLAY"

Ready for another test? Press any key to start.~"

CHAIN i:\two3.kbs;

NEW FILE: TTWO3.KBS

!EXECUTE

BKCOLOR = 0;

RUNTIME;

ENDOFF;

ACTIONS

WOPEN 1,1,1,9,77,0

ACTIVE 1

COLOR = 15

DISPLAY"

INSTRUCTIONS:

For multiple choice: Use arrow keys to highlight answer,
then <Enter>. For short answer: Type answer at cursor,
then use UNDERSCORE between words, and WATCH YOUR SPELLING!
<Backspace> to fix typos. Case doesn't matter. Use acronyms
if you can."

WOPEN 2,11,1,10,77,7

ACTIVE 2

COLOR = 1

FIND correct6

FIND correct7

FIND correct8

FIND correct9

FIND correct10

SAVEFACTS i:\tempdata2

CHAIN i:\restwo3.kbs;

!=====RULES BLOCK=====

RULE 1

IF answer6 = Set_LIST_NAME_MAIL

THEN correct6 = yes

ELSE correct6 = no;

RULE 2

IF answer7 = INFO?

THEN correct7 = yes

ELSE correct7 = no;

RULE 3

IF answer8 = INFO_DATABASE

THEN correct8 = yes

ELSE correct8 = no;

RULE 4

IF answer9 = SUB_LIST_NAME_YOUR_NAME

THEN correct9 = yes

ELSE correct9 = no;

RULE 5

IF answer10 = REV_LIST_NAME

THEN correct10 = yes

ELSE correct10 = no;

!=====QUESTIONS BLOCK=====

ASK answer6: "When you have gone on vacation for a while and return,
how do you resume your mail?";

ASK answer7: "How can you get a list of topics from LISTSERV?";

ASK answer8: "If you want to do a detailed search of archived lists, what
message do you send to a LISTSERV host?

- a. SEARCH_LIST,
- b. INFO_DATABASE,
- c. GET_INFO,
- d. INFO_LIST";

CHOICES answer8: a, b, c, d;

ASK answer9: "How do you subscribe to a list?

- a. SUB_LIST_NAME_YOUR_NAME,
- b. SIGNIN_YOUR_NAME,
- c. SUB_YOUR_NAME,
- d. ENTER_YOUR_NAME_LIST_NAME";

CHOICES answer9: a, b, c, d;

ASK answer10: "How can you view the subscriber list?

- a. INDEX_LIST_GET_SUBLIST,
- b. FTP_VIEW_LIST_NAME,

c. REV_LIST_NAME,
d. VIEW_HELP-NET";

CHOICES answer10: a, b, c, d;

NEW FILE: RESTWO3.KBS

!EXECUTE;

BKCOLOR = 1;

RUNTIME;

ACTIONS

WOPEN 1,3,10,13,60,5

COLOR = 1

ACTIVE 1

! PRINTON

LOADFACTS i:\tempdata2

FIND Q1

FIND Q2

FIND Q3

FIND Q4

FIND Q5

! PRINTOFF

CHAIN i:\V3E.kbs;

RULE 1

IF correct1 = yes

THEN Q1 = yes

DISPLAY "Restwo3.kbs: You have answered question 1 correctly."

ELSE Q1 = no

DISPLAY "Restwo3.kbs: You have answered question 1 incorrectly.";

RULE 2

IF correct2 = yes

THEN Q2 = yes

DISPLAY "You have answered question 2 correctly."

ELSE Q2 = no

DISPLAY "You have answered question 2 incorrectly.";

RULE 3

IF correct3 = yes

THEN Q3 = yes

DISPLAY "You have answered question 3 correctly."

ELSE Q3 = no

DISPLAY "You have answered question 3 incorrectly.";

RULE 4

IF correct4 = yes

THEN Q4 = yes

DISPLAY "You have answered question 4 correctly."

ELSE Q4 = no

DISPLAY "You have answered question 4 incorrectly.";

RULE 5

IF correct5 = yes

THEN Q5 = yes

DISPLAY "You have answered question 5 correctly.

Please wait for instructions to continue.~"

ELSE Q5 = no

DISPLAY "You have answered question 5 incorrectly.

Please wait for instructions to continue.~";

NEW FILE: V3E.KBS

!EXECUTE;

RUNTIME;

BKCOLOR = 0;

ACTIONS

WOPEN 1,1,1,3,77,5

ACTIVE 1

COLOR = 15

DISPLAY"

Thank you for participating in this thesis project.~";

Appendix A: Prototype Version #4

STARTING FILE: V4A

!EXECUTE;

BKCOLOR = 0;

RUNTIME;

ACTIONS

WOPEN 1,1,1,20,77,0

ACTIVE 1

COLOR = 15

DISPLAY"

LISTSERV: A Powerful Mail Manager

This article describes the LISTSERV program,
which is used by BITNET and other systems to add and delete
customers from mailing lists, among other functions.

The article discusses several areas:

- Background and history of LISTSERV
- Problems associated with LISTSERV
- How To section which outlines some common commands

You will be tested on each of the sections.

Make sure you pay particular attention as you are reading.

You do not have the option to reread sections once you have
gone to the next screen."

COLOR = 12

DISPLAY" Press any key to continue~"

WCLOSE 1

WOPEN 1,1,1,3,77,3

ACTIVE 1

COLOR = 1

DISPLAY" BACKGROUND AND HISTORY (THIS IS TESTABLE)"

WOPEN 2,5,1,16,77,0

ACTIVE 2

COLOR = 15

DISPLAY"

When people are first interested in exploring the Internet, they often do not know other people already on the net. One way to meet people is to put your name on mailing lists based on an area of interest. If you are interested in science, for example, many groups are active on Internet. Each group is different, and often the group's name has nothing to do with its conversations. It is, therefore, advisable to subscribe to a group and read the messages for a week or so before writing your own commentary."

COLOR = 12

DISPLAY" Press any key to continue~"

COLOR = 15

CLS

DISPLAY"

To subscribe to a mailing list, two options can be used (Levine & Baroudi, 1993). The first way is for someone to send a message requesting to be put on or taken off a particular list to a list maintainer.

This list maintainer, in turn, updates the list manually.

This method is ineffective because the list maintainers usually hold real jobs somewhere and often do not update the lists in a timely manner.

The second way is for a computer to automatically update the requests. LISTSERV is an abbreviation for List Server, a family of programs developed by IBM which use automatic mail managers."

```
COLOR = 12
```

```
DISPLAY" Press any key to continue~"
```

```
COLOR = 15
```

```
CLS
```

```
DISPLAY"
```

According to Crossing the Internet Threshold, LISTSERV is a 'BITNET service providing distributed messages that form conferences and allow the archiving of files and messages which can be searched and retrieved.' In other words, it is a software program which runs on a BITNET computer and handles all the administrative functions of a list, such as subscribing people or listing people on the net (LaQuey, 1992)."

```
COLOR = 12
```

```
DISPLAY" Press any key to continue~"
```

```
COLOR = 15
```

```
CLS
```

```
DISPLAY"
```

LISTSERV came about when people using BITNET

realized just how much the size and number of groups were growing. At first only those who were directly connected to BITNET and its IBM mainframe computers were able to use LISTSERV (Levine & Baroudi, 1993). Today, all Internet subscribers have access to this service.

LISTSERV is a somewhat user-unfriendly, but it is worth the extra effort to learn because of its outstanding ability to handle large mailing lists quickly. For instance, LISTSERV can send mail to 1000 addresses in about 5 minutes. The Internet sendmail program would take over 1 hour to perform the same task (Levine & Baroudi, 1993)."

```
COLOR = 12
```

```
DISPLAY" Press any key to continue~"
```

```
CLS
```

```
COLOR = 12
```

```
DISPLAY"
```

```
READY FOR YOUR FIRST TEST? Press any key to begin.~"
```

```
CLS
```

```
CHAIN i:\firstt4;
```

NEW FILE: FIRSTT4.KBS

```
!EXECUTE;
```

```
BKCOLOR = 0;
```

```
RUNTIME;
```

```
ENDOFF;
```

```
ACTIONS
```

```
WOPEN 1,1,1,9,77,0
```

ACTIVE 1
COLOR = 15
DISPLAY"

INSTRUCTIONS:

For multiple choice: Use arrow keys to highlight answer,
then <Enter>. For short answer: Type answer at cursor,
use UNDERSCORE between words, and WATCH YOUR SPELLING!
<Backspace> to fix typos. Case doesn't matter. Use
acronyms if you can."

WOPEN 2,11,1,9,77,7

ACTIVE 2

COLOR = 1

FIND correct1

FIND correct2

FIND correct3

FIND correct4

FIND correct5

SAVEFACTS i:\temp1

CHAIN i:\bl4.kbs;

!=====RULES BLOCK=====

RULE 1

IF answer1 = b

THEN correct1 = yes

ELSE correct1 = no;

RULE 2

IF answer2 = BITNET

THEN correct2 = yes

ELSE correct2 = no;

RULE 3

IF answer3 = c

THEN correct3 = yes

ELSE correct3 = no;

RULE 4

IF answer4 = d

THEN correct4 = yes

ELSE correct4 = no;

RULE 5

IF answer5 = a

THEN correct5 = yes

ELSE correct5 = no;

!=====QUESTIONS BLOCK=====

ASK answer1: "LISTSERV is a BITNET service which provides distributed messages
that

- a. Allow_archiving_of_files_and_system_implementation,
- b. Form_conferences_and_allow_file_archiving,
- c. List_people_on_the_net,
- d. None_of_the_above";

CHOICES answer1: a, b, c, d;

ASK answer2: "Initially only those who were directly connected to < ? >
were able to use LISTSERV?";

ASK answer3: "Who can use LISTSERV today?"

- a. Anyone_with_a_computer,
- b. Only_those_who_own_IBM_compatible_machines,
- c. Any_Internet_subscriber,
- d. Only_AFIT_students";

CHOICES answer3: a, b, c, d;

ASK answer4: "About how long does it take Internet to send mail to 1000 addresses?"

- a. 2_days,
- b. 1000_minutes,
- c. 3_minutes,
- d. 1_hour";

CHOICES answer4: a, b, c, d;

ASK answer5: "About how long does it take LISTSERV to send mail to 1000 addresses?"

- a. 5_minutes,
- b. 3_minutes,
- c. 2_days,
- d. 1_hour";

CHOICES answer5: a, b, c, d;

NEW FILE: BL4.KBS

!EXECUTE;

BKCOLOR = 1;

RUNTIME;

ACTIONS

WOPEN 1,3,10,13,60,5

```
COLOR = 1
ACTIVE 1
LOADFACTS i:\temp4
PRINTON
  FIND Q1
  SHIP Q1, k:\%USID%.res
  FIND Q2
  SHIP Q2, k:\%USID%.res
  FIND Q3
  SHIP Q3, k:\%USID%.res
  FIND Q4
  SHIP Q4, k:\%USID%.res
  FIND Q5
  SHIP Q5, k:\%USID%.res
PRINTOFF
RULE 1
IF correct1 = yes
THEN Q1 = yes
  DISPLAY "B14.kbs: You have answered question 1 correctly."
ELSE Q1 = no
  DISPLAY "B14.kbs: You have answered question 1 incorrectly.";
RULE 2
IF correct2 = yes
THEN Q2 = yes
  DISPLAY "You have answered question 2 correctly."
ELSE Q2 = no
```

DISPLAY "You have answered question 2 incorrectly.";

RULE 3

IF correct3 = yes

THEN Q3 = yes

DISPLAY "You have answered question 3 correctly."

ELSE Q3 = no

DISPLAY "You have answered question 3 incorrectly.";

RULE 4

IF correct4 = yes

THEN Q4 = yes

DISPLAY "You have answered question 4 correctly."

ELSE Q4 = no

DISPLAY "You have answered question 4 incorrectly.";

RULE 5

IF correct5 = yes

THEN Q5 = yes

DISPLAY "You have answered question 5 correctly.

Please wait for instructions to continue.~"

ELSE Q5 = no

DISPLAY "You have answered question 5 incorrectly.

Please wait for instructions to continue.~";

NEW FILE: NORTON UTILITY PACKAGE

!This package captures timing and test results of each machine by the person's login

@echo off

verify on

cls

```
ncc /n /start:1
ncc /n /start:2
rem ...Insert call to VPX program, part #1
r:\vpexpert\vpk k:\vpctest\v4a.kbs
echo. >>c:\%UID%.dat
echo. >>c:\%UID%.dat

echo Part #1 of the test was finished in: >>c:\%UID%.dat
ncc /n /1 /stop:2 >>c:\%UID%.dat
ncc /n /start:3
rem ...Insert call to VPX program, part #2
r:\vpexpert\vpk k:\vpctest\v4c.kbs
echo. >>c:\%UID%.dat
echo. >>c:\%UID%.dat

echo Part #2 of the test was finished in: >>c:\%UID%.dat
ncc /n /1 /stop:3 >>c:\%UID%.dat
ncc /n /start:4
rem ...Insert call to VPX program, part #3
r:\vpexpert\vpk k:\vpctest\v4d.kbs
echo. >>c:\%UID%.dat
echo. >>c:\%UID%.dat

echo Part #3 of the test was finished in: >>c:\%UID%.dat
ncc /n /1 /stop:4 >>c:\%UID%.dat
echo. >>c:\%UID%.dat
```

echo. >>c:\%UID%.dat

echo The entire test was finished in: >>c:\%UID%.dat

ncc /n /1 /stop:1 >>c:\%UID%.dat

echo. >>c:\%UID%.dat

echo. >>c:\%UID%.dat

echo The test results from part #1: >>c:\%UID%.dat

type i:\t1res.kbs >>c:\%UID%.dat

echo. >>c:\%UID%.dat

echo. >>c:\%UID%.dat

echo The test results from part #2: >>c:\%UID%.dat

type i:\t2res.kbs >>c:\%UID%.dat

echo. >>c:\%UID%.dat

echo. >>c:\%UID%.dat

echo The test results from part #3: >>c:\%UID%.dat

type i:\t3res.kbs >>c:\%UID%.dat

echo. >>c:\%UID%.dat

echo. >>c:\%UID%.dat

NEW FILE: V4C.KBS

!EXECUTE;

RUNTIME;

BKCOLOR = 0;

ACTIONS

WOPEN 1,1,1,3,77,5

ACTIVE 1

COLOR = 1

DISPLAY" PROBLEMS (THIS IS TESTABLE)"

WOPEN 2,5,1,16,77,0

ACTIVE 2

COLOR = 15

DISPLAY"

As most A.F.I.T. students know, mailing lists sometimes cause your computer to be loaded with many unwanted messages. Because of its incredible power, LISTSERV can also contribute to this problem. Some people liked to receive junk mail and others did not. If you are the type of person who gets depressed when 3000 messages are not queued and waiting after a 3-day weekend, then subscribe to multiple groups. Otherwise, it is probably advisable to join one group at a time to see if they are in line with your interests.~"

CLS

DISPLAY"

Another problem LISTSERV had was processing requests for changes of address. Many people change jobs. When this happens, often people's e-mail addresses also change. Because most people were too caught up in moving, they did not consider that LISTSERV did not know about their e-mail address change. Once settled, they try to send something to their group and realize their address is wrong.

It should be an easy thing to change addresses, but because of the way LISTSERV operates, it does not recognize the changes automatically. The people wrote to the keeper of the list with an address change and explanation of the situation.~"

CLS

DISPLAY"

Because of this problem, sometimes messages are also rejected by LISTSERV and error messages are sent to other subscribers. This can become irritating when several people move at once."

COLOR = 14

DISPLAY"

LISTSERV will change names easily, however. Just send another subscribe message with the correct spelling"

COLOR = 15

DISPLAY"

Something more of an embarrassment to the originator, rather than a problem inherent in the LISTSERV, is that beginners often send their intention to join a group directly to the group, rather than to the LISTSERV. Using LISTSERV also will require that you know the addresses, which will differ from regular Internet addresses (LaQuey, 1992).~"

CLS

DISPLAY"

Care should be exercised when responding to someone on the list. Some lists have automatic defaults so your response only goes to the original author, whereas others send your reply

to the entire group."

COLOR = 14

DISPLAY"

You will have to change the address line if you did not want your message to go to the default (LaQuey, 1992).~"

CLS

COLOR = 15

DISPLAY"

One of the biggest obstacles to overcome when trying to use the LISTSERV application is determining what lists to explore. Volumes have been written about the different lists, but groups come and go so quickly that it is tough to find that perfect list."

COLOR = 14

DISPLAY"

You could have received the current list-of-lists by sending an e-mail message to LISTSERV@BITNIC.bitnet with the command LIST GLOBAL in the body (Tennant, Ober, & Lipow, 1993)."

COLOR = 15

DISPLAY"

WARNING: It is a large list.~"

CLS

DISPLAY"

The public Internet mailing lists are also regularly updated and posted to Usenet newsgroups such as news.lists, news.announce, newusers, etc. (Hahn & Stout, 1994). You could also have download the

list by FTPing rtfm.mit.edu with Anonymous. The files are part1, part2, etc. in the directory /pub/usenet/news.answers/mail/mailling-lists.

Another good list-of-lists was managed by Kent State University. It lists scholarly and academic subject lists. Once again, use the Anonymous FTP from ksuvxa.kent.edu. Change directory (CD) to library and download the file acadlist.readme.~"

COLOR = 14

CLS

DISPLAY"

You will usually download lists by signing in as Anonymous."

COLOR = 15

DISPLAY"

Once you sign in, use your e-mail password to identify yourself.

Often the groups monitor who is permitted to join the group by asking questions. Sometimes groups were moderated to keep the volume of traffic at a manageable level (Levine & Baroudi, 1993)."

COLOR = 14

DISPLAY"

The best advice is to get on the system and surf or look around at different groups.~"

CLS

COLOR = 12

DISPLAY"

Ready for a test? Press any key to start.~"

CHAIN i:\tone4.kbs;

NEW FILE: TONE4.KBS

!EXECUTE;

BKCOLOR = 0;

RUNTIME;

ENDOFF;

ACTIONS

WOPEN 1,1,1,9,77,0

ACTIVE 1

COLOR = 15

DISPLAY"

INSTRUCTIONS:

For multiple choice: Use arrow keys to highlight answer,
then <Enter>. For short answer: Type answer at cursor,
use UNDERSCORE between words, and WATCH YOUR SPELLING!
<Backspace> to fix typos. Case didn't matter. Use
acronyms if you can."

WOPEN 2,11,1,9,77,7

ACTIVE 2

COLOR = 1

FIND correct1

FIND correct2

FIND correct3

FIND correct4

FIND correct5

SAVEFACTS i:\tempdata1

CHAIN i:\resone4.kbs;

!=====RULES BLOCK=====

RULE 1

IF answer1 = c

THEN correct1 = yes

ELSE correct1 = no;

RULE 2

IF answer2 = b

THEN correct2 = yes

ELSE correct2 = no;

RULE 3

IF answer3 = d

THEN correct3 = yes

ELSE correct3 = no;

RULE 4

IF answer4 = c

THEN correct4 = yes

ELSE correct4 = no;

RULE 5

IF answer5 = a

THEN correct5 = yes

ELSE correct5 = no;

!=====QUESTIONS BLOCK=====

ASK answer1: "How will you change your name in LISTSERV?"

a. Change_name_with_correct_spelling.

- b. Delete_original_and_reregister,
- c. Subscribe_with_correct_spelling,
- d. Tell_list_keeper_to_change_it";

CHOICES answer1: a, b, c, d;

ASK answer2: "What do you do if you do not want your message to go to the default?"

- a. Change_the_automatic_default,
- b. Change_the_address_line,
- c. Change_the_query_line,
- d. Use_LISTSERV_function_key";

CHOICES answer2: a, b, c, d;

ASK answer3: "How can you get the current list-of-lists?"

- a. Query_BITNET @_LISTSERV &_use_GET_LIST,
- b. Query_LISTSERV @_BITNET &_use_LIST_GLOBAL,
- c. Query_LISTSERV @_BITNET.bitnic &_use_GET_LIST,
- d. Query_LISTSERV @_BITNIC.bitnet &_use_LIST_GLOBAL";

CHOICES answer3: a, b, c, d;

ASK answer4: "What login did you use to download lists?"

- a. Guest,
- b. Your_e-mail_password,
- c. Anonymous,
- d. Readme";

CHOICES answer4: a, b, c, d;

ASK answer5: "What is it called when you look around at different groups?"

- a. Surfing,
- b. Skating,
- c. Jumping,
- d. Moderating";

CHOICES answer5: a, b, c, d;

NEW FILE: RESONE4.KBS

!EXECUTE;

BKCOLOR = 1;

RUNTIME;

ACTIONS

WOPEN 1,3,10,13,60,5

COLOR = 1

ACTIVE 1

! PRINTON

LOADFACTS i:\tempdata1

FIND Q1

FIND Q2

FIND Q3

FIND Q4

FIND Q5

! PRINTOFF

CHAIN i:\V4D.kbs;

RULE 1

IF correct1 = yes

THEN Q1 = yes

```
    DISPLAY "Resone4.kbs: You have answered question 1 correctly."
ELSE Q1 = no
    DISPLAY "Resone4.kbs: You have answered question 1 incorrectly.";

RULE 2
IF correct2 = yes
THEN Q2 = yes
    DISPLAY "You have answered question 2 correctly."
ELSE Q2 = no
    DISPLAY "You have answered question 2 incorrectly.";

RULE 3
IF correct3 = yes
THEN Q3 = yes
    DISPLAY "You have answered question 3 correctly."
ELSE Q3 = no
    DISPLAY "You have answered question 3 incorrectly.";

RULE 4
IF correct4 = yes
THEN Q4 = yes
    DISPLAY "You have answered question 4 correctly."
ELSE Q4 = no
    DISPLAY "You have answered question 4 incorrectly.";

RULE 5
IF correct5 = yes
THEN Q5 = yes
    DISPLAY "You have answered question 5 correctly.
    Please wait for instructions to continue.~"
```

ELSE Q5 = no

DISPLAY "You have answered question 5 incorrectly.

Please wait for instructions to continue.~";

NEW FILE: V4D.KBS

!EXECUTE;

RUNTIME;

BKCOLOR = 0;

ACTIONS

WOPEN 1,1,1,3,77,5

ACTIVE 1

COLOR = 1

DISPLAY" HOW TO (THIS IS TESTABLE)"

WOPEN 2,5,1,19,77,0

ACTIVE 2

COLOR = 14

DISPLAY"

SUBSCRIBING TO A LIST:"

COLOR = 15

DISPLAY"

How you can subscribe: SUB <LIST NAME <YOUR NAME> or SIGNUP

<LIST NAME> <Your Name>

Describe: Subscribing to a list is quite simple. In the body just type either the sub or signup command as shown above. If you want to try a Listserv, but don't feel like committing to the list today, try PRACTICE Listserv as shown in the following subscribing example: To: INTERNET'LISTSERV@ULKYVM.BITNET' then Subj: Your

choice, then SUB GSS-L Susan Stewart and Ctrl/Z (Use your real name)"

COLOR = 12

DISPLAY" Press any key to continue~"

CLS

COLOR = 14

DISPLAY"

UNSUBSCRIBE:"

COLOR = 15

DISPLAY"

Two ways to unsubscribe are: UNSUB <LIST NAME> or SIGNOFF <LIST NAME>

Description of unsubscribing: These two commands can be used to unsubscribe you to a list. Some groups will send you a welcome message, which reminds you how to unsubscribe. Since each group has a different level of involvement by its members, you may have to join several lists before you get the right mix for you. If you decide you've had enough and want to cancel all your groups, use UNSUB * (you don't need your name or the groups). Examples of unsubscribing: UNSUB GSS-L, SIGNOFF GSS-L."

COLOR = 12

DISPLAY" Press any key to continue~"

CLS

COLOR = 14

DISPLAY"

HOW TO SUSPEND YOUR MAIL:"

COLOR = 15

DISPLAY"

When do you suspend your mail? When you want to temporarily suspend your messages, use the command: SET <LIST NAME> NOMAIL. This will allow you to go on vacation and not have a huge backlog when you return. You can also send NOFiles if you don't want any non-email files sent. Example: SET GSS-L NOMAIL. (The computer automatically reads your email address from the message you are sending)"

COLOR = 12

DISPLAY" Press any key to continue~"

CLS

COLOR = 14

DISPLAY"

WHAT TO DO TO RESUME YOUR MAIL:"

COLOR = 15

DISPLAY"

WHAT TO DO TO RESUME YOUR MAIL: When you want to resume mail, use the command: SET <LIST NAME> MAIL. This is the opposite of the NOMAIL feature. By sending this message, you cancel your previous NOMAIL message and resume mail. Again, the computer will automatically read your email address. SET GSS-L MAIL"

COLOR = 12

DISPLAY" Press any key to continue~"

CLS

COLOR = 14

DISPLAY"

IF YOU WANT TO VIEW THE SUBSCRIBER LIST:"

COLOR = 15

DISPLAY"

Viewing Description: The following command returns a complete list of subscribers and their email addresses: REV <LIST NAME>.

You have several options you can use such as how the list is sorted, etc. Send the command 'info refcard' for information about the current options. You can also use the command 'rev help-net' for information. One example might be: REV GSS-L"

COLOR = 12

DISPLAY" Press any key to continue~"

CLS

COLOR = 14

DISPLAY"

RETRIEVING FILES:"

COLOR = 15

DISPLAY"

WHAT DOES IT MEAN TO RETRIEVE A FILE?: Use INDEX <LIST NAME> then GET <FILE NAME> <FILE TYPE> when you want to retrieve one of the files you identified with INDEX. You may want to log on manually right after you request a file. Files are usually sent to you within five minutes or so and sometimes they are rather large. Since you probably can't read 400K as it scrolls by, extract the message to a file and download it from your workspace. For instance: INDEX GSS-L then GET FUZZY-LOGIC LOG4312."

COLOR = 12

DISPLAY" Press any key to continue~"

CLS

COLOR = 14

DISPLAY"

LIST OF TOPICS:"

COLOR = 15

DISPLAY"

DESCRIPTION OF TOPICS: If a topic has additional information available, use the INFO? command to retrieve it. You can then use the INFO command again for specific topics. One handy feature in LISTSERV is that you can send multiple commands in one message. Make sure to use a separate line for each command, however. Example-- To: Internet'listserv@bitnic.bitnet'; Subj: Fuzzy Logic; info database; info refcard; Ctrl/Z"

COLOR = 12

DISPLAY" Press any key to continue~"

CLS

COLOR = 14

DISPLAY"

WHEN YOU NEED TO SEARCH AN ARCHIVE:"

COLOR = 15

DISPLAY"

ARCHIVE INFORMATION: For more information about doing detailed searches of archived lists, send the message INFO DATABASE to any host which uses LISTSERV software. The archived files may contain thousands of messages, so it is best to send keywords. Usually your search will be done in batch mode, but there are

many options available. INFO DATABASE will give you the most up-to-date info about performing these searches.

Here's an example: To: Internet'listserv@frenk.bitnet'; Subj: Book;

INFO DATABASE"

COLOR = 12

DISPLAY" Press any key to continue~"

CLS

COLOR = 12

DISPLAY"

Ready for another test? Press any key to start.~"

CHAIN i:\two4.kbs;

NEW FILE: TTWO4.KBS

!EXECUTE

BKCOLOR = 0;

RUNTIME;

ENDOFF;

ACTIONS

WOPEN 1,1,1,9,77,0

ACTIVE 1

COLOR = 15

DISPLAY"

INSTRUCTIONS:

For multiple choice: Use arrow keys to highlight answer, then <Enter>. For short answer: Type answer at cursor, then use UNDERSCORE between words, and WATCH YOUR SPELLING! <Backspace> to fix typos. Case doesn't matter. Use acronyms

if you can."

WOPEN 2,11,1,10,77,7

ACTIVE 2

COLOR = 1

FIND correct6

FIND correct7

FIND correct8

FIND correct9

FIND correct10

SAVEFACTS i:\tempdata2

CHAIN i:\restwo4.kbs;

!=====RULES BLOCK=====

RULE 1

IF answer6 = Set_LIST_NAME_MAIL

THEN correct6 = yes

ELSE correct6 = no;

RULE 2

IF answer7 = INFO?

THEN correct7 = yes

ELSE correct7 = no;

RULE 3

IF answer8 = INFO_DATABASE

THEN correct8 = yes

ELSE correct8 = no;

RULE 4

IF answer9 = SUB_LIST_NAME_YOUR_NAME

THEN correct9 = yes

ELSE correct9 = no;

RULE 5

IF answer10 = REV_LIST_NAME

THEN correct10 = yes

ELSE correct10 = no;

!=====QUESTIONS BLOCK=====

ASK answer6: "When you have gone on vacation for a while and return,
how do you resume your mail?";

ASK answer7: "How can you get a list of topics from LISTSERV?";

ASK answer8: "If you want to do a detailed search of archived lists, what
message do you send to a LISTSERV host?

- a. SEARCH_LIST,
- b. INFO_DATABASE,
- c. GET_INFO,
- d. INFO_LIST";

CHOICES answer8: a, b, c, d;

ASK answer9: "How do you subscribe to a list?

- a. SUB_LIST_NAME_YOUR_NAME,
- b. SIGNIN_YOUR_NAME,
- c. SUB_YOUR_NAME,
- d. ENTER_YOUR_NAME_LIST_NAME";

CHOICES answer9: a, b, c, d;

ASK answer10: "How can you view the subscriber list?"

- a. INDEX_LIST_GET_SUBLIST,
- b. FTP_VIEW_LIST_NAME,
- c. REV_LIST_NAME,
- d. VIEW_HELP-NET";

CHOICES answer10: a, b, c, d;

NEW FILE: RESTWO4.KBS

!EXECUTE;

BKCOLOR = 1;

RUNTIME;

ACTIONS

WOPEN 1,3,10,13,60,5

COLOR = 1

ACTIVE 1

! PRINTON

LOADFACTS i:\tempdata2

FIND Q1

FIND Q2

FIND Q3

FIND Q4

FIND Q5

! PRINTOFF

CHAIN i:\V4E.kbs;

RULE 1

IF correct1 = yes

THEN Q1 = yes

DISPLAY "Restwo4.kbs: You have answered question 1 correctly."

ELSE Q1 = no

DISPLAY "Restwo4.kbs: You have answered question 1 incorrectly.";

RULE 2

IF correct2 = yes

THEN Q2 = yes

DISPLAY "You have answered question 2 correctly."

ELSE Q2 = no

DISPLAY "You have answered question 2 incorrectly.";

RULE 3

IF correct3 = yes

THEN Q3 = yes

DISPLAY "You have answered question 3 correctly."

ELSE Q3 = no

DISPLAY "You have answered question 3 incorrectly.";

RULE 4

IF correct4 = yes

THEN Q4 = yes

DISPLAY "You have answered question 4 correctly."

ELSE Q4 = no

DISPLAY "You have answered question 4 incorrectly.";

RULE 5

IF correct5 = yes

THEN Q5 = yes

DISPLAY "You have answered question 5 correctly."

Please wait for instructions to continue.~"

ELSE Q5 = no

DISPLAY "You have answered question 5 incorrectly.

Please wait for instructions to continue.~";

NEW FILE: V4E.KBS

!EXECUTE;

RUNTIME;

BKCOLOR = 0;

ACTIONS

WOPEN 1,1,1,3,77,5

ACTIVE 1

COLOR = 15

DISPLAY"

Thank you for participating in this thesis project.~";

Appendix C: SAS Programs

PROGRAM 1: REPEATED MEASURES TESTS

FOR TEST RESULTS:

```
Options pagesize=78;
data new;
infile res1 missover;
input group 1 (Y1-Y5) (1.);
Y1-Y1+1;
Y2=Y2+1;
Y3=Y3+1;
Y4=Y4+1;
Y5=Y5+1;
y=mean(Y1+Y2+Y3+Y4+Y5);
proc glm;
  class group;
  model Y1-Y5=group / noint;
repeated time;
title 'REPEATED MEASURE TEST FOR RES1';
```

FOR ACCURACY:

```
Options pagesize=78;
data new;
infile res1 missover;
input group 1 (Y1-Y4 (1.);
Y1=6-Y1;
Y2=6-Y2;
Y3=6-Y3;
Y4=6-Y4;
y=mean(Y1+Y2+Y3+Y4);
proc glm;
  class group;
  model y1-y4=group / nouni;
repeated time;
title 'REPEATED MEASURE TEST FOR ACCURACY';
```

PROGRAM 2: DUNNETT TEST

```
Options pagesize=78;
data new;
infile res1 missover;
input group 1 (a1-a5_ (1.);
a1=a1+1;
a2=a2+1;
a3=a3+1;
a4=a4+1;
a5=a5+1;
a=a1+a2+a3+a4+a5;
y=mean(a);
proc glm;
  class group;
  model y=group;
means group / dunnett ('0');
```

PROGRAM 3: TIMING

```
Options pagesize=78;  
data stewart;  
infile timing1 missover;  
input group 1 a 2-4;  
y=mean(a);  
proc glm;  
class group;  
model y=group;  
means group/dunnett ('1');
```

Bibliography

Andre, Anthony D. and Christopher D. Wickens (1992). "Compatibility and consistency in display-control systems: implications for aircraft decision aid design." Human Factors, 34 (6), 639-653.

Bennett, Kevin B, Mona L. Toms, and David D. Woods (1993). "Emergent features and graphical elements: designing more effective configural displays." Human Factors, 35 (1), 71-97.

Boehm, Harry W. Software Engineering Economics Englewood Cliffs NJ: Prentice-Hall, Inc., 1981. p 18.

Boles, David B. and Christopher D. Wickens (1987). "Display formatting in information integration and nonintegration tasks." Human Factors, 29 (4), 395-406.

Calvin, William H. (1994) "The Emergence of Intelligence," Scientific American, Vol 271 Number 4, October, 100-7.

Campbell, Donald T. and Julian C. Stanley Experimental & Quasi-Experimental Designs for Research. Chicago IL: Rand McNally, 1963

Casti, John L. Complex-ification New York NY: Harper Collins Publications Inc., 1994.

Cronin, Mary J. Doing Business on the Internet. New York NY: Van Nostrand Reinhold, 1994.

Davies, Owen, Editor The Omni Book of Computers and Robots. New York NY: Zebra Books, 1981.

Dern, Daniel P. (1994) "Quick tips" Internet World: 40-41 March/April.

Dunn, Ashley (1994) "In California, the numbers add up to anxiety," The New York Times, Sunday October 30.

Emory, C. William and Donald R. Cooper. Business Research Methods (4th edition). Boston MA: Richard C. Irwin, Inc., 1991.

Endoso, Joyce (1994) "DOD IT budget takes smaller hit from Senate." Government Computer News. Vol 13 No 17: 1 August

Fisk, Arthur D. and Cerita D. Jones (1992). "Global versus local consistency: effects of degree of within-category consistency on performance and learning." Human Factors, 34

(6), 693-705.

Floreak, Michael J. (1989) "Designing for the real world: using research to turn a 'target audience' into real people" Technical Communication 36, No 4. 373-81.

Frenzel, Louis E., Jr. Crash Course in Artificial Intelligence and Expert Systems. Indianapolis IN: Howard W. Sams & Co, 1987.

Garcia, Anne Leopold (1990) "Intelligent training systems: smart tutors," Bulletin of the American Society for Information Science. 16: 8-9 August/September.

Guillemette, Ronald A. (1989) "Usability in computer documentation design: Conceptual and methodological considerations" IEEE Transactions on Professional Communication Vol 32 No 4, 217-229

Hubbard, Scott E. (1989) "A practical approach to evaluating test results" IEEE Transactions on Professional Communication Vol 32, No 4, 283-8

Jones, A. Russell and Ester R. Steinberg. "Interface design: an educational perspective," The Journal of Systems and Software, 11: 199-204 (March 1990).

Leibs, Scott (1994) "Why can't PCs be more fun?" Information Week., August 15, 1994 27-34.

Levin, John R. & Carol Baroudi. The Internet for Dummies. San Mateo CA: IDG Books Worldwide, Inc, 1993.

Maul, Gary P. & David Spotts. (1993) "A comparison of computer-based training and classroom instruction." Industrial Engineering. 25-27 February.

McClave, T. James and P. George Benson. Statistics for Business and Economics (5th edition). San Francisco CA: Dellen Publishing Company, 1991.

Mockler, Robert J. & D. G. Dologite. An Introduction to Expert Systems. New York NY: Macmillan Publishing Company, 1991.

Pastoor, Siegmund (1990). "Legibility and subjective preference for color combinations in text." Human Factors, 32 (2) April 1990, 157-71.

Peat, F. David. Artificial Intelligence: How Machines Think. New York NY: Baen Books, 1988.

Queipo, Larry (1991) "Taking the mysticism out of usability test objectives" Technical Communication Second Quarter, 1991, 185-9

Ramez, J. (1988). "How people use computer documentation: Implications for book design." in S. Dohery-Fariner (Ed.), Effective Documentation: What we have learned from Research Cambridge MA: MIT (pp143-158).

Ravden, Susannah J. & Graham I. Johnson. Evaluating Usability of Human-Computer Interfaces: A Practical Method. Chichester, England: Ellis Horwood Limited, 1989.

Redish, Janice C., Daniel B. Felker, & Andrew M. Rose. (1981) "Evaluating the effects of document design principles" Information Design Journal, Vol 2, no 3 & 4, 236-43.

Schraver, Karen A. (1989) "Evaluating text quality: the continuum from text-focused to reader-focused methods" IEEE Transactions on Professional Communication, Vol 32, 1104, 238-55.

Snyder, Joel. (1994) "Diving into the Internet." Internet World. 30-34 March/April.

Solomon, Martin B. (1994) "What's wrong with multimedia in higher education?" THE Journal: Technological Horizons in Education. Vol 21, Number 7: 81-83 February.

Vicente, Kim J. & Jens Rasmussen (1992) "Ecological interface design: theoretical foundations." IEEE Transactions on Systems, Man, and Cybernetics, Vol. 22, No. 4, July/August 1992 589-606.

Waldrop, Mitchell. Complexity. New York NY: Simon and Schuster, 1992.

Wang, Wally & John Mueller. Illustrated VP-Expert. Plano TX: Wordware Publishing, 1989.

Wenger, Michael J. & Jan H. Spyridakis. (1989) "The relevance of reliability and validity to usability testing" IEEE Transactions on Professional Communication, Vol 32 No 4 265-71.

Wickens, Christopher D. & Anthony D. Andre. (1990) "Proximity compatibility and information display: Effects of color, space, and objectiveness on information integration" Human Factors 32(1), 61-77

Vita

Captain Susan Stewart was born on 19 November 1961 in a small mining town in Western Pennsylvania. She holds a Bachelor of Arts Degree in Business Management from the University of Maryland and a Master of Science Degree in Education from the University of Southern California. She was a Russian and Romanian translator with the Air Force as an enlisted member for over nine years. She was commissioned in 1989 and has held the positions of Executive Officer and Squadron Section Commander at Robins Air Force Base, Georgia before coming to the Air Force Institute of Technology. Her awards and decorations include the Meritorious Service Medal, Robins Air Force Base Company Grade Officer of the Year for 1990, and Strategic Air Command Wing Member of the Year for Georgia's Air Force Association chapters. She is married to Gene Stewart. They have three sons: Scott, Keelan, and Niklaus.

Permanent Address: 1004 Tigerville Road

Travelers Rest SC 29690

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE December 1994	3. REPORT TYPE AND DATES COVERED Master's Thesis
----------------------------------	---------------------------------	---

4. TITLE AND SUBTITLE FACTORS IN HUMAN-COMPUTER INTERFACE DESIGN (A PILOT STUDY)	5. FUNDING NUMBERS
---	--------------------

6. AUTHOR(S) Susan Stewart, Captain, USAF
--

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Air Force Institute of Technology, WPAFB OH 45433-6583	8. PERFORMING ORGANIZATION REPORT NUMBER AFIT/GIR/LAR/94D-6
--	--

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) HQ AFRES/IM 155 Second Street Robins AFB GA 31098-1635	10. SPONSORING/MONITORING AGENCY REPORT NUMBER
---	--

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION /AVAILABILITY STATEMENT Approved for public release; distribution unlimited	12b. DISTRIBUTION CODE
--	------------------------

13. ABSTRACT (Maximum 200 words) The DoD has budgeted over \$9.8 billion for 1995 for information technology, yet many government office workers let their existing systems sit idle. This thesis explores why these computers are sitting idle. This researcher's initial hypothesis was that certain features of the human-computer interface can positively or negatively affect efficiency, retention, and satisfaction level of workers. Although some research is being done in this area, interfaces continue to be of poor quality, especially in the DoD, where long procurement cycles, forced purchases, and limited budgets result in out-of-date software. Intuitively most programmers know the human-computer interface impacts on a person's ability to learn, but to what degree? This study is designed to test four basic human-computer interface features: color, white space, verb tense, and parallelism. In addition, it is designed to test the interaction between visual and linguistic features, color and verb tense, as well as between white space and parallelism.

14. SUBJECT TERMS Interface, Computer, Design, Consistency, Feature Selection, Expert System, Color, Parallelism, White Space, Verb Tense	15. NUMBER OF PAGES 204
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UL
---	--	---	----------------------------------