# PROTOCOL ANALYSIS OF MAN-COMPUTER LANGUAGES: DESIGN AND PRELIMINARY FINDINGS

John F. Heafner

University of Southern California

Prepared for:

Defense Advanced Research Projects Agency

July 1975

John F. Heafner

# Protocol Analysis of Man-Computer Languages: Design and Preliminary Findings





INFORMATION SCIENCES INSTITUTE

UNIVERSITY OF SOUTHERN CALIFORNIA

4676 Admiralty Way/Marina del Rey/California 90291
(213) 822-1511

ARPA ORDER NO. 2223

ISI/RR-75-34
July 1975

John F. Heafner

# Protocol Analysis of Man-Computer Languages: Design and Preliminary Findings

INFORMATION SCIENCES INSTITUTE
4676 Admiralty Way/Marina del Rey/California 90291
(213) 822-1511
UNIVERSITY OF SOUTHERN CALIFORNIA

THIS RESEARCH IS SUPPORTED BY THE ADVANCED RESEARCH PROJECTS AGENCY UNDER CONTRACT NO. DAHC15 72 C 0308, ARPA ORDER NO. 2223 PROGRAM CODE NO. 3D30 AND 3P10

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

| REPORT DOCUMENTATION PAGE | | READ INSTRUCTIONS BEFORE COMPLETING FORM |
|---|---|---|
| 1. REPORT NUMBER<br>ISI/RR-75-34 | 2. GOVT ACCESSION NO. | 3. RECIPIENT'S CATALOG NUMBER |
| 4. TITLE (and Subtitle)<br>Protocol Analysis of Man-Computer Languages: Design and Preliminary Findings | | 5. TYPE OF REPORT & PERIOD COVERED<br>Research<br>6. PERFORMING ORG. REPORT NUMBER |
| 7. AUTHOR(s)<br>John F. Heafner | | 8. CONTRACT OR GRANT NUMBER(s)<br>DAHC 15 72 C 0308 |
| 9. PERFORMING ORGANIZATION NAME AND ADDRESS<br>USC/Information Sciences Institute<br>4676 Admiralty Way<br>Marina del Rey, CA 90291 | | 10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS<br>ARPA Order #2223<br>Program Code 3D30 & 3P10 |
| 11. CONTROLLING OFFICE NAME AND ADDRESS<br>Defense Advanced Research Projects Agency<br>1400 Wilson Blvd.<br>Arlington, VA 22209 | | 12. REPORT DATE<br>July 1975<br>13. NUMBER OF PAGES<br>277 |
| 14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)<br>----- | | 15. SECURITY CLASS. (of this report)<br>Unclassified<br>15a. DECLASSIFICATION/DOWNGRADING SCHEDULE |

16. DISTRIBUTION STATEMENT (of this Report)

This document approved for public release and sale; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

-----

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

application-oriented language design, man-computer language design, man-machine communication, message processing, military message processing, protocol analysis, statistical analysis of computer languages, user performance.

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

(OVER)

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-014-6601



UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

## 20. ABSTRACT

This report describes an on-going study in man-machine communications. The study's main premise is that in developing man-computer languages one should consider the users' needs and habits as well as features of the computer service. The problem in doing so is that the designer does not have sufficient quantitative information about the users to enable him to specify languages permitting near-optimal performance. The study proposes and tests a method to achieve a closer fit between users and their computer languages by involving potential users in the design process.

Token languages of several syntactic forms are defined. Then, research hypotheses are stated concerning the users' perferences regarding the language structure and vocabulary. Next, an experiment design is described, based on a statistical model of observations of commands entered by users as they perform a standardized task. The method is tested by protocol analysis with subjects who are potential users. In the protocol analysis, subjects vocally stated commands in each of the token languages as they performed the standardized task. These respondents were requested to change the grammar of each language (during the task) to make it natural for them to use. Their task inputs were used to test the hypotheses. The report concludes that the method of modelling users and then testing draft languages is useful in language design, since there was a consensus of users' opinions as to specific language improvements.

# CONTENTS

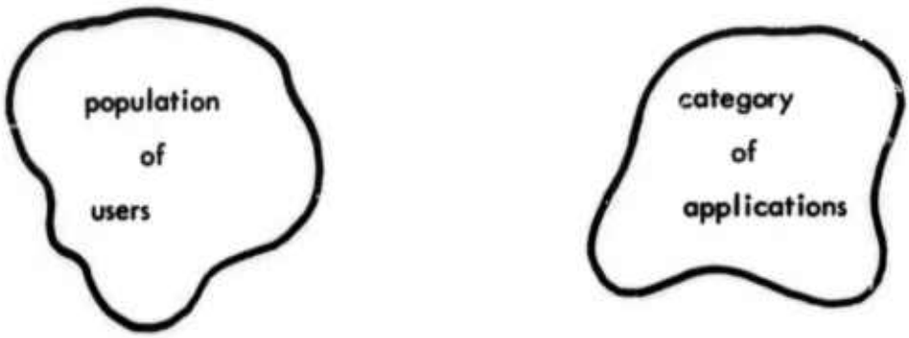
# FOREWORD

## *AN EXPLANATION FOR THE LAYMAN*

In addition to computer scientists, this report addresses managers, military personnel and others who are not computer professionals. The non-computer-trained audience may find the problem definition in the Introduction somewhat terse and jargon-filled. Thus, we take a moment here to explain the protocol analysis problem in the context of the larger problem of choosing optimal languages.

Let us pose the larger problem. Assume we are given a population of users and a category of applications as shown below.

Given:




Our problem is to find an on-line computer language that will maximize the users' performance as they use some service. We may state the problem formally by saying that we wish to find the language as a function of (i.e., by considering) the particular population of users and the particular category of applications. Thus:

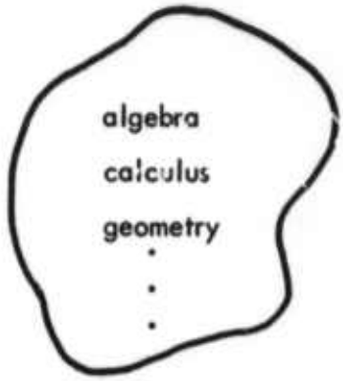
$$L<k> = f\ (U<i>, A<j>) \qquad (1)$$

where L<k> represents the language, U<i> represents some model of the ith population of users, for example, modeled from the point of view of their behaviors and attitudes, and A<j> denotes a specific set (the jth set) of applications, modeled, for example, according to the provided functions. The symbol f denotes the functional relationship among L<k>, U<i>, and A<j>, that is, it tells how we blend together the relative importance of what we know about the users and the service.

Now we also assume that there are many languages from which to choose in deciding L<k>. Let us take this out of the realm of computer services for a moment. We can define families of languages as shown below.

Example language paradigms:


Natural modern languages




Mathematical languages

We can now view our problem as having two steps. First, we want to pick the right language from each family and then we want to pick the best one of those for our purposes. We would like to determine f empirically by measuring performance using various languages for different populations of users and categories of applications. This would give us some basis from which to generalize the equation (1).

What are the pitfalls in choosing languages to be tested? Suppose we decide to test whether English is better than algebra for a teacher to explain certain mathematical problems to students. If the teacher and pupils were French-speaking, then the English probably wouldn't score very well, even if a natural language were better suited for the explanations. The problem here is that English is not representative in terms of the characteristics of the population. Now assume that the algebraic language we choose does not include the feature for directly expressing exponentiation. As a consequence the user would have to be content with representing expressions involving exponentiation by repeated multiplication. For example: $x = 2^3$ would be expressed as $x = 2 * 2 * 2$. Imagine how our user would feel about the language if he wanted to express $x = 2^n$. Again, our language is contaminated in that it is not representative of the family of mathematical languages including algebra. Finally, suppose you were a student of art. Certainly, a language expressing the axioms and theorems of projective geometry would be more useful to you than many other mathematical languages. If we had chosen a language expressing some random field of mathematics, then that language would be biased in that it would not be representative of the application.

This notion of representativeness pervades the application, the user population, and the family from which the language is derived. We have seen contaminants in each of these forms. Thus, our first order of business, before making comparative tests among languages from different families, is to ensure that the languages being compared are representative.

[Returning now to computer languages, we point out that there is usually much more homogeneity within a family and also across families than is evidenced by our non-computer examples above.]

We now postulate that the designer of computer languages does not, in general, construct representative languages. Let us look at his reference frame. It is possible that he may know a moderate amount about the properties that define the user

population. On the other hand, and we feel that this is often the case, he may (for various reasons) know very little about those users. Clearly, he must know the kinds of operations performed by the application. However, what he may not know is the way in which the application is to be used, e.g., such as the common logical sequences of operations, the precise vocabulary natural to the user, and so forth.

Therefore, we wish to aid the language designer in building representative languages -- languages which can, in turn, be used to develop equation (1). We believe that the most direct means of ensuring representativeness is to engage the users' help in specifying the language. The way we accomplish this is as follows. The designer picks a token language (from a family) that he feels is "best", based on his limited knowledge of the user and the service. He then works with users to specify features of syntax and vocabulary. (He will be responsible, of course, for keeping the language unambiguous, parsable, etc.) The procedure for this cooperative design is the subject of this report.

# PREFACE

Under ARPA sponsorship, USC/Information Sciences Institute (ISI) is currently developing methods to automate various aspects of military message processing. In particular, the Information Automation project at ISI is providing a prototype message service to consolidate, through automation, parts of the communication task now handled manually and semiautomatically by military Action Officers at CINCPAC Headquarters, Camp Smith, Oahu.

This document reports progress on the investigation of man-computer language forms. Specifically, a protocol analysis of three language forms is discussed which determines for each language its ease of learning (i.e., form and presentation) and its sufficiency for use by computer-naive individuals. Its purpose is to elicit from the user his syntactic preferences within the domain of each of the three different language forms in order to be able to systematically remove biases from the languages for later comparative tests of performance. The protocol analysis is intended to be administered to future military users of the prototype message service. To test the methods used, it has been given to members of the professional staff at ISI.

The reader is assumed to be familiar with the reports and papers referenced herein, that is, those dealing explicitly with the message service. However, to understand the relationship of the protocol analysis to the man-machine dialogue study, and to the whole of the Information Automation project, one should have an appreciation for the materials cited.

Intended readership includes ARPA, those persons at CINCPAC, NAVCOSSACT, NAVELEX and MITRE who are involved in the planned Oahu message service test, and other ARPA contractors with an interest or involvement in message processing.

# SUMMARY

The general field of study of the work reported here is man-machine communications. In particular, *protocol analysis* is used as an aid in developing languages for persons inexperienced in the use of computers.

Because Computer Science is a relatively new discipline, recent research has largely been exploratory. Consequently, a gulf has developed between the demonstration of sound principles and their application to useful problems in society. Notable examples occur in the area of, for example, artificial intelligence, where some good ideas have been applied to "toy" problems. But in many cases, the level of work has not yet reached a point at which it can be usefully applied in a larger context to a meaningful problem. The work reported here is an instance of applying some *researched notions* to the solution of an existing problem. In this case, central to the researched ideas are networks of computers and automatic message processing, and the relevant problem is to upgrade, by automation, parts of the message handling tasks now done manually and semiautomatically by the military. USC/Information Sciences Institute is developing a prototype military message service, and part of the remaining *applied* research necessary to make the service useful involves determining languages that will enhance users' performance.

The introductory chapter states the problem and an approach to solving it. A first step in developing languages for computer-naive users, which actually precedes comparative tests of performance of the languages, is to remove or reduce biases in the language specifications introduced by the designers. The method employed for this purpose is to model, analyze, and evaluate the use of some typical languages as they are applied to a representative task by potential users of the message service. The model is statistical; the task is accomplished by means of protocol analysis.

Preliminary remarks justify the alleviation of experimental bias. One dependent variable (i.e., the ordering of parameters within a command) is chosen to illustrate contamination effects of languages where bias is not systematically removed. The scope of the protocol analysis exercise is limited to the study of 14 dependent variables (such as parameter ordering) and three independent variables (i.e., three input language forms). The languages selected - keyword, positional and English-like - were chosen because they appear frequently as forms of problem-oriented languages.

Finally, the introduction highlights the three-part protocol analysis exercise. Collectively, subjects are first presented an overview of the message service along with the task to be performed and descriptions of, and examples from, the languages. The second part of the exercise is the separate and individual performance of a typical, prespecified message handling task by each subject. It is a vocal scenario between the subject and two analysts who act as consultant and simulator of the service. The third part is a post-task interview with each subject. The interview is loosely structured in order to acquire information not necessarily anticipated. It focuses on the languages but includes the participants' opinions on other subject matter of tangential interest,

namely, message service functions and tutoring of users. Parts two and three of the exercise are recorded for later analysis and evaluation.

Chapter II is really a paper with slides and lecture notes, prepared to cover the first part of the exercise.* In describing the task and the test languages to the subjects, several key points are stressed where the subjects are free to change the languages to suit their individual interaction style. Naturally, these areas of allowable change are among the dependent variables. They include the following:

- Composition of several commands,
- Vocabulary substitutions to all language elements,
- Changes in structure, such as re-ordering arguments of a command,
- Omission of arguments where they can be implied by context or prestated.

Chapters III and IV portray the roles of the simulator and the consultant in the exercise. The simulator's part is minor, since handouts of simulated display responses are preprogrammed. Generally speaking, the consultant plays a passive role, yet he may at his volition rectify some obvious misunderstanding.

Essential features of Chapter V are the dependent variables observed and measured. The categories of the variables are vocabulary changes, inter- and intracommand structuring, advice sought and references used, "errors," and language element omissions.

Chapter VI describes a statistical model ***a propos*** of the exercise. It is one particular model used for analysis of variance which allows us to test the significance of differences among the languages with respect to each of the dependent variables. Where further analysis and description of the acquired data are evidently needed, correlation is employed along with graphical representation of data.

The findings reported here relate to a population of computer programmers. As such, they are not useful in any final sense, but they do serve as a checkpoint to let us pause to examine and evaluate the methods used in the protocol analysis before proceeding with the population of Action Officers. Chapter VII contains the research hypotheses, underlying assumptions, the analytical work, and evaluations based on the analysis. The findings are, on the whole, congruent with the research hypotheses. Several salient results are summarized here, though it should be borne in mind that we would not anticipate a high correlation between these results and those that might be obtained from subjects representing the military population.

---

*Initial remarks in this chapter carefully distinguish between the intended population of military Action Officers and a second population containing the experienced programmers who participated in the preliminary application of the protocol analysis. The results presented in this report address the second population.

Research hypotheses are classified according to the 14 dependent variables. They were supported in ten of the categories, partly supported in another, and rejected in three categories. For instance, from those hypotheses dealing with vocabulary, it was found that changes in punctuation were far more significant than word substitutions. These experienced users were concerned not with the human readability of the input but with minimizing keystrokes and with the locations of the different keys on the keyboard. Among the test languages, the English-like had been constructed "right" by using the natural delimiter of a blank space. To look at a different kind of example, composition of commands was stated as the Null hypothesis, i.e., we didn't expect any language-dependent differences. This was found to be the case when the languages were considered as wholes. However, upon examining short, logically related sequences, language dependencies were uncovered. Surprisingly, the positional language and the English-like language were significant over keyword with respect to the same command sequences, and at present we offer no satisfactory explanation for this finding. In the investigation of some hypotheses, the conclusions are supported by graphical representations of the data. In particular, trend lines for the variable "references to materials" versus order of languages sharply points out the inadequacy of our sample size.

Among the specious hypotheses, the most surprising outcome was a refutation that reordering of parameters would be most prevalent in the positional language. In retrospect it was concluded that one contributing factor was that many commands had one or two parameters, and also that these experienced users could quite easily adapt to an arbitrary ordering.

The major sources of experiment error are discussed in Chapter VIII. Controls used to reduce such errors are those often applied to similarly designed experiments in the behavioral and social sciences. The strategy was to identify certain variables and hold them as near constant as possible. Classes of such variables include situation variables (such as room arrangement, amount of noise and distraction, etc.), treatment variables (such as the amount of indoctrination and practice and the responses by the analysts), and population variables (such as sex or previous experience). In retrospect we feel that sources most significantly contributing to errors were rating (i.e., scoring observations) by the observer and the *standard error* inherent in small samples.

The main aphorism resulting from this interim test of the protocol analysis is that it is indeed judged to be a useful tool for language design. Several improvements in the procedure, contents, and equipment are recommended: a larger sample size, some additional dependent variables, some modifications to the syntax of the test languages, more reliable recording equipment, and a less amorphous interview that yields to analytical study.

Appendices A through F are materials used in the protocol analysis. Appendix A, a composite of the task instructions, language syntax, and simulated output displays, is a step-by-step guide for the simulator, the consultant, and the observer to follow each operation of the message processing task. Appendix B contains the task instructions for the subject. Appendices C, D, and E constitute the language reference manuals used by the subject in the exercise. In actual use they are broken down into subsections specific to task units of the exercise. Appendix F is the display handouts.

Appendix G contains transcripts of the post-task discussions with each subject. The subjects remark on the languages, suggest ways to train users, and recommend service functions. The purpose of the interviews is to gather information for the subjective evaluation of the message system by the designers. Excerpts from the task dialogue that are germane to these issues are also included. A summary of the remarks is also included in the appendix.

# ACKNOWLEDGMENTS

Planning and conducting a small experiment such as this one and preparing the report requires the help of many people. The willingness and assistance of those mentioned below are very much appreciated.

Clearly outstanding among those contributing to this work is Katie Patterson, who contributed much time and her many talents in essential ways reflected throughout the report. Special thanks are due Katie for her performance as typist, programmer, graphic artist, and friend.

Others to whom the author is indebted are:

Guidance committee: Professor Irving Reed, chairman; Professor Robert Anderson, advisor; Professor Stephen Madigan; Professor Thomas Hibbard; and Professor Donald Oestreicher.

Technical reviewers: Dr. Thomas Bell, TRW Systems, Inc.; Dr. Marcia Hopwood, Rand; and Dr. Stephen Kimbleton, USC/ISI.

Review and assistance by project personnel: Dr. Donald Oestreicher and Project Manager Rob Stotz who acted as "sounding board" for various ideas, and Dr. James Carlisle, who provided occasional yet thoughtful comments on the statistical analysis.

Editor: Dr. Nancy Bryan.
Graphics: Nelson Lucas and Katie Patterson.
Typists: Katie Patterson and Rennie Simpson.

Comments on the English-like language: Dr. Russell Abbott, California State University at Northridge.

Advice on service functions and token language construction: Ronald Tugender.

The cast of characters in the "dry run" experiment:
Spokesman for indoctrination lecture: Ronald Tugender
Consultant and Simulator: Larry Miller
Subjects: Richard Bisbey, Tom Boynton, Joel Goldberg, Dr. Norton Greenfeld, Dr. Lee Richardson, Dale Russell, Dr. David Wilczynski, Dr. David Wile, Dr. Martin Yonke

I would like to thank Dr. Greenfeld for comments on a draft of this report, in addition to his role as subject in the experiment.

Thanks are also owed to Professor Stuart Mandell, University of Southern California, for his remarks on the research process, which were helpful in organizing this material. And apologies are owed as well, for not strictly following his recommendations.

# I. INTRODUCTION

## *MOTIVATION FOR THE STUDY*

The general field of study of the work reported here is man-machine communications. In particular, ***protocol analysis*** is used as an aid in developing languages for military message processing [ELLIS 73]. The protocol analysis focuses on ease of learning and sufficiency for performance of tasks by persons inexperienced in the use of computers.

Because Computer Science is a relatively new discipline, recent research has been largely exploratory. Consequently, a gulf has developed between the demonstration of sound principles and their application to useful problems in society. Notable examples occur in the area of, for example, artificial intelligence, where some good ideas have been applied to "toy" problems. But in many cases, the level of work has not yet reached a point at which it can be usefully applied in a larger context to a meaningful problem. The work proposed here is an instance of applying some ***researched notions*** to the solution of an existing problem. In this case, central to the researched ideas are networks of computers and automatic message processing, and the relevant problem is to upgrade, by automation, parts of the message handling task now done manually and semiautomatically by the military. USC/Information Sciences Institute is developing a prototype military message service, and part of the remaining ***applied*** research necessary to make the service useful involves determining languages that will enhance users' performance.

Literature reporting interdisciplinary work, such as the cross-fertilization of computer and behavioral sciences, is scant. Much work has been done in computer systems performance [MILLER 73], and also some in human factors in man-machine communications (e.g., user motivation as determined by behaviors and attitudes) as reported in Human Factors and Ergonomics. Yet, there is a paucity of the application of experimental and quasi-experimental research in language selection methods which tailor man-machine interactions to kinds of users and types of service. The work reported here borrows two research methods, namely, experimental research [CAMPBELL and STANLEY 63] and survey research [BABBIE 73], and applies them to one instance of examining languages with respect to user characteristics and application idiosyncrasies.

One prior related study [BOIES 74] recommends the investigation of the language forms addressed by this study. This earlier study discusses observations of the usage of an interactive computing system in a research environment. Empirical data on user behavior are discussed that concern command usage (among other variables), and Boies concludes the following:

> Based on our studies up to this time, we believe that it is important to develop an understanding of the behavioral criteria that may be useful in designing command languages for interactive systems. . . .Since almost any command language format can be implemented, behavioral criteria can be used as the basis for selecting formats that best suit users' needs and habits. Because virtually any command language has parameters associated with at least some

of the commands, basic behavioral work should be undertaken to explore the advantages and disadvantages of positional, keyword, and mixed formats from the standpoint of user performance.

## *PROBLEM STATEMENT AND OBJECTIVE*

This report describes a protocol analysis used as a pretest to develop tractable languages suitable for testing the language selection methodology proposed earlier [HEAFNER 74].* Formal pretests such as described here are not common to the computer language design task. Thus it is important to stress exactly what the pretest examines and why, and to indicate as well what it deliberately omits. It is an exercise planned to discover how a certain group of users would elect to express statements (within the constraints of each of several language forms) to accomplish a given kind of task. Knowing the users' predilections allows the language designer to specify languages representative of the tested forms such that each language is on an "equal footing," so to speak, with regard to the users and the tasks. ***Technically, then, its purpose is to eliminate the language designer's bias in specifying the grammar of input languages.***

## *WHY A PRETEST IS NEEDED*

In general, in the absence of some form of critical review by the users, it would be a fallacy to assume that the designer could specify languages representative of the users' needs and habits. It is reasonable to assume (and the results of this study confirm) that biases introduced by the designer are not randomized, hence not effectively cancelling, with respect to the metrics on which the language is judged.

To illustrate this need for pretesting, let us use two common language forms -- keyword and positional. An example of a dependent variable to be measured is the user-preferred sequence of parameters in commands. Assume, all other factors being equal, that users perform equally well with a given command in both a certain keyword language and a positional language if the parameters in the positional language are entered in a certain order. By contrast, assume that if the parameters are given in some other sequence, then the keyword language results in higher performance, according to some criterion. Then, if one analyzed users' performance as a function of these languages using the second parameter arrangement, the results would be unjustly biased against positional notation. Thus the pretest is a ***preparatory step*** to comparative testing of differences among languages with respect to user's performance. Similar arguments can be stated for each dependent variable that the pretest considers.

---

*As a pretest it does not test comparative language performance differences. Such tests are to be conducted later with a prototype message service.

## *EXTENT OF THE PROTOCOL ANALYSIS*

Artifacts of communication, such as different forms of output responses to commands, indeed the terminal itself, and in general the "interaction style" are purposely not examined by the protocol analysis. This in no way implies that these variables of man-computer interaction are of little consequence in determining user's performance.[1] Furthermore, we are not asserting that there are no interaction effects between, say, the order of parameters in a particular command of positional notation and a given variety of command recognition. We are simply isolating one main effect, namely ***input syntactic forms***, and studying these forms in the absence of other factors which perhaps contribute to performance. They are so partitioned because, first, we believe they can effectively be studied alone, and second (and more importantly), if languages are not representative, then they can confound (in a statistical sense) tests of performance. The pretest does address the following: given an input syntactic form, how would a particular group of users like to customize it for a unique application, hence how can one design languages which are sufficiently representative of the users and tasks such that follow-on experiments can test meaningful variations in performance? Thus we are studying one independent variable (language form) at three levels (keyword, positional, and English-like). The metrics for gauging the user's activity with respect to these languages consists of fourteen dependent variables (see Chapter V) dealing with vocabulary and syntax.

## *PROTOCOL ANALYSIS OVERVIEW*

With regard to the message service [OESTREICHER 74, TUGENDER 74] we must assure that the specific languages are indeed equally sufficient for the Action Officers[2] to conduct message processing tasks. This is done by defining "strawman" languages[3] and then pretesting them by protocol analysis where the subjects of the pretest are the intended users: Action Officers. The results of the protocol analysis then provide the most important ingredient in designing the actual languages that will be used for experimentation; i.e., the Action Officers themselves, through protocol analysis, will contribute essentially to the language design What is the protocol analysis involved in

---

[1] These variables will be accounted for (either as constants, or tested as variables) in the later tests.

[2] The preliminary results reported in this document pertain to a population of computer programmers.

[3] Input by function keys (such as anticipated for editing control commands) is not tested. The pretest employs commands only at the message service functional level, not at the editing level. We expect that function keys, for example, will be used (in conjunction with output menus) as a natural part of each of the languages. However, we do not envision a ***strictly*** function key language simply because the message service will eventually handle too many operations and also operations which are too complex for natural and easy use solely of function keys, at least by any one who has used the service several times or more.

language pretesting? Action Officers (potential users) are shown the strawman languages and with an analyst (but no computer system) they participate in a vocal scenario as if they were performing one of their daily, typical tasks using the language. They are invited to--in fact their mission is to--comment on the strengths and weaknesses of 1) syntactic idiosyncrasies and anomalies of individual operations such as parameter arrangements, abbreviations, and vocabulary, 2) the service functions themselves, and 3) observations concerning user training. Although we are formally analyzing only the languages, information on service functions and user training is also of interest and will be passed along to those responsible for planning those portions of the message system.

The protocol analysis is conducted as follows. In an indoctrination lecture all Action Officers participating in the experiment are given a general (language-independent) description of the message service functions [TUGENDER 74]. Next they receive a general description of the experiment and its purpose (see Chapter II). They then go over a sample task, which will be used in the experiment, using examples of the different language forms. They are then divided into test groups and each individual of each group participates in several sessions of simulated automatic message handling, using a different language form in each session. At the beginning of each session, the language (See Appendices C, D, and E) and task (Appendix B) are explained in greater detail. After performing the tasks, which consists of the task described to him earlier and one or more which he feels is close to his daily activity, the Action Officer is asked to comment as indicated earlier.*

The arrangement of a session is depicted in Fig. I-1. An analyst plays the part of the automated message service (simulator). The observer's responsibility is to witness and record interesting portions of the task scenarios. In particular, he observes certain

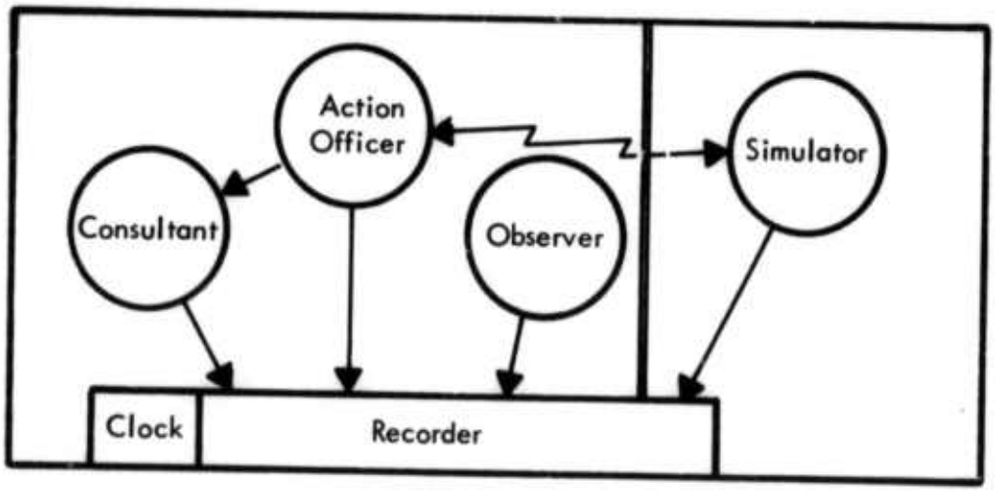



Figure I-1 Language pretest setup

---

*Comments appear in the form of an interview, employing survey research methods unlike the experimental research design for the session scenarios.

properties of the dialogue which have been earlier earmarked as important to the language design. The consultant gives advice to the Action Officer upon request and additionally when he feels the situation calls for it, and he also provides the Action Officer with handouts of simulated display responses as appropriate. The sessions are tape-recorded. The clock shown in Fig. I-1 is slaved to the servo of the recorder to allow the observer to synchronize interesting portions of the tape with pencil-and-paper data. The standard tape annotation for each condition (session) is 1) user identification, 2) task order number, and 3) tape position.

Some number of Action Officers will participate in the experiment using the Latin square design shown in Table I-1. (See also Chapter VI.) Each condition consists of about 45 minutes, composed of the events shown in Table I-2. The standard, prepared task is included to insure a basis of comparison among groups. Extemporaneous tasks suggested by the Action Officers are included to insure that the scenarios, and hence the languages, will reflect their actual tasks.

Table I-1

Latin square design for session ordering

| Group | Order | | | | |
|---|---|---|---|---|---|
| | 1 | 2 | 3 | 4 | 5 |
| 1 | L<1> | L<2> | L<3> | N | N |
| 2 | L<2> | L<3> | L<1> | N | N |
| 3 | L<3> | L<1> | L<2> | N | N |

L<1> = positional functional
L<2> = keyword functional
L<3> = English-like
N = natural language

Each group contains the same number of Action Officers.
Orders 1–4 represent standard task and order 5 is a user-suggested task.

Table I-2

Session composition

| Approximate Time (minutes) | Event |
|---|---|
| 5 | Explanation |
| 35 | Task |
| 5 | Comments |

## *NOTE ON THE REMAINDER OF THE REPORT*

The hypotheses of the study, which would normally appear earlier in such a report, are given in Chapter VII to provide continuity in reading the findings.

The remaining sections describe the protocol analysis in more detail from the viewpoint of each of the participants. The experiment design is given, along with results obtained from a "dry run" administered to members of ISI's professional staff.

The appendices are the nucleus of the materials used with the pretest. They are included so that the exercise may be reproduced. Those on language descriptions (C, D, and E) may at first glance seem amorphous and perfunctory. They are, however, carefully structured, yet purposefully incomplete in two particular ways. The syntax is not always wholly specified, and the set of commands does not represent the entire collection of functions to be supported by the prototype service. An example of incomplete or inconsistent syntax can be illustrated by the naming of a message, where in the English-like language the key phrase appears as (THE) MESSAGE (WHOSE) in one command and again as THE MESSAGE in another. This is a deliberate attempt to get across two points to the subjects of the experiment, namely, that the forms provided are just some suggested ways of stating a command and that the subject may replace them by a new form of his choosing. Only those commands needed for the exercise are included in the descriptions. In administering the exercise, the language descriptions were each broken into four parts: one each for the author, the coordinator, the releaser, and the recipients. This provides a smaller and more manageable language manual for the exercise.

Appendix A is a composite of the task instructions (Appendix B), the language descriptions (Appendices C, D, and E), and the simulated display handouts (Appendix F). Appendix A is to be used by the simulator, consultant, and observer.

# II. THE ACTION OFFICER: INDOCTRINATION LECTURE

## *TARGET AND DRY RUN POPULATIONS*

Throughout this report reference is made to two entirely different populations. Care has been taken to distinguish between them, and the reader should be aware of the distinction at all times. Action Officers comprise the population for which the protocol analysis was planned. This chapter appears in much the same way that it is to be presented to the Action Officers. The second population, which we will refer to here as a subset of computer professionals, will be defined more carefully in a later chapter. The sample from the second population is the group of nine ISI computer professionals who participated in the "dry run" of this protocol analysis.

The remainder of this chapter describes the indoctrination lecture planned for the population of Action Officers. It was presented essentially the same way to the ISI group. The most notable exception is that the ISI group was instructed to play the role of ***experienced*** users.

## *NATURE AND PURPOSE OF EXERCISE*

The purpose of this exercise is to allow potential users to help design the message service languages. Only from cooperation by the users of the service and the builders can we hope to achieve a service that effectively provides the needed functions in a form that is somewhat natural and convenient to use. As a first step in that cooperative design, we would like to conduct an exercise with your help. To help us understand how you process messages, the exercise includes you as a participant in an off-line, simulated message service. The simulated exercise involves performing a message processing task which we have devised, using several different language forms, and also performing a simulated task suggested by you, one which you feel reflects your daily message-handling activities. After the simulated message processing, we ask you to comment on your likes and dislikes of the languages.

Now we would like to describe 1) the setting for the exercise, 2) the standard task to be performed, and 3) the languages to be used in performing the task.

## *MILIEU FOR THE EXERCISE*

We would like to carry out the message processing exercise separately with each of you, in a setting like this (SLIDE 1). You perform a given message handling task where the message service simulator is really one of our analysts. You enter commands by talking (instead of typing). The message service responds in two ways: sometimes the simulator will talk to you and sometimes you will be given printed handouts that look like outputs you would expect to see on a display.

You may also talk freely with the consultant. The consultant's main job is to act as an expert adviser on the message service. He can answer questions about the service functions implied by a command or about the syntax and meaning of commands, parameters, and their values.

Incidentally, the observer shown here does not actively participate in the task dialogue. As the scenario progresses his job is to fill out a checklist that records your preferences--things that are difficult, easy and natural, and so forth. The purpose of this observation is to collect data on how you personally like to carry out the task. Don't be concerned about making so-called "errors". In fact, the occurrence of errors pinpoints places where the languages need to be improved; it is really our error and not yours. Also, don't be concerned about the amount of time you take to enter a command. There are no time limits. If you need to think considerably about a command, then that too points out a deficiency of the language.

You perform the same task in each of three input language forms (SLIDE 2), and also in natural language. That is, the task is completely done in the first language, then in the second, then the third, and finally in English. Lastly, we would like you to suggest and then perform (in English) another task which you feel is imitative of the kinds of message handling tasks you normally do.

## *THE TASK DESCRIPTION*

In performing the standard simulated task you have at your disposal (SLIDE 3) the following 1) instructions for the task to perform, 2) a description of the command language to be used in performing the task, 3) handouts of simulated display responses, and 4) access to a consultant for advice and interpretation.

The standard task (SLIDE 4) you are asked to perform is to handle a single message as it passes through its various phases (creation, coordination, release, etc.). You are to assume the role of the active party--author, coordinator, and so forth. As such you are to take the actions described to you, to process that particular step of the message handling. The task description is broken up into logical units of work (SLIDE 5) for each active party to complete. They should be completed in the order given you. Each unit contains a series of operations for that active party to perform. Again, the operations should be done in the order given. ***However, each operation is stated separately, although some of them may (at your discretion) be combined into a single command.*** Also, there may be several ways to state an operation, and you should choose the expression most natural to you.

You will note that the task operations consists only of message service functions at the level of routing and taking actions on a message. The actual typing involved in creation and editing are omitted due to constraints inherent in the nature of the test.

Some of the operations result in the output of lots of information--too much, in fact, to be given and assimilated by talking. For example, the "display" of an entire message would be prohibitive. Where this occurs, a prepared handout of a simulated display of the information is provided by the consultant at the appropriate time (SLIDE 6).

The commands to perform the operations are "entered" by you simply by speaking them to the "simulator". Since your use of abbreviations, synonyms, punctuation, and so forth are of great interest, the command should be spoken by saying the command or parameter or value, or saying its abbreviation if that is the way you feel that you would type it using a real service. Likewise, the punctuation marks, such as commas, should

also be spoken. ***Since we want to know how you prefer to enter the commands, you are encouraged to freely make substitutions of the words or abbreviations, using those most natural to you. The same is true for punctuation. Also, the order in which parameters are entered is important. Again, enter them in the order that seems most natural to you, disregarding the order in which they appear in their descriptions.***

Sometimes, in performing a sequence of operations, the parameter you wish to use recurs from command to command; it is obvious to you what it should be, and it should be equally obvious to the message service. For example, if you are performing a sequence of operations on a particular message, it seems unnecessary to have to name that message after its first appearance in the command sequence (SLIDE 7). ***We refer to the omission of such recurrent parameters as contextual defaults, and you are encouraged to make use of this feature if it seems natural to you.*** There is a similar case, where you would normally like to use a certain parameter value and have the service understand that you intend its use if you don't say otherwise. Yet it would not be apparent to the service from the context of the dialogue as in the example above. ***We call these programmable defaults and you are encouraged to use them if you wish, by saying to the service that you intend to use a given value for a parameter unless you say otherwise*** (SLIDE 8). Now, in the language descriptions you will receive, there is generally no explicit command to program defaults. However, if it occurs to you at any time that you'd like to have some default, then just tell the simulator (in English) that you are setting a default.

For each unit of work--say as you assume the role of a coordinator, for example--the consultant will give you that task description unit which contains the series of operations to perform, as well as the commands necessary to perform the operations. A brief meaning of the command and each parameter is given along with the structure of the command. [We will look at some examples of these in a few minutes.] Read the task instructions and glance over the commands before beginning the task unit.

## *LANGUAGE FORMS*

### *Exercise Objectives*

Three computer language forms are being used in the exercise. It is important to understand exactly what is and what is not being tested in the exercise, with respect to the languages. We are not trying to determine which form is better, in some sense, than another form. We are merely trying to ascertain the shortcomings and faults so that we can devise languages, within each of the forms, that are adequate for your message handling needs and that reflect your preferences.

### *The Three Language Forms*

The three computer language forms we wish to use are 1) keyword, 2) positional, and 3) English-like (SLIDE 9). For example, in a keyword language one might say AUTHOR = JONES where AUTHOR is the keyword which denotes a class of "objects", namely, those who write or author messages. JONES is then the value of that keyword, i.e., we specifically mean author Jones. In positional notation, the keyword does not appear; we simply say JONES and it is understood that we mean author Jones because of the

position or place within the sequence of parameters in which JONES appears. In English-like we might say BY JONES where the word BY serves the same role as the keyword AUTHOR. Note that, in keyword and English-like, the order in which parameters are given is of no consequence, since their values immediately follow special words like AUTHOR and BY. ***Now, in this exercise, we want you to treat the positional notation in just the same way; that is, you are encouraged to disregard the position of JONES and give each sequence in any order that you like.*** This will allow us to determine what the order should be for your use.

***Commands, Language Descriptions, and Value Definitions***

The commands to perform the various steps of the task are issued to the service by speaking the command followed by the appropriate parameters. The form in which you state parameters depends on which of the language forms you are using.

Let us go over some examples, using an excerpt from the language descriptions you will be using. We shall take the corresponding page from each of the three forms and discuss the examples provided. Accompanying the language descriptions we shall need a description of the permissible values of the parameters.

## *NOTE TO THE LECTURER*

At this juncture, examples are to be discussed with the subjects. In administering this lecture to the ISI group, eleven additional slides were used. Ten of them were essentially reproductions of pages from Appendices C, D, E, and F and thus are not included here. The eleventh, here labeled SLIDE 10, illustrates the message selector.

The following is a guideline of essential points to be covered by the lecturer.

1. Purpose: to discover how the Action Officer wants to use each language
2. Method: by simulating a message processing task
3. Setting
   a) Subjects speak commands
   b) Simulator speaks responses and refers to handouts
   c) Consultant helps subjects when asked
   d) Observer records subjects' preferences
4. Subjects do task three times with different languages
5. Action Officers' aids

a) task instructions

b) language descriptions

c) use of consultant

d) display handouts

6. Task: to handle all phases of a message

7. Stress the following:

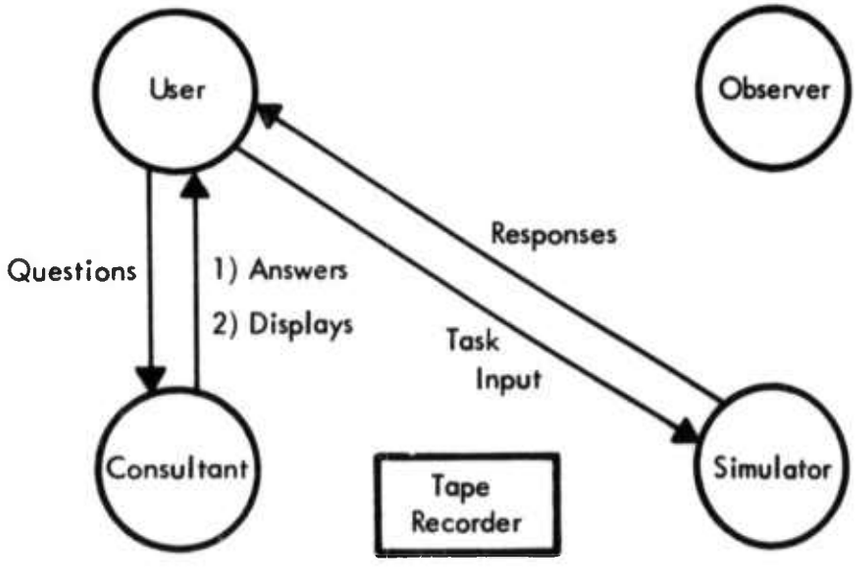
a) no time constraints and no concern for errors

b) composition is encouraged

c) vocabulary substitution is encouraged

d) use of defaults is encouraged

e) repositioning of parameters is encouraged




Slide 1 Setting for simulation of message processing

1. Perform standard task using language 1
2. Perform standard task using language 2
3. Perform standard task using language 3
4. Perform standard task using natural language
5. Perform typical task using natural language

Slide 2 - Conditions for Message Processing Exercise

1. Written task instructions
2. Written description of language being used
3. Handouts of simulated display responses
4. Responses from the simulator
5. Consultant to answer questions and offer advice

Slide 3 - Materials and Services Available for Performing Tasks

1. Author (message creation and coordination)
2. Coordinator (delegate subordinate for coordination)
3. Subordinate coordinator (initial review of message)
4. Author (review coordination, edit, resubmit)
5. Subordinate coordinator (final review)
6. Releaser (transmit)
7. Information recipient (disseminate and hardcopy)
8. Action recipient (delegate action)
9. Action Officer delegated to act on message
10. Author (check message status)

Slide 4 - Active Parties in Message Processing Task

TASK UNIT NO. 3: SUBORDINATE COORDINATOR CAPT. GREEN

1) Accept message for coordination from superior
2) Query the capabilities
3) Display message (assume you add comments)
4) Signoff NG

Slide 5 - Example Task Unit

Message-ID: CINCPAC/November 18, 1974; 10:20/by J6126

Type: Formal

Priority: Routine

Classification: Unclassified

Author: J6126

Releaser (From): J6

Action List: J52

Information List: Col. Smith

Coordination List: J612

Distribution List: J612, J6

Subject: Distinguished Visitor Coming

Body:

Congressman Blake will be visiting Camp Smith to confer with J6, J612, and Col. Smith with regard to operation of the pilot project on communications. Please arrange to transport him from airport to Camp Smith at 0930 on November 23, 1974. Meeting will convent at 1100 in conference room 12, CINCPAC HQ.

Slide 6 - Simulated Display of a Message

NOTE: THIS EXAMPLE IS *NOT* GIVEN IN ANY OF THE TEST LANGUAGE FORMS

Without Default:

Operation 1 - Display message X

Operation 2 - Delete message X

Using Contextual Default:

Operation 1 - Display Message X

Operation 2 - Delete

Slide 7 - Example of Contextual Default

NOTE: THIS EXAMPLE IS *NOT* GIVEN IN ANY OF THE TEST LANGUAGE FORMS.

Without Default:

.
.
.

Operation j - Assign comment and signoff privileges to subordinate coordinator Lt. Jones

.
.
.

Operation k - Assign comment privileges to subordinate coordinator Lt. Jones

Using Programmable Default:

.
.
.

Operation i - When I assign a subordinate coordinator, assume Lt. Jones and assume comment privileges, unless I say otherwise

.
.
.

Operation j - Assign signoff privileges

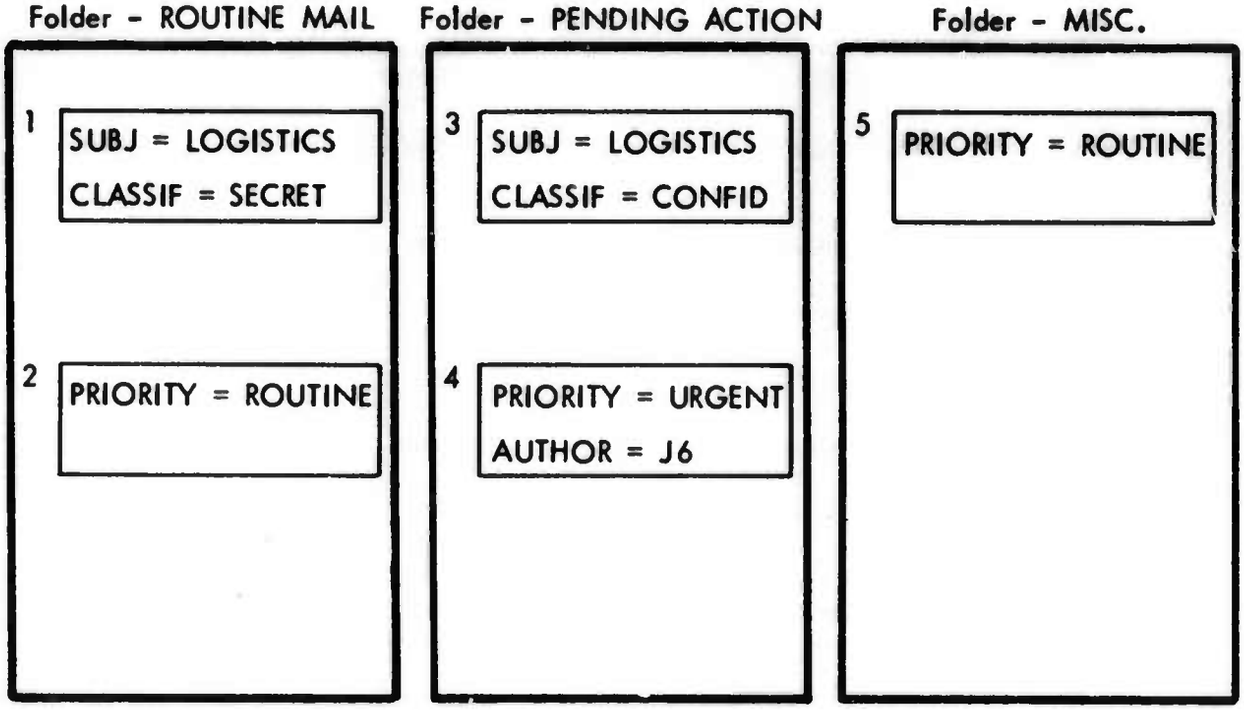
.
.
.

Operation k - Assign

Slide 8 - Example of Programmable Default

| *Language Form* | *Example Parameter and Value* |
|---|---|
| 1. Keyword | ...,AUTHOR=JONES,... |
| 2. Positional | ...,JONES,... |
| 3. English-like | ...BY JONES... |

Slide 9 - Language Forms




ALL (SUBJECT = LOGISTICS, CLASSIFICATION = CONFIDENTIAL) = {3}

ANY (PRIORITY = ROUTINE, AUTHOR = J6), EXCLUDING (FOLDER = ROUTINE MAIL) = {4,5}

Slide 10 Examples of message selectors (keyword format)

# III. THE SIMULATOR

An analyst simulates the message service by providing vocal feedback in response to each command entered by the Action Officer. He should strive for consistency by interpreting with fidelity what the Action Officer says and by emitting similar responses in like situations. Essentially, he can give four kinds of responses:

1. A "oneliner" response providing the information requested.

2. A "see handout" response, which will cue the consultant to display a handout.

3. A brief, yet precise "error statement".

4. An "OK, proceed".

Let us highlight his role by noting what he does not do and what he does.

The simulator does not recognize as an error:

1. Juxtaposition of parameters.

2. Name replacement for commands, parameter keywords, or values. The experiment, in fact, encourages such substitutions by the Action Officer.

The simulator does recognize and report as errors syntactic and semantic anomalies such as:

1. The assumption of a nonexistent function, parameter, or value.

2. Insufficient or incongruous qualification of a value.

3. Wrong interpretation of a command or its results.

4. Inconsistent default value presumed by the Action Officer.

From the point of view of the simulator, the purpose of the experiment is to elicit from the Action Officer his preferred communication style and the particular vocabulary it entails. In keeping with this notion, then, error handling should take the form of polite attention to the error (perhaps addressed to the consultant, who can in turn discuss it with the Action Officer) or perhaps an informative inquiry such as "Did you intend A or B?"

# IV. THE CONSULTANT

The main job of the consultant is to answer questions which the Action Officer asks him. To a lesser extent the consultant also plays the role of an on-line tutor. And in a minor way the consultant aids the simulator by issuing display handouts to the Action Officer.

In the role of consultant, he should be able to answer questions in the following areas:

1. Interpret task instructions.

2. Field language-dependent questions on both the structure of language elements and their interpretation.

3. Interpret message service responses, from both the simulator and display handouts.

4. Explain the user's current state.

Again, the goal of the experiment is to extract information from the user rather than instill in him our way of thinking. Thus the consultant should provide help only when needed or asked for, and provide it in the form of a succinct explanation of the point in question. The following dialogue should illustrate the point (see Instructions for Message Handling Task, Task Unit No. 2, operation no. 2, Appendix B).

Action Officer: "As coordinator J612, do I reference the message by author or subject as shown or can I just call it BIGWIG?"

Consultant: "In general you can reference it by any of its fields whose values are both accessible to you and known by you, such as author or subject. The name BIGWIG was given to the message by the author; it is private and can only be used by the author. However, you also can name the message anything you like, for your personal reference as coordinator J612."

# V. THE OBSERVER: DEPENDENT VARIABLES

## *OBSERVATIONS AND THE CHECKLIST*

The duties of the observer are described in terms of the materials that he uses. For each condition of the Latin square design (that is, each session with one of the respondents), the observer has at hand the following materials, as aids for scenario observation and comment:

1. Tape channel for comments.

2. Task instructions (see Appendix A).

3. Command language description (see Appendix A).

4. Observation checklist (see Fig. V-1).

Item 1 was described in the Introduction, and items 2 and 3 were described in Chapter II. The observation checklist (Fig. V-1) serves as a record of the scenario in terms of those dependent variables identified as properties to be observed. The observer fills in the checklist based on what the Action Officer says and does as he "executes" the standard task. The format and use of the checklist is as follows:

1. The checklist as shown in Fig. V-1 is in the form of a matrix whose ith row is in one-one correspondence with the ith expected command of the standard task.

2. Each row has k columns, each of which express some property to be observed.

3. As the Action Officer "enters" the ith command, the observer compares it to the appropriate syntactic form shown in item 3 above.

4. As the Action Officer "enters" the ith command, the observer fills in the appropriate columns of row i with numerical values which represent the number of occurrences or the degree of change of that property in the command, as entered. This is explained in more detail below.

Vocabulary changes are encoded as follows. Consider the command to delete a message. In the language descriptions, several alternatives for the command name are given, viz., DESTROY, PURGE, REMOVE, and DELETE. If the user had no prior tendency to use, say PURGE, or perhaps some other command name not listeo among the alternatives, then we might expect that he would more often than not enter the first command name given, namely DESTROY. To encode his choice we assign the integer 0 to the first name appearing in the description, the integer 1 to all other alternatives, and the integer 2 to any new command name which the user might substitute.

Task No. _____ Action Officer_____ Condition No. _____

Transcript No. ____ Group No. _____

OBSERVATIONS

| Operation No. | Vocabulary Changes: Parameters | Command | Delimiters | Values | Noise Words | Errors | Word Omissions | Reordering | Abbreviations | References | Advice | Contextual | Defaults: Programmable | Composition | Transcript Reference: Clock Value |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | | | | | | | | | | | | | | | |
| 2 | | | | | | | | | | | | | | | |
| 3 | | | | | | | | | | | | | | | |
| 4 | | | | | | | | | | | | | | | |
| 5 | | | | | | | | | | | | | | | |
| 6 | | | | | | | | | | | | | | | |
| 7 | | | | | | | | | | | | | | | |
| 8 | | | | | | | | | | | | | | | |
| 9 | | | | | | | | | | | | | | | |
| 10 | | | | | | | | | | | | | | | |

REMARKS:

Figure V-1 Checklist of observations

Composition of commands is coded as the integer 1, for *each* command making up the composite. For example, if commands i, j, and k are combined as command i, then rows i, j and k will each be encoded as a 1.

The clock value serves as a pointer into the tape transcript for the language designer where something noteworthy was observed. To the extent that transaction timing is measured and analyzed, it will be taken from the transcript at a later time.

All other variables are coded as the sum of the number of changes of that particular kind in the command.

### *DEPENDENT VARIABLES*

To be precise concerning the metric being analyzed in Chapter VII, each dependent variable is defined below.

1. Vocabulary changes

   a. to parameter names: A parameter name is a special identifier (keyword) associated with arguments (parameters) in a command. For example, in the parameter fields AUTHOR JONES or BY JONES, the words AUTHOR and BY are parameter names identifying Jones as the writer of a message. The metric applied in our study is the number of substitutions of names of parameters.

   b. to command names: A command name is the verb in the command, such as the word SEND in SEND MESSAGE. The metric under study is the number of substitutions of command names.

   c. to values of parameters: The value of a parameter is the particular datum (or data) associated with the parameter name. In the example in 1a, JONES is the value of the parameter names AUTHOR and BY. In the instruction and language manuals available to the subjects, certain values are supplied and suggested. The number of substitutions of such values are measured.

   d. to noise words: Noise words are those words entered by a subject (for clarity, readability, etc., to himself and others) that are not essential to correctly parsing and interpreting the command. For example, in, ---- THE MESSAGE ---- , the definite article is a noise word. The metric under study is the number of noise words used other than those suggested.

   e. to delimiters: Delimiters are the punctuation marks and coordinating conjunctions in a command. For example, in, ---- RECIPIENTS JONES, SMITH AND BROWN ---- , the delimiters

are the comma and the AND. The metric is the number of changes to suggested delimiters.

2. Errors: Since the purpose of the exercise is for the respondent to suggest changes to vocabulary and syntax, such elements of the command which would be in error in a fixed language are not so interpreted here. The types of inputs recognized as errors are those enumerated in Chapter III. The metric is the number of occurrences of such errors.

3. Keyword omissions and service prompts: The omission of a keyword is the failure to enter any parameter name where one or more was (were) given in the language manual. Service prompts are requests by the user for the service (simulator) to supply a correct next keyword in the command. The metric is a tally of explicit omissions and requests.

4. Reordering of parameters: Reordering is the juxtaposition of fields (i.e., argument names and accompanying values) in a command. From the standpoint of presentation of a language to a user, this is considered equally important in each form. The metric is the amount of reordering or sequential movement of parameter names and their values.

5. Abbreviations of language elements: Abbreviations take the form of initial substrings of words, standard abbreviations, and acronyms. The metric is the number of abbreviations to any vocabulary elements.

6. References to materials: References to materials means, upon attempting to compose an operation, the user examines or scans the language manual as an aid in recalling the syntax, semantics or vocabulary needed to correctly construct the command. The metric is the number of such references. Note that the users are given language manuals in sections, as needed, each of which corresponds to the operations of a given task unit. The users are requested to examine the commands prior to beginning each unit.

7. Advice from consultant: Advice from consultant means either when the user asks the consultant a question relevant to the command language or the consultant spontaneously offers advice. Where the subjects requested clarification on the military's meaning of some function, this is not considered as advice. The metric is the number of instances of advice given.

8. **Contextual defaults:** Defaults are the omission of a parameter (name and value) by the user upon entering a command, where the service then supplies the necessary parameter. Contextual defaults are those that are known to the user and to the service because of the setting established by a preceding sequence of dialogue. The metric is the number of such defaults.

9. **Programmable defaults:** Programmable defaults are those defaults explicitly and earlier identified by the user to the service as values to be assumed if no value is given in the applicable situation. The metric is the number of ***uses*** (as opposed to definitions) of such defaults.

10. **Composition of commands:** The tasks performed by the user consist of elementary operations where each operation corresponds to a command as defined in the language manual. The respondent may combine several sequential operations into a single command. The metric is the number of elementary operations so combined.

# VI. THE EXPERIMENT DESIGN

## *ANALYSIS OF VARIANCE*

The analysis of variance lends itself to testing the research hypotheses which collectively state that the languages are not equally well suited to the users and the task, according to the criteria variables observed. The hypotheses are given in Chapter VII and the dependent variables are given in Chapter V.

## *LATIN SQUARE DESIGN*

A Latin square design [WINER 71] is used for the protocol analysis.* It was chosen to counterbalance order effects while reducing the number of conditions needed for a factorial design. The model for an observation in cell ikj is

$$\mathbf{X}_{ijk} = \mu + \alpha_i + \beta_j + \gamma_k + res + \epsilon$$

where $\mu$ = true score, a fixed but unknown constant,
$\alpha$ = group to which individual belongs,
$\beta$ = order of language conditions
$\gamma$ = language forms
and res + $\epsilon$ = residual variation and errors.

In fact, each group should represent a random sample from the population under study. Clearly, the ordering is an artificial variable. Then, there should be no intrinsic interest in factors $\alpha$ and $\beta$ and the model should reduce to

$$\mathbf{X} = \mu + \gamma + res + \epsilon.$$

## *VARIANCE ESTIMATION AND MEAN COMPARISONS*

See [WINER 71] for the analysis of variance needed to compute F-ratios for main effects.

The observer's checklist (see Chapter V) lists the dependent variables for which data is taken. Significant F-ratios indicate a difference in the suitability among the languages (according to the variable under analysis). The first step of our analysis involves variance analysis for each $v_i \in V$ (variables). Where significant differences are found, the means of the independent variable levels (languages) are compared pairwise. For comparison between means, consider

---

*The natural language conditions mentioned in the Introduction were not carried out in the "dry run".

$\Sigma$ a<i>T<i>

where $\Sigma$ a<i> = 0 and a<i> are coefficients which determine the means to be compared. That is, to compare means T<1> and T<3>, a<1> and a<3> are set to 1 and -1 and a<2> is set to zero. Standard scores (Z-scores) are computed for the comparison of means. Where further inquiry is suggested by the data, these steps are repeated for each individual operation.

The variance analysis can be summarized by the following steps.

1. Determine main effects for each variable.

2. Make pairwise comparison of means from step 1.

3. Determine main effects for each variable for each operation.

4. Make comparison of means.

Where it is more illuminating, the data are represented in graphical form, see Chapter VII.

### *SUPPLEMENTARY CORRELATIONS*

The analysis of variance is supplemented by congeries of correlations and attendant scatter diagrams, no one of which may have a profound effect on the target languages but which should noticeably improve the resulting languages when considered collectively by the designer. The Pearson *r* is used. [GUILFORD 73].

## VII. FINDINGS AND DISCUSSIONS

The findings* of the task exercise portion of the protocol analysis study are presented here as an adaptation of one particular format [MICHAEL]. They consist of 14 hypotheses. With each hypothesis is given the underlying assumptions, the results of the data analysis, and conclusions. The conclusions attempt to interpret or explain the results; decisions are also given to indicate how they might be used by the practitioner. To illustrate the format used we shall "walk through" the first hypothesis and explain each step.

### *EXAMPLE: RESEARCH HYPOTHESIS 1: PARAMETER NAMES*

The first step is a declarative statement of the research hypothesis.

Research Hypothesis 1

***There will be a significant difference in the average number of parameter name changes among the three languages. The English-like and keyword will be changed to a significant extent contrasted with positional.***

We have stated a one-tailed hypothesis, which means that we predict not only a difference among languages but also the direction of that difference, namely, changes to English-like and changes to keyword will be greater than changes to positional. As a complement to the research hypothesis, often the Null hypothesis is stated as well. It is a composite of all outcomes other than that predicted by the research hypothesis. Since the Null hypothesis can be directly inferred from the research hypothesis, we will omit this redundancy.

The next step of our format will be a statement of reason or motivation for choosing the research hypothesis. Typically, the reasoning appears earlier in the problem study and leads up to the hypothesis. It is inserted here to provide continuity and understandability in reading the findings.

Assumptions

***Individual preferences coupled with the fact that some parameter names are argot for the military Action Officers and alien to ISI computer professionals will result in significant changes. The ordering of English-like, keyword, and positional reflect the naturalness of those languages, hence the degree to which the subject will readily mold them to natural language. Note also that in positional, the parameters (i.e., keywords) appear only where further qualification of a value is required [although in the task exercise (Appendix B) further qualification was frequently necessary].***

---

*These findings apply to a specific population of experienced computer professionals.

At this stage either an acceptable level of significance is announced or the researcher states that the level of significance attained will simply be observed. Since so little precedence is available, good judgment dictates that the level attained will only be observed, not forecasted. Next the statistical model is described. We use the analysis of variance model identified in Chapter VI. Where significance is found among the languages, pairwise comparisons of means will be shown. [In mean calculations, frequently an observation may take on an extreme value relative to all the other observations. Such "outliers" are commonly caused by, for example, the subject not following instructions because they were not presented clearly, or for some other reason. Where extreme deviants are present they will be trimmed (as is acceptable in statistical analysis in other research areas) by discarding the highest and also the lowest value.] A confidence interval or probability statement generally follows the resultant statistical calculations. Equivalently, we shall show the standard deviations and standard error.

As a frame of reference for the analysis of variance F ratios, the following are threshold values for significance at several levels between 0.75 and 0.99. The notation [WINER 71], using F.75 (2, 18) = 1.50 to illustrate, means that our data have 2 degrees of freedom in the numerator and 18 degrees of freedom in the denominator. A value equal to or greater than 1.50 must be realized for significance at the 0.75 level. The 0.75 level means that only 1 time in 4 (i.e., 1 - 0.75 = 0.25 = 1/4) would we expect to get a value as large as 1.50 purely by chance.

F.75 (2, 18) = 1.50
F.90 (2, 18) = 2.62
F.95 (2, 18) = 3.55
F.99 (2, 18) = 6.01

For the comparison of means, significant Z values are shown below for one-tailed tests. [Z values are *standard scores* which allow us to assign a standard meaning to the values from different kinds of tests.]

Z.75 = 0.67
Z.90 = 1.28
Z.95 = 1.64
Z.99 = 2.33

Calculations

***Table VII-1a shows the observed data and analysis of variance statistics.*** Looking at the observed data section, languages 1, 2, and 3 represent keyword, positional, and English-like, respectively. The sources of variation are identified as the three groups, the three languages, order (an artificial variable with no intrinsic interest), the residual (which represents variation due to experimental error and untested factors), and the within cell variation. The sums of squares (SS), degrees of freedom (DF), and the mean squares (MS) are intermediate calculations; thus they can be ignored. Of interest is the column labeled F which shows the F ratios. Specifically, we are interested in the F

ratio of variation due to differences among languages. ***In our example (Table VII-1a) notice that F = 7.12 which is significant beyond the 0.99 level of F = 6.01 (from above).*** Since the F ratio for languages is highly significant (we would expect a value of 7.12 less than one time in one hundred times by chance) we proceed with the pairwise comparison of means (Tables VII-1b, c, and d).

The next step or steps generally involve(s) the statement of a triad such as (in our example) we reject the Null hypothesis, which implies that the research hypothesis is tenable, which implies that there is reason to believe that a real difference exists among the mean values of the changes to parameters. To eliminate the redundancy of the triad, we simply state whether or not the research hypothesis is tenable.

Returning to our example, Table VII-1b compares the mean of positional to the mean of English-like. The means (M), standard deviations (SD), standard error of the difference between means (SE), and the Z value are given. Figure VII-1a illustrates the interpretation of the Z score of -3.45. That is, we would not expect the difference in the average number of changes in the English-like language compared to the average number of changes in the positional language to be as great as (13.71) - (4.00) = 9.71. This value would occur less than one time in one hundred due to chance alone. ***Figure VII-1b illustrates the distributions and mean values from Table VII-1b. The average number of changes in English-like is much higher than in positional. In Table VII-1c we note that the comparison of keyword to positional is significant just beyond the 0.95 level, and in Table VII-1d there is no significant difference between English-like and keyword.***

The final step, and most difficult one, is to state the practical implications of the results.

Conclusions

***There is reliable evidence of a differential between the means. If in fact we were really interested in designing languages of these forms for the population represented by the sample groups of ISI computer professionals, then we would suggest that the language designer give careful consideration to modifications of parameter names in the English-like and keyword versions. We would then provide an analysis of variance for each command type (rather than the language as a whole). In addition, we would supply the designer with the task input commands for those language elements (in English-like and in keyword) exhibiting significant substitutions.***

The remainder of the research hypotheses follow in much the same format as our example above. In some instances the data are also presented in graphic form (e.g., trend lines, histograms, scatter diagrams (with correlation coefficients)) where further exploration is indicated to illuminate the ramifications of the calculations.

OBSERVED DATA

| SUBJECTS | | | GROUP | LANGUAGE |
|---|---|---|---|---|
| 13.00 | 17.00 | 37.00 | 1 | 1 |
| 2.00 | 4.00 | 16.00 | 1 | 2 |
| 15.00 | 14.00 | 29.00 | 1 | 3 |
| 30.00 | 13.00 | 18.00 | 2 | 1 |
| 2.00 | 6.00 | 4.00 | 2 | 2 |
| 17.00 | 11.00 | 14.00 | 2 | 3 |
| 7.00 | 14.00 | 6.00 | 3 | 1 |
| 2.00 | 2.00 | 8.00 | 3 | 2 |
| 12.00 | 2.00 | 13.00 | 3 | 3 |

| SOURCE OF VARIATION | SS | DF | MS | F |
|---|---|---|---|---|
| GROUPS | 369.85 | 2 | 184.93 | 3.70** |
| LANGUAGES | 712.07 | 2 | 356.04 | 7.12*** |
| ORDER | 32.07 | 2 | 16.04 | 0.32 |
| RESIDUAL | 90.74 | 2 | 45.37 | 0.91 |
| WITHIN CELL | 900.67 | 18 | 50.04 | 1.00 |

Table VII-1a Vocabulary changes to parameters -- over all commands

OBSERVED DATA

| POSITIONAL | ENGLISH |
|---|---|
| 2.00 | 15.00 |
| 4.00 | 14.00 |
| 2.00 | 17.00 |
| 6.00 | 11.00 |
| 4.00 | 14.00 |
| 2.00 | 12.00 |
| 8.00 | 13.00 |

M(1) = 4.00 SD(1) = 2.14
M(2) = 13.71 SD(2) = 1.83
SE = 2.81
Z = -3.45***

Table VII-1b Vocabulary-parameters (all commands)

OBSERVED DATA

| KEYWORD | POSITIONAL |
|---|---|
| 13.00 | 2.00 |
| 17.00 | 4.00 |
| 30.00 | 2.00 |
| 13.00 | 6.00 |
| 18.00 | 4.00 |
| 14.00 | 2.00 |
| 6.00 | 8.00 |

M(1) = 15.86 SD(1) = 6.79
M(2) = 4.00 SD(2) = 2.14
SE = 7.12
Z = 1.67**

Table VII-1c Vocabulary-parameters (all commands)

OBSERVED DATA

| ENGLISH | KEYWORD |
|---|---|
| 15.00 | 13.00 |
| 14.00 | 17.00 |
| 17.00 | 30.00 |
| 11.00 | 13.00 |
| 14.00 | 18.00 |
| 12.00 | 7.00 |
| 13.00 | 6.00 |

M(1) = 13.71 SD(1) = 1.83
M(2) = 14.86 SD(2) = 7.47
SE = 7.69
Z = -0.15

Table VII-1d Vocabulary-parameters (All commands)

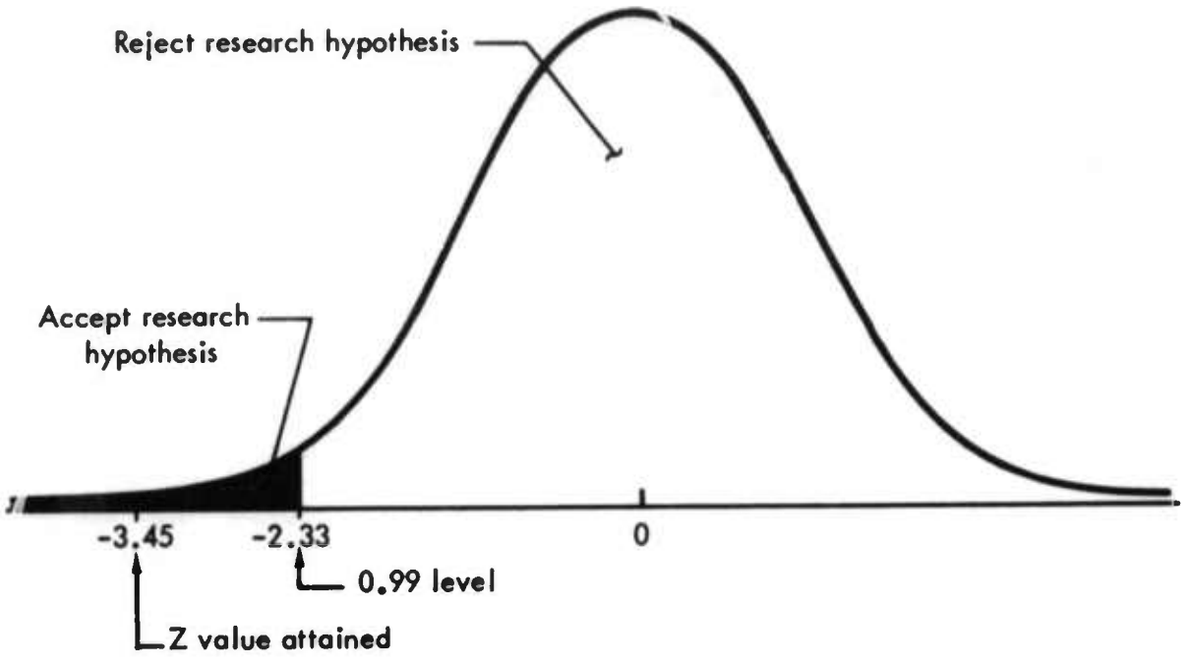



Figure VII-1a Interpretation of Z score

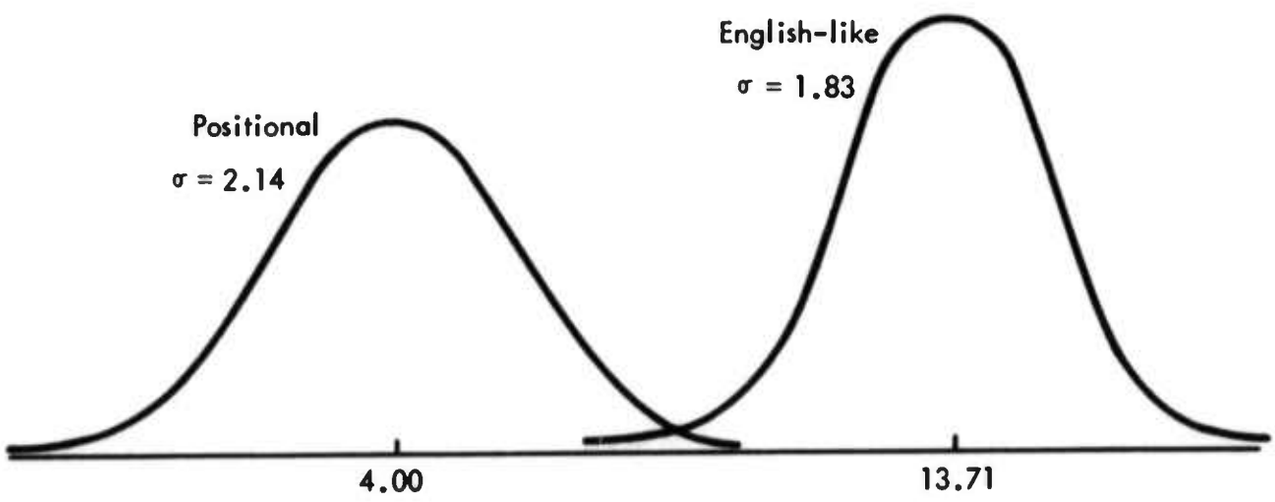



Figure VII-1b Distribution of scores

## *RESEARCH HYPOTHESIS 2: COMMAND NAMES*

There will be a difference in the average number of ***command name changes*** among the three languages. Although many changes will be made to each language, the significance, in decreasing order, will be English-like, keyword, and positional.

Assumptions

The reasoning is the same as in parameter name changes (hypothesis 1). Overall, we anticipate change to each language because of individual taste; thus subjects will likely supply their preference rather than consult the manual (Appendices C, D, and E).

Calculations

Table VII-2a shows no significant difference (F = 1.06) among the languages when all commands are considered. However, a glance at the observed data indicates that many vocabulary changes were made, as expected. Thus we suspect that the (language-independent) changes are perhaps command- or operation-dependent. Additional probing is called for. Firstly, the amount of change is plotted against the command types, as shown in Fig. VII-2. In the task exercise, command name choices were presented as a preferred command along with several alternatives. It is reasonable to expect that if the subject had no initial preference, he would more often than not choose the first (preferred) command name given. When the subject chose the preferred command, he was assigned a score of 0. When he chose an alternate he was assigned a score of 1. If, on the other hand, he substituted a command name in place of those offered, this was adjudged to be a very meaningful name to him and his score was weighted as a 2. Since some commands (e.g., DISPLAY) appeared more frequently in the exercise than others, usage is normalized on the ordinal axis of Fig. VII-2. The equation for normalization is

$$\frac{(\#\text{ alternates used}) + 2*(\#\text{ new substitutions})}{\#\text{instances of this kind of command}}$$

Aside from the operation-by-operation differences, an analysis of variance was performed on each command type. Only the FANOUT command (Table VII-2b) was found to be significant with respect to languages. The significance level is low and we would in fact expect to get significance at this level on about one of the 19 command types by chance.

Conclusions

The research hypothesis is not tenable. In examining Fig. VII-2, the maximum amount of change that could have occurred for a given command type is (9 subjects) x (3 languages) x (2 units change) = 54. Thus one subject could effect at most 6 units of change. We would recommend that the language designer consider at least those commands with a total change of more than 12 units (which implies that at least 3 subjects contributed to the change). By studying the task input commands for the command types, the designer would discover the following.

1. NOTE was frequently replaced by ALERT.

2. ASSIGN was frequently replaced by DELEGATE.

3. SIGNOFF was frequently replaced by OK or NG.

4. RELEASE was combined with SEND and called SEND.

5. CREATE (folder) was most frequently called CREATE, DEFINE, or BUILD.

6. FANOUT was called DISTRIBUTE or SEND.

7. SELL was replaced by a diversity of names such as TRANSFER, FORWARD, TO, GIVE, etc.

8. COMPLETED was called FINISHED and also a variety of other names.

9. DESTROY was called DELETE.

If the language designer examined the commands at an instance-of-use level, then it would also be apparent that the DISPLAY command was renamed to be the field name in the cases of status and capabilities, and often for signoff.

As to why no differences exist among languages, the command name is the verb of the command and also the first word in the command, and it would appear that the subject chooses a word to describe the action and that this single assimilation is carried across languages.

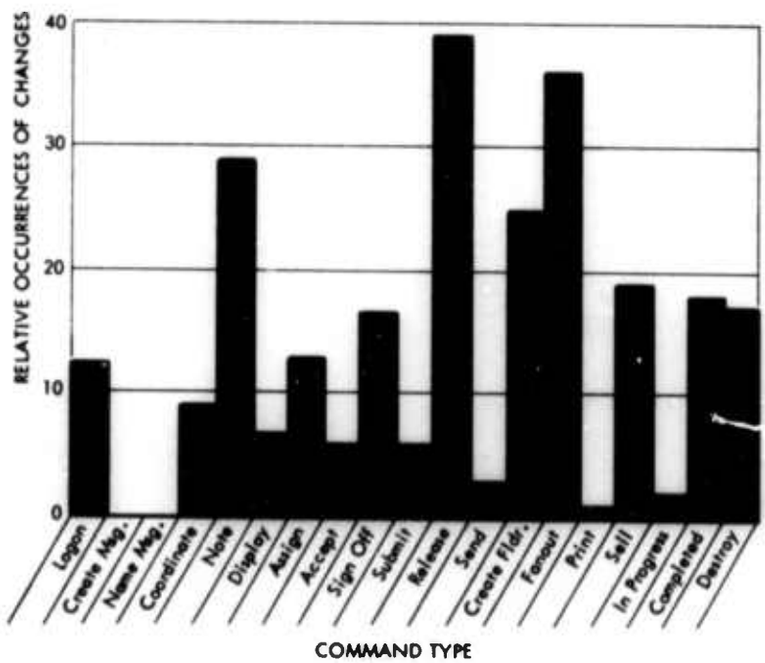



Figure VII-2 Occurrences of changes to command names

OBSERVED DATA

| SUBJECTS | | | GROUP | LANGUAGE |
|---|---|---|---|---|
| 8.00 | 12.00 | 20.00 | 1 | 1 |
| 8.00 | 13.00 | 15.00 | 1 | 2 |
| 7.00 | 9.00 | 17.00 | 1 | 3 |
| 25.00 | 21.00 | 13.00 | 2 | 1 |
| 18.00 | 16.00 | 15.00 | 2 | 2 |
| 25.00 | 37.00 | 14.00 | 2 | 3 |
| 18.00 | 24.00 | 20.00 | 3 | 1 |
| 14.00 | 8.00 | 21.00 | 3 | 2 |
| 22.00 | 20.00 | 10.00 | 3 | 3 |

| SOURCE OF VARIATION | SS | OF | MS | F |
|---|---|---|---|---|
| GROUPS | 320.67 | 2 | 160.33 | 4.20** |
| LANGUAGES | 80.67 | 2 | 40.33 | 1.06 |
| ORDER | 14.00 | 2 | 7.00 | 0.18 |
| RESIDUAL | 98.00 | 2 | 49.00 | 1.28 |
| WITHIN CELL | 686.67 | 18 | 38.15 | 1.00 |

Table VII-2a Vocabulary changes to commands -- over all commands

OBSERVED DATA

| SUBJECTS | | | GROUP | LANGUAGE |
|---|---|---|---|---|
| 1.00 | 1.00 | 2.00 | 1 | 1 |
| 1.00 | 1.00 | 1.00 | 1 | 2 |
| 1.00 | 1.00 | 1.00 | 1 | 3 |
| 2.00 | 1.00 | 2.00 | 2 | 1 |
| 2.00 | 0.00 | 2.00 | 2 | 2 |
| 2.00 | 1.00 | 2.00 | 2 | 3 |
| 2.00 | 2.00 | 2.00 | 3 | 1 |
| 1.00 | 1.00 | 0.00 | 3 | 2 |
| 2.00 | 2.00 | 0.00 | 3 | 3 |

| SOURCE OF VARIATION | SS | DF | MS | F |
|---|---|---|---|---|
| GROUPS | 0.89 | 2 | 0.44 | 1.00 |
| LANGUAGES | 2.00 | 2 | 1.00 | 2.25 |
| ORDER | 0.22 | 2 | 0.11 | 0.25 |
| RESIDUAL | 0.89 | 2 | 0.44 | 1.00 |
| WITHIN CELL | 8.00 | 18 | 0.44 | 1.00 |

Table VII-2b Vocabulary changes to commands -- fanout command

OBSERVED DATA

| POSITIONAL | ENGLISH |
|---|---|
| 31.00 | 4.00 |
| 39.00 | 4.00 |
| 25.00 | 5.00 |
| 31.00 | 1.00 |
| 32.00 | 3.00 |
| 24.00 | 1.00 |
| 12.00 | 5.00 |

M(1) = 27.71 SD(1) = 7.89
M(2) = 3.29 SD(2) = 1.58
SE = 8.04
Z = 3.04***

Table VII-3b Vocabulary-delimiters (all commands)

OBSERVED DATA

| ENGLISH | KEYWORD |
|---|---|
| 4.00 | 24.00 |
| 4.00 | 79.00 |
| 5.00 | 13.00 |
| 1.00 | 11.00 |
| 5.00 | 42.00 |
| 3.00 | 10.00 |
| 1.00 | 37.00 |

M(1) = 3.29 SD(1) = 1.58
M(2) = 30.86 SD(2) = 22.91
SE = 22.97
Z = -1.20

Table VII-3c Vocabulary-delimiters (all commands)

OBSERVED DATA

| KEYWORD | POSITIONAL |
|---|---|
| 41.00 | 35.00 |
| 24.00 | 31.00 |
| 13.00 | 25.00 |
| 11.00 | 31.00 |
| 42.00 | 12.00 |
| 10.00 | 32.00 |
| 37.00 | 24.00 |

M(1) = 25.43 SD(1) = 13.38
M(2) = 27.14 SD(2) = 7.16
SE = 15.18
Z = -0.11

Table VII-3d Vocabulary-delimiters (all commands)

## *RESEARCH HYPOTHESIS 3: DELIMITERS*

***Changes to delimiters*** (i.e., punctuation) will be significant across languages. Both keyword and positional will be greater than English-like.

Assumptions

Keyword involves the heaviest use of delimiters. Delimiters of other keyword languages in use vary from one language to the next; thus there is no reason to suspect the ones chosen for the strawman language will be the subjects' choices. In positional, commas separate parameters and parentheses set off sublists. These may perhaps be more natural than the keyword delimiters, but not significantly so. Blank spaces are used for field separators in English-like; these are natural and will not change.

Calculations

Table VII-3a shows an extremely significant difference of F = 11.12 among languages. The Z values (Tables VII-3b, c, and d) indicate that both positional (3.04) and keyword (1.20) are significant as compared to English-like.

Conclusions

There is very strong evidence to support the research hypothesis of real differences (among languages) of changes to delimiters. By and large the attitude of the respondents was that typing should be easy and minimal even at the expense of the appearance (readability) of the command. This was probably largely due to their experience with the computer system at ISI and that they were experienced programmers, consequently confident of the correctness of their inputs, thus indifferent to readability. Thus in positional the space bar was used primarily, where the command was parsable. Several different delimiters were used in keyword by the various subjects, yet in each instance a lower instance punctuator was selected, near the home position on the keyboard. It would appear that the need for changes in punctuation in positional and keyword far surpasses the need for other vocabulary modifications.

OBSERVED DATA

| SUBJECTS | | | GROUP | LANGUAGE |
|---|---|---|---|---|
| 41.00 | 24.00 | 79.00 | 1 | 1 |
| 35.00 | 31.00 | 39.00 | 1 | 2 |
| 0.00 | 4.00 | 4.00 | 1 | 3 |
| 13.00 | 11.00 | 42.00 | 2 | 1 |
| 25.00 | 31.00 | 12.00 | 2 | 2 |
| 5.00 | 1.00 | 5.00 | 2 | 3 |
| 10.00 | 37.00 | 11.00 | 3 | 1 |
| 32.00 | 24.00 | 12.00 | 3 | 2 |
| 3.00 | 1.00 | 5.00 | 3 | 3 |

| SOURCE OF VARIATION | SS | DF | MS | F |
|---|---|---|---|---|
| GROUPS | 1019.56 | 2 | 509.78 | 2.95* |
| LANGUAGES | 3840.67 | 2 | 1920.33 | 11.12*** |
| ORDER | 440.22 | 2 | 220.11 | 1.27 |
| RESIDUAL | 350.89 | 2 | 175.44 | 1.02 |
| WITHIN CELL | 3109.33 | 18 | 172.74 | 1.00 |

Table VII-3a Vocabulary changes to delimiters -- over all commands

## *RESEARCH HYPOTHESIS 4: VALUES*

The average number of ***changes to values*** in English-like will exceed that of keyword and positional, but not significantly so.

Assumptions

In English-like the subject will tend to state the command more naturally, which implies that he will choose values most meaningful to himself. Significance is not expected due to the limited range of operations in the exercise, hence the restricted opportunity to use different values.

Calculations

As shown in Table VII-4a, the F ratio of 1.58 is significant between the 0.75 and 0.90 levels. Tables VII-4b and 4c show no significant Z.

Conclusions

The research hypothesis is supported and Tables VII-4b and 4c show that Z is going in the expected direction, that is, more value substitutions took place in English-like. Recalling the assumption above, we still feel that this variable may be more important in an operational environment than the results show. It is recommended that a more explicit test of the variable be planned before discounting it.

OBSERVED DATA

| SUBJECTS | | | GROUP | LANGUAGE |
|---|---|---|---|---|
| 11.00 | 11.00 | 11.00 | 1 | 1 |
| 10.00 | 5.00 | 5.00 | 1 | 2 |
| 12.00 | 7.00 | 5.00 | 1 | 3 |
| 11.00 | 2.00 | 7.00 | 2 | 1 |
| 11.00 | 7.00 | 3.00 | 2 | 2 |
| 11.00 | 4.00 | 10.00 | 2 | 3 |
| 6.00 | 4.00 | 1.00 | 3 | 1 |
| 5.00 | 8.00 | 2.00 | 3 | 2 |
| 16.00 | 11.00 | 6.00 | 3 | 3 |

| SOURCE OF VARIATION | SS | DF | MS | F |
|---|---|---|---|---|
| GROUPS | 18.30 | 2 | 9.15 | 0.73 |
| LANGUAGES | 39.41 | 2 | 19.70 | 1.58 |
| ORDER | 21.41 | 2 | 10.70 | 0.86 |
| RESIDUAL | 64.96 | 2 | 32.48 | 2.60 |
| WITHIN CELL | 224.67 | 18 | 12.48 | 1.00 |

Table VII-4a Vocabulary changes to values -- over all commands

OBSERVED DATA

| POSITIONAL | ENGLISH |
|---|---|
| 10.00 | 12.00 |
| 5.00 | 7.00 |
| 5.00 | 5.00 |
| 11.00 | 11.00 |
| 7.00 | 4.00 |
| 3.00 | 10.00 |
| 5.00 | 16.00 |
| 8.00 | 11.00 |
| 2.00 | 6.00 |

M(1) = 6.22 SD(1) = 2.86
M(2) = 9.11 SD(2) = 3.66
SE = 4.65
Z = -0.62

Table VII-4b Vocabulary-value
(all commands)

OBSERVED DATA

| ENGLISH | KEYWORD |
|---|---|
| 12.00 | 11.00 |
| 7.00 | 11.00 |
| 5.00 | 11.00 |
| 11.00 | 11.00 |
| 4.00 | 2.00 |
| 10.00 | 7.00 |
| 16.00 | 6.00 |
| 11.00 | 4.00 |
| 6.00 | 1.00 |

M(1) = 9.11 SD(1) = 3.66
M(2) = 7.11 SD(2) = 3.87
SE = 5.33
Z = 0.38

Table VII-4c Vocabulary-value
(all commands)

OBSERVED DATA

| KEYWORD | POSITIONAL |
|---|---|
| 11.00 | 12.00 |
| 11.00 | 7.00 |
| 11.00 | 5.00 |
| 11.00 | 11.00 |
| 2.00 | 4.00 |
| 7.00 | 10.00 |
| 6.00 | 16.00 |
| 4.00 | 11.00 |
| 1.00 | 6.00 |

M(1) = 7.11 SD(1) = 3.87
M(2) = 9.11 SD(2) = 3.66
SE = 5.33
Z = -0.38

Table VII-4d Vocabulary-value
(all commands)

## *RESEARCH HYPOTHESIS 5: NOISE WORDS*

Vocabulary ***changes of noise words*** will be more prevalent in English-like.

Assumptions

To state the command more naturally some additional noise words will be added in the English-like language. Since an ample supply of noise words already appears in the English-like language, this will not be highly significant. The structure of keyword and positional, as more canonical-like languages, are not as conducive to addition of noise words.

Calculations

The value of 3.14 in Table VII-5a is significant and along with the Z values of +1.28 and -1.54 from Tables VII-5c and 5d, the research hypothesis is supported at the 0.90 level. Also (Table VII-5b) there is no meaningful difference between the means 4.71 and 4.57 of of keyword ***versus*** positional. However, all of the F values in Table VII-5a are inexplicably high, especially the residual error term of 4.88.

Conclusions

Although the research hypothesis is supported at about the anticipated level, these data are highly suspect because of the residual error. It is suggested that the practitioner construct a frequency distribution of the added noise words in English-like as a guideline for inclusion in the vocabulary. This increase in vocabulary size is not considered a serious problem, since the totality of noise words is small and the penalty paid in the parse is negligible.

OBSERVED DATA

| SUBJECTS | | | GROUP | LANGUAGE |
|---|---|---|---|---|
| 9.00 | 5.00 | 20.00 | 1 | 1 |
| 6.00 | 3.00 | 13.00 | 1 | 2 |
| 7.00 | 8.00 | 6.00 | 1 | 3 |
| 1.00 | 3.00 | 1.00 | 2 | 1 |
| 5.00 | 5.00 | 8.00 | 2 | 2 |
| 13.00 | 13.00 | 11.00 | 2 | 3 |
| 2.00 | 8.00 | 5.00 | 3 | 1 |
| 2.00 | 3.00 | 5.00 | 3 | 2 |
| 10.00 | 10.00 | 6.00 | 3 | 3 |

| SOURCE OF VARIATION | SS | DF | MS | F |
|---|---|---|---|---|
| GROUPS | 38.74 | 2 | 19.37 | 1.58 |
| LANGUAGES | 76.74 | 2 | 38.37 | 3.14* |
| ORDER | 56.07 | 2 | 28.04 | 2.29 |
| RESIDUAL | 119.41 | 2 | 59.70 | 4.88** |
| WITHIN CELL | 220.00 | 18 | 12.22 | 1.00 |

Table VII-5a Vocabulary changes to noise words -- over all commands

OBSERVED DATA

| KEYWORD | POSITIONAL |
|---|---|
| 9.00 | 6.00 |
| 5.00 | 3.00 |
| 3.00 | 5.00 |
| 1.00 | 8.00 |
| 2.00 | 2.00 |
| 8.00 | 3.00 |
| 5.00 | 5.00 |

M(1) = 4.71 SD(1) = 2.76
M(2) = 4.57 SD(2) = 1.92
SE = 3.36
Z = 0.04

Table VII-5b Vocabulary-noise words (all commands)

OBSERVED DATA

| ENGLISH | KEYWORD |
|---|---|
| 7.00 | 9.00 |
| 8.00 | 5.00 |
| 13.00 | 3.00 |
| 11.00 | 1.00 |
| 10.00 | 2.00 |
| 10.00 | 8.00 |
| 6.00 | 5.00 |

M(1) = 9.29 SD(1) = 2.25
M(2) = 4.71 SD(2) = 2.76
SE = 3.56
Z = 1.28*

Table VII-5c Vocabulary-noise words (all commands)

OBSERVED DATA

| POSITIONAL | ENGLISH |
|---|---|
| 6.00 | 7.00 |
| 3.00 | 8.00 |
| 5.00 | 13.00 |
| 5.00 | 13.00 |
| 8.00 | 11.00 |
| 3.00 | 10.00 |
| 5.00 | 6.00 |

M(1) = 5.00 SD(1) = 1.60
M(2) = 9.71 SD(2) = 2.60
SE = 3.06
Z = -1.54*

Table VII-5d Vocabulary-noise words (all commands)

## *RESEARCH HYPOTHESIS 6: ERRORS*

***Syntactic and semantic errors*** will be uniform over languages.

Assumptions

Keyword is syntactically more complex than the other languages; thus a few more errors might occur there. Positional contains less "guideline" information (in the form of keywords); consequently there may be a few more semantic errors. But for these experienced subjects (i.e., this particular sample) there should be no significance.

Calculations

The F ratio of 0.88 (Table VII-6) supports the research hypothesis.

Conclusions

Regarding this hypothesis, we are as concerned with user training as with language deficiencies. A natural question to ask is whether or not there is a relationship between the occurrence of errors and other variables measuring activity related to training aids. More specifically, are errors related to references-to-materials or advice-from-consultant? See the correlation findings which follow later in this chapter.

We would also like to pinpoint the conceptually more difficult tasks, if any. Figures VII-6a and 6b show syntax and semantic errors plotted against the task steps. Looking at the syntactic error plot, it is evident that syntax errors occur frequently as the first step or two of a task unit. Note furthermore that this seems to occur when the subject takes on a new, unfamiliar role. Note the absence of such errors in task units 3, 4, 5 and 9 where the subject is assuming a role he has played previously, and in each case where errors occur (task units 1, 2, 6, 7, 8) except task unit 10 the subject is cast in a new role. This takes on even more significance when we observe that the first command, for example, in tasks 2, 4, 5, and 6 is the DISPLAY command. This strongly suggests that the syntax errors were just surface phenomena and that they were perhaps caused by incomplete understanding of the role being played. This should in turn suggest to the training personnel more careful attention to describing the mode of the user.

The problem underlying semantic errors (Fig. VII-6b) is more localized. [It is not related to the user's role.] In five instances semantic errors occurred with a frequency greater than one. Two of these operations involved the display of multiple fields of a message and the error resulted from the subject not requesting the proper fields called for by the instructions. In these cases the subject no doubt requested the information he would actually want to see and not that stated in the instructions. The other three instances point out a genuine semantic difficulty. In each of these cases the user was asked to either display or disseminate selected message information from a larger context (of messages or fields of a message). In each case, the message selector specified was not resolved finely enough and the result was the presentation or transmission of much unwanted information. This kind of problem is not serious (for displaying information to oneself) other than the file access and CPU time involved, since

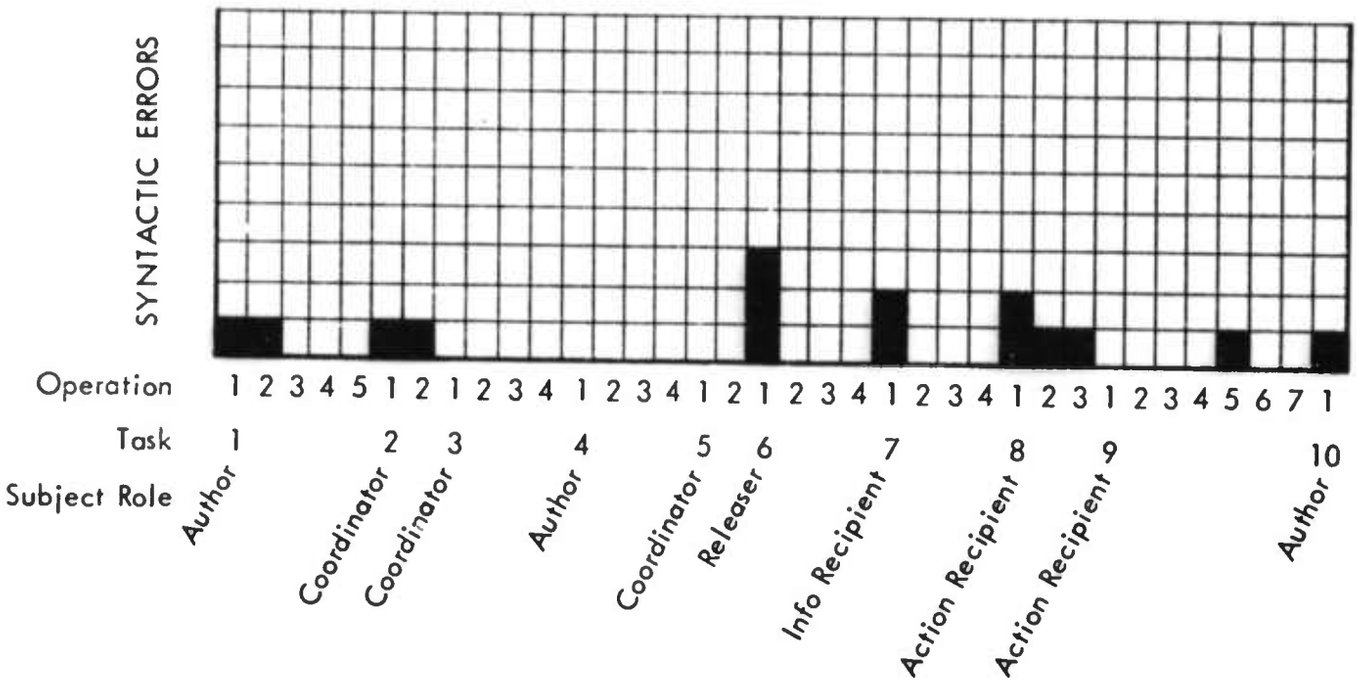
the subject will readily observe the mistake and take corrective action. Where messages are being transmitted, the error is less obvious. Perhaps the service should, in this case, display identifiers of the data to be transmitted and require that the user acknowledge. Such a closed-loop approach should reduce the ill effects of semantic errors.

OBSERVED DATA

| SUBJECTS | | | GROUP | LANGUAGE |
|---|---|---|---|---|
| 2.00 | 4.00 | 3.00 | 1 | 1 |
| 2.00 | 0.00 | 0.00 | 1 | 2 |
| 0.00 | 1.00 | 1.00 | 1 | 3 |
| 0.00 | 2.00 | 0.00 | 2 | 1 |
| 1.00 | 3.00 | 0.00 | 2 | 2 |
| 1.00 | 3.00 | 0.00 | 2 | 3 |
| 13.00 | 3.00 | 1.00 | 3 | 1 |
| 3.00 | 2.00 | 0.00 | 3 | 2 |
| 14.00 | 3.00 | 2.00 | 3 | 3 |

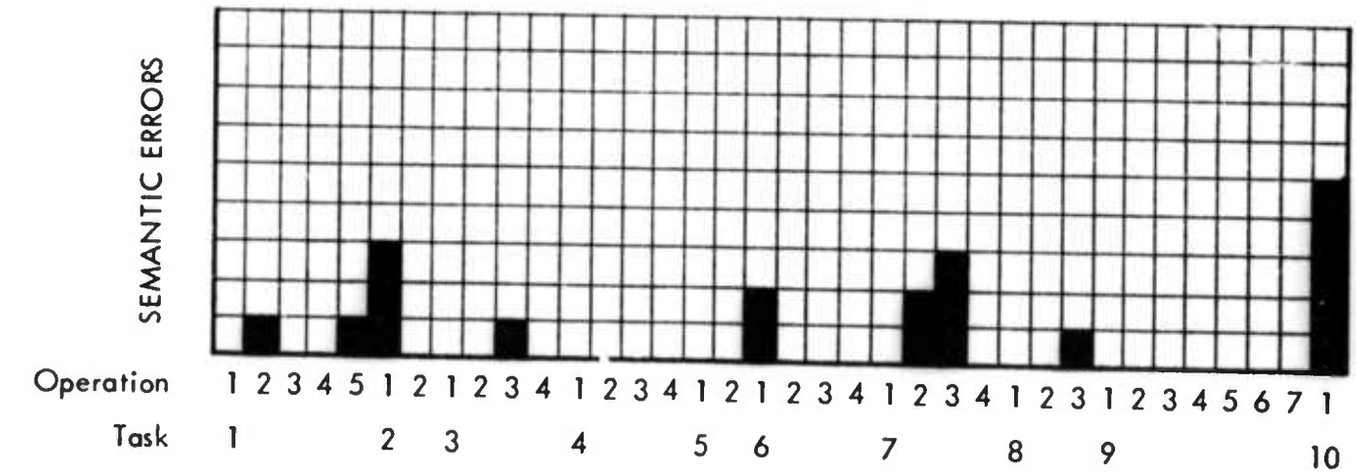
| SOURCE OF VARIATION | SS | DF | MS | F |
|---|---|---|---|---|
| GROUPS | 64.96 | 2 | 32.48 | 3.02* |
| LANGUAGES | 18.30 | 2 | 9.15 | 0.85 |
| ORDER | 1.85 | 2 | 0.93 | 0.09 |
| RESIDUAL | 29.85 | 2 | 14.93 | 1.39 |
| WITHIN CELL | 193.33 | 18 | 10.74 | 1.00 |

Table VII-6 Errors: syntax and semantics -- over all commands




NOTE: The one "outlier" subject has been omitted from this graph to give a more accurate picture of the group.

Figure VII-6a Syntactic errors per operation




NOTE: The "outlier" subject has been omitted from this graph to give a more accurate picture of the group.

Figure VII-6b Semantic errors per operation

## *RESEARCH HYPOTHESIS 7: OMISSIONS AND PROMPTS*

***Keyword omission and service prompts*** will be significant in keyword and English-like over positional.

Assumption

The common experience of these subjects in using a positional language will result in keyword omission simply as a motor response. Positional will not be significant, since keywords appear there only where qualification is necessary.

Calculations

The F ratio of 4.31 in Table VII-7a is significant at the 0.95 level. There is also a rather high (not significant) value of 2.37 for the order effect. The Z-values in Tables VII-7a, b, and c in part support the research hypothesis, i.e., English-like is significant at the 0.90 level but keyword is not.

Conclusions

The unexpected order effect (an artificial variable) can be explained by noting the 60 service prompts issued to one subject in column 1 of Table VII-7a. This particular individual refused to enter any keywords in the keyword language. Clearly, keyword is not an effective language for this subject. With respect to language adjustments, however, upon examination of the task input commands the language designer would discover that the preponderance of keyword omissions by the other eight subjects occurred in two situations. Keywords were frequently dropped with single-parameter commands and also they were often dropped from the first parameter (the subject) of multi-parameter commands, as in natural language.

Perhaps the English-like language should be based on positional recognition rather than keyword recognition as in the proffered English-like language.

OBSERVED DATA

| SUBJECTS | | | GROUP | LANGUAGE |
|---|---|---|---|---|
| 20.00 | 11.00 | 6.00 | 1 | 1 |
| 5.00 | 4.00 | 2.00 | 1 | 2 |
| 24.00 | 19.00 | 13.00 | 1 | 3 |
| 2.00 | 6.00 | 27.00 | 2 | 1 |
| 3.00 | 2.00 | 8.00 | 2 | 2 |
| 10.00 | 9.00 | 21.00 | 2 | 3 |
| 60.00 | 8.00 | 34.00 | 3 | 1 |
| 7.00 | 1.00 | 11.00 | 3 | 2 |
| 13.00 | 11.00 | 8.00 | 3 | 3 |

| SOURCE OF VARIATION | SS | DF | MS | F |
|---|---|---|---|---|
| GROUPS | 254.89 | 2 | 127.44 | 1.12 |
| LANGUAGES | 981.56 | 2 | 490.78 | 4.31** |
| ORDER | 540.67 | 2 | 270.33 | 2.37 |
| RESIDUAL | 284.22 | 2 | 142.11 | 1.25 |
| WITHIN CELL | 2051.33 | 18 | 113.96 | 1.00 |

Table VII-7a Keyword omissions and prompts -- over all commands

OBSERVED DATA

| ENGLISH | KEYWORD |
|---|---|
| 24.00 | 20.00 |
| 19.00 | 11.00 |
| 13.00 | 6.00 |
| 9.00 | 6.00 |
| 21.00 | 27.00 |
| 11.00 | 8.00 |
| 8.00 | 34.00 |

M(1) = 15.00 SD(1) = 5.83
M(2) = 16.00 SD(2) = 10.35
SE = 11.88
Z = -0.08

Table VII-7b Keyword omissions & prompts (all commands)

OBSERVED DATA

| POSITIONAL | ENGLISH |
|---|---|
| 5.00 | 24.00 |
| 4.00 | 19.00 |
| 2.00 | 13.00 |
| 3.00 | 10.00 |
| 2.00 | 9.00 |
| 8.00 | 21.00 |
| 7.00 | 13.00 |

M(1) = 4.43 SD(1) = 2.19
M(2) = 15.57 SD(2) = 5.34
SE = 5.77
Z = -1.93**

Table VII-7c Keyword omissions & prompts (all commands)

OBSERVED DATA

| KEYWORD | POSITIONAL |
|---|---|
| 20.00 | 5.00 |
| 11.00 | 4.00 |
| 6.00 | 2.00 |
| 6.00 | 2.00 |
| 27.00 | 8.00 |
| 8.00 | 1.00 |
| 34.00 | 11.00 |

M(1) = 16.00 SD(1) = 10.35
M(2) = 4.71 SD(2) = 3.37
SE = 10.89
Z = 1.04

Table VII-7d Keyword omissions & prompts (all commands)

## *RESEARCH HYPOTHESIS 8: REORDERING*

***Reordering*** the sequence of parameters will be significant in the positional language contrasted with each of keyword and English-like.

Assumption

In the "strawman" languages an attempt was made to make the English-like language "sound right" with respect to the order of parameters. Keyword and positional were modeled from the English-like. Since special or keywords appear most frequently in keyword and English-like, their somewhat natural order will be left intact. In positional the information-providing keywords do not appear; thus we would expect positional to be rearranged to allow easy contextual defaulting, since no other natural order prevails.

Calculations

Table VII-8a shows an F-ratio of 0.54 which means that we must reject the research hypothesis. Tables VII-8b through 8g show analysis of variance and comparison of means for the more significant commands at a command level.

Conclusions

Many single-parameter commands contribute to the lack of significant variation over all commands. Examined on a command level, several commands are, however, significant. An overall observation from examining the task transcripts shows that the most frequently rearranged parameter was the message selector. In each case the user had established a contextual default for the message identification and was moving it to the right end to omit it.

Tables VII-8b, c, and d show values for reordering the folder definition command. In this case the parameters were not reordered for the purpose of defaulting, since both parameters must be given unless preprogrammed. This is a highly significant change in the positional language. In the NOTE command, the SHOW and WHEN parameters were interchanged more frequently in English-like. Table VII-8e gives a value of 1.88 although the comparison of means were not significant. MESSAGE and TO were often interchanged in the SELL command (see Table VII-8f). Again, the English-like was changed more often, although the Z-values were not significant. In the ASSIGN command (Table VII-8g) CAPABILITIES and MESSAGE were interchanged more often in positional.

OBSERVED DATA

| SUBJECTS | | | GROUP | LANGUAGE |
|---|---|---|---|---|
| 12.00 | 15.00 | 17.00 | 1 | 1 |
| 11.00 | 15.00 | 13.00 | 1 | 2 |
| 14.00 | 16.00 | 11.00 | 1 | 3 |
| 6.00 | 13.00 | 9.00 | 2 | 1 |
| 19.00 | 18.00 | 8.00 | 2 | 2 |
| 14.00 | 11.00 | 9.00 | 2 | 3 |
| 3.00 | 16.00 | 13.00 | 3 | 1 |
| 8.00 | 12.00 | 17.00 | 3 | 2 |
| 15.00 | 16.00 | 11.00 | 3 | 3 |

| SOURCE OF VARIATION | SS | DF | MS | F |
|---|---|---|---|---|
| GROUPS | 17.56 | 2 | 8.78 | 0.54 |
| LANGUAGES | 17.56 | 2 | 8.78 | 0.54 |
| ORDER | 4.67 | 2 | 2.33 | 0.14 |
| RESIDUAL | 48.22 | 2 | 24.11 | 1.49 |
| WITHIN CELL | 292.00 | 18 | 16.22 | 1.00 |

Table VII-8a Reordering of parameters -- over all commands

OBSERVED DATA

| SUBJECTS | | | GROUP | LANGUAGE |
|---|---|---|---|---|
| 0.00 | 0.00 | 0.00 | 1 | 1 |
| 1.00 | 0.00 | 1.00 | 1 | 2 |
| 0.00 | 0.00 | 0.00 | 1 | 3 |
| 0.00 | 0.00 | 0.00 | 2 | 1 |
| 1.00 | 1.00 | 1.00 | 2 | 2 |
| 0.00 | 0.00 | 0.00 | 2 | 3 |
| 0.00 | 0.00 | 0.00 | 3 | 1 |
| 1.00 | 1.00 | 2.00 | 3 | 2 |
| 0.00 | 0.00 | 0.00 | 3 | 3 |

| SOURCE OF VARIATION | SS | OF | MS | F |
|---|---|---|---|---|
| GROUPS | 0.22 | 2 | 0.11 | 1.50 |
| LANGUAGES | 6.00 | 2 | 3.00 | 40.50**** |
| ORDER | 0.22 | 2 | 0.11 | 1.50 |
| RESIDUAL | 0.22 | 2 | 0.11 | 1.50 |
| WITHIN CELL | 1.33 | 18 | 0.07 | 1.00 |

Table VII-8b Reordering -- create folder

OBSERVED DATA

| KEYWORD | POSITIONAL |
|---|---|
| 0.00 | 1.00 |
| 0.00 | 0.00 |
| 0.00 | 1.00 |
| 0.00 | 1.00 |
| 0.00 | 1.00 |
| 0.00 | 1.00 |
| 0.00 | 1.00 |
| 0.00 | 1.00 |
| 0.00 | 2.00 |

M(1) = 0.00 SD(1) = 0.00
M(2) = 1.00 SD(2) = 0.47
SE = 0.47
Z = -2.12**

Table VII-8c Reordering -- create folder

OBSERVED DATA

| POSITIONAL | ENGLISH-LIKE |
|---|---|
| 1.00 | 0.00 |
| 0.00 | 0.00 |
| 1.00 | 0.00 |
| 1.00 | 0.00 |
| 1.00 | 0.00 |
| 1.00 | 0.00 |
| 1.00 | 0.00 |
| 1.00 | 0.00 |
| 2.00 | 0.00 |

M(1) = 1.00 SD(1) = 0.47
M(2) = 0.00 SD(2) = 0.00
SE = 0.47
Z = 2.12**

Table VII-8d Reordering -- create folder

OBSERVED DATA

| SUBJECTS | | | GROUP | LANGUAGE |
|---|---|---|---|---|
| 0.00 | 3.00 | 3.00 | 1 | 1 |
| 1.00 | 1.00 | 2.00 | 1 | 2 |
| 2.00 | 5.00 | 2.00 | 1 | 3 |
| 2.00 | 0.00 | 1.00 | 2 | 1 |
| 3.00 | 4.00 | 1.00 | 2 | 2 |
| 4.00 | 1.00 | 3.00 | 2 | 3 |
| 1.00 | 4.00 | 0.00 | 3 | 1 |
| 1.00 | 0.00 | 2.00 | 3 | 2 |
| 4.00 | 3.00 | 1.00 | 3 | 3 |

| SOURCE OF VARIATION | SS | OF | MS | F |
|---|---|---|---|---|
| GROUPS | 0.67 | 2 | 0.33 | 0.15 |
| LANGUAGES | 8.22 | 2 | 4.11 | 1.88 |
| ORDER | 2.89 | 2 | 1.44 | 0.66 |
| RESIDUAL | 2.89 | 2 | 1.44 | 0.66 |
| WITHIN CELL | 39.33 | 18 | 2.19 | 1.00 |

Table VII-8e Reordering -- note

OBSERVED DATA

| SUBJECTS | | | GROUP | LANGUAGE |
|---|---|---|---|---|
| 1.00 | 2.00 | 3.00 | 1 | 1 |
| 0.00 | 3.00 | 2.00 | 1 | 2 |
| 1.00 | 2.00 | 3.00 | 1 | 3 |
| 1.00 | 3.00 | 1.00 | 2 | 1 |
| 2.00 | 2.00 | 0.00 | 2 | 2 |
| 3.00 | 3.00 | 1.00 | 2 | 3 |
| 0.00 | 3.00 | 3.00 | 3 | 1 |
| 0.00 | 0.00 | 2.00 | 3 | 2 |
| 3.00 | 3.00 | 1.00 | 3 | 3 |

| SOURCE OF VARIATION | SS | DF | MS | F |
|---|---|---|---|---|
| GROUPS | 0.22 | 2 | 0.11 | 0.07 |
| LANGUAGES | 4.67 | 2 | 2.33 | 1.50 |
| ORDER | 0.22 | 2 | 0.11 | 0.07 |
| RESIDUAL | 1.56 | 2 | 0.78 | 0.50 |
| WITHIN CELL | 28.00 | 18 | 1.56 | 1.00 |

Table VII-8f Reordering -- sell

OBSERVED DATA

| SUBJECTS | | | GROUP | LANGUAGE |
|---|---|---|---|---|
| 1.00 | 2.00 | 3.00 | 1 | 1 |
| 2.00 | 2.00 | 1.00 | 1 | 2 |
| 1.00 | 0.00 | 0.00 | 1 | 3 |
| 1.00 | 2.00 | 3.00 | 2 | 1 |
| 1.00 | 2.00 | 3.00 | 2 | 2 |
| 0.00 | 1.00 | 3.00 | 2 | 3 |
| 0.00 | 1.00 | 0.00 | 3 | 1 |
| 1.00 | 1.00 | 2.00 | 3 | 2 |
| 1.00 | 1.00 | 1.00 | 3 | 3 |

| SOURCE OF VARIATION | SS | DF | MS | F |
|---|---|---|---|---|
| GROUPS | 3.56 | 2 | 1.78 | 2.40 |
| LANGUAGES | 2.89 | 2 | 1.44 | 1.95 |
| ORDER | 2.67 | 2 | 1.33 | 1.80 |
| RESIDUAL | 1.56 | 2 | 0.78 | 1.05 |
| WITHIN CELL | 13.33 | 18 | 0.74 | 1.00 |

Table VII-8g Reordering -- assign

## *RESEARCH HYPOTHESIS 9: ABBREVIATIONS*

***Abbreviations of language elements*** will occur more frequently in positional.

**Assumption**

The subjects will abbreviate more in the abstract, functional-like languages and less so in the natural-like language.

**Calculations**

Table VII-9 shows no differences among languages with respect to abbreviations.

**Conclusions**

The research hypothesis must be rejected. However, the observed data in Table VII-9, strongly indicates that abbreviations are important (and equally so) in each language. When we examine the task input commands, we find that three types of abbreviations are necessary. The first is a substring of the word that is long enough to be unique but not necessarily the shortest unique substring. Example: DI or DISP for DISPLAY. The second requisite is the dictionary's abbreviation, such as MSG for MESSAGE as opposed to MES. The third and most difficult type to anticipate or uncover are first-letter abbreviations or acronyms such as DTG for date-time group. Since offering these varieties is not difficult nor time-consuming in CPU, they should be a feature of each language. The language designer should converse further with the target population to identify common acronyms.

Two other observations are in order. One might raise serious objections with respect to the way this variable was treated, since inputs were vocal rather than typed. Past experience indicates that more abbreviations would take place in typing. The second observation concerns a naturally suspected difference in populations. The target population for the prototype message service may find it more natural to spell out words in English-like. Since the feature is inexpensive, our concern is not an economic one but centers around an issue of user training. Considering both of these observations, an automated version of the protocol analysis (where the subject types commands) is suggested for a sample from the target population.

OBSERVED DATA

| SUBJECTS | | | GROUP | LANGUAGE |
|---|---|---|---|---|
| 0.00 | 24.00 | 49.00 | 1 | 1 |
| 0.00 | 37.00 | 33.00 | 1 | 2 |
| 2.00 | 41.00 | 27.00 | 1 | 3 |
| 12.00 | 5.00 | 0.00 | 2 | 1 |
| 7.00 | 3.00 | 0.00 | 2 | 2 |
| 5.00 | 2.00 | 0.00 | 2 | 3 |
| 6.00 | 53.00 | 3.00 | 3 | 1 |
| 1.00 | 43.00 | 2.00 | 3 | 2 |
| 1.00 | 32.00 | 3.00 | 3 | 3 |

| SOURCE OF VARIATION | SS | DF | MS | F |
|---|---|---|---|---|
| GROUPS | 1811.19 | 2 | 905.59 | 2.61 |
| LANGUAGES | 87.63 | 2 | 43.81 | 0.13 |
| ORDER | 23.19 | 2 | 11.59 | 0.03 |
| RESIDUAL | 23.41 | 2 | 11.70 | 0.03 |
| WITHIN CELL | 6239.33 | 18 | 346.63 | 1.00 |

Table VII-9 Abbreviations -- over all commands

## *RESEARCH HYPOTHESIS 10: REFERENCES*

***References*** to the command manuals will be uniform across languages.

Assumptions

References will decrease (due to learning effect) with each pass through the task, i.e., with each language application. The Latin square design will insure uniformity over languages. The command names are essentially the same for each language form. Perhaps there will be a slight increase in the use of the manuals to look up keywords while using the keyword language, but it shouldn't be significant. In English-like, the subject will tend to "invent" needed parameters. In positional, fewer instances of keywords appear.

Calculations

Table VII-10 shows that the differences in languages are not significant at the 0.90 level. Note the high unexplained error variation of 6.77. The Z-values in Tables VII-10b, c, and d support the research hypothesis.

Conclusions

Although the research hypothesis is tenable according to the Z-values and the F-ratio for languages, there should be concern for the high variation among groups (3.01) and the large residual. This suggests that we should look at the data in a different way that will provide more insight into what took place. Figure VII-10a shows references ***versus*** the order of using languages. The left panel is ideally what we would hope for, and expect. It clearly shows the learning effect over time. The center panel shows relatively the same thing. Subjects A and B follow the anticipated trend; the scaling is probably due to the fact that these particular individuals have had a broader exposure to on-line systems and to language design than the others. The slight upswing by C in the keyword language is insignificant and was accounted for in the assumptions above. In the rightmost panel subject D obeys our prediction. E and F present somewhat of a problem. They made fewer references in English-like, their first language, than in keyword and positional. Yet, these two individuals were staunch supporters of positional as done in the computing system at ISI. The reader should compare these results with those of hypotheses 6 and 11.

One observation is that where a stated preference by the subject and his observed performance are incongruous, then some process such as that proposed in [HEAFNER 74] should be applied to aid in selecting the appropriate language for that subject. Language selection, of course, should be based on a weighted function of many variables.

Another observation is that a glance at Fig. VII-10a clearly demonstrates the need for a much larger sample size for studies such as this. The results in Fig. VII-10a are definitely not due to treatment effects. They reflect an honest attitude of these particular subjects about these forms of language. Further appraisal of this variable is recommended because of the extreme residual variation and the consistently high

standard errors of the differences between means. The standard error is inversely proportional to sample size, thus the need for a larger sample. Of interest in Fig. VII-10b are those operations which were extraordinarily difficult. We suggest further examination of the top five percent (i.e., $2\sigma$) of those most referenced.

OBSERVED DATA

| SUBJECTS | | | GROUP | LANGUAGE |
|---|---|---|---|---|
| 16.00 | 15.00 | 18.00 | 1 | 1 |
| 11.00 | 10.00 | 5.00 | 1 | 2 |
| 1.00 | 2.00 | 0.00 | 1 | 3 |
| 0.00 | 4.00 | 0.00 | 2 | 1 |
| 3.00 | 17.00 | 6.00 | 2 | 2 |
| 1.00 | 3.00 | 4.00 | 2 | 3 |
| 16.00 | 7.00 | 8.00 | 3 | 1 |
| 15.00 | 7.00 | 1.00 | 3 | 2 |
| 11.00 | 2.00 | 18.00 | 3 | 3 |

| SOURCE OF VARIATION | SS | DF | MS | F |
|---|---|---|---|---|
| GROUPS | 142.89 | 2 | 71.44 | 3.01* |
| LANGUAGES | 108.67 | 2 | 54.33 | 2.29 |
| ORDER | 28.22 | 2 | 14.11 | 0.59 |
| RESIDUAL | 321.56 | 2 | 160.78 | 6.77 |
| WITHIN CELL | 427.33 | 18 | 23.74 | 1.00 |

Table VII-10a References to materials -- over all commands

OBSERVED DATA

| KEYWORD | POSITIONAL |
|---|---|
| 16.00 | 11.00 |
| 15.00 | 10.00 |
| 18.00 | 5.00 |
| 4.00 | 17.00 |
| 0.00 | 6.00 |
| 7.00 | 7.00 |
| 8.00 | 1.00 |

M(1) = 9.71 SD(1) = 6.25
M(2) = 8.14 SD(2) = 4.73
SE = 7.84
Z = 0.20

Table VII-10b Reference to materials (all commands)

OBSERVED DATA

| POSITIONAL | ENGLISH |
|---|---|
| 11.00 | 1.00 |
| 10.00 | 2.00 |
| 3.00 | 1.00 |
| 6.00 | 4.00 |
| 15.00 | 11.00 |
| 7.00 | 2.00 |
| 1.00 | 18.00 |

M(1) = 7.57 SD(1) = 4.47
M(2) = 5.57 SD(2) = 6.02
SE = 7.50
Z = 0.27

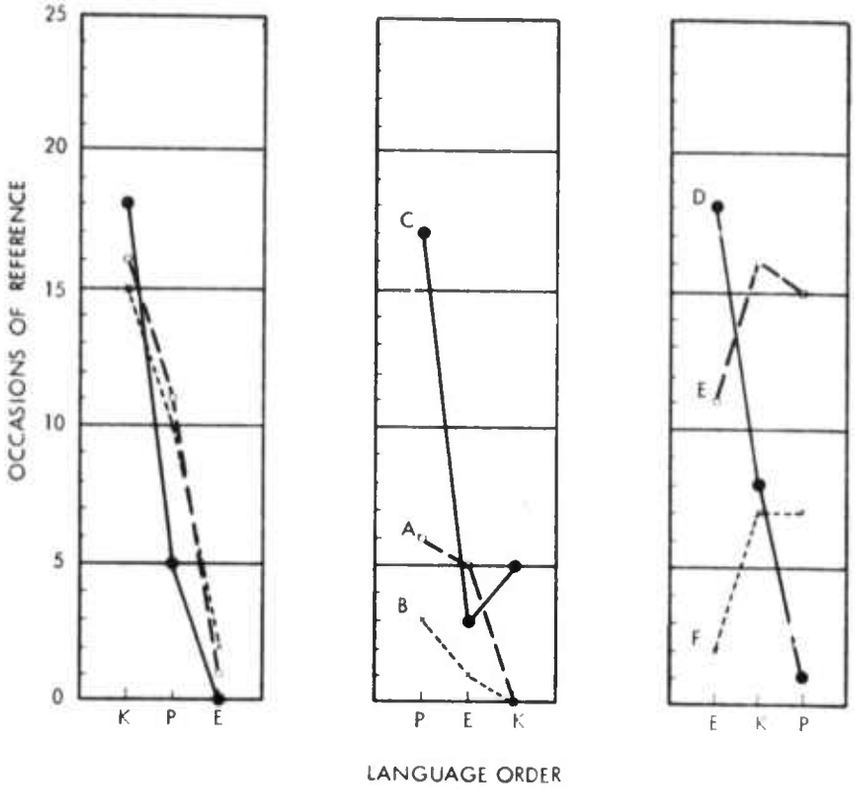
Table VII-10c Reference to materials (all commands)

OBSERVED DATA

| ENGLISH | KEYWORD |
|---|---|
| 1.00 | 16.00 |
| 2.00 | 15.00 |
| 0.00 | 18.00 |
| 3.00 | 4.00 |
| 4.00 | 0.00 |
| 11.00 | 16.00 |
| 2.00 | 7.00 |

M(1) = 3.29 SD(1) = 3.37
M(2) = 10.86 SD(2) = 6.56
SE = 7.37
Z = -1.03

Table VII-10d Reference to materials (all commands)




Legend: K = keyword
P = positional
E = English-like

Figure VII-10a References as a function of order

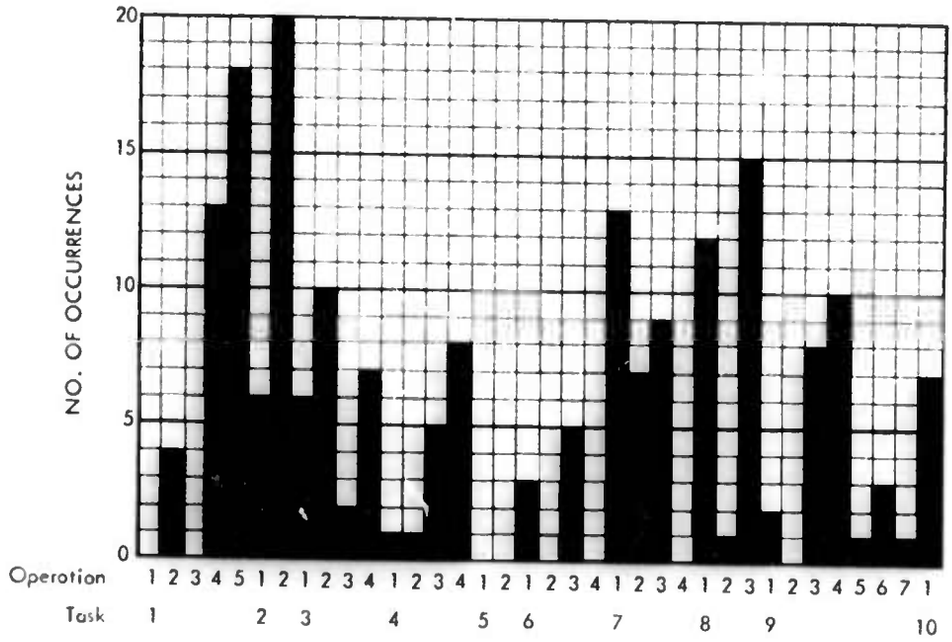



Figure VII-10b References per operation

## *RESEARCH HYPOTHESIS 11: ADVICE*

***Advice from the consultant*** will be uniform across languages.

Assumption

Learning will result in a trend effect, but the Latin square design will spread this uniformly across languages. There is no reason to believe that advice will be language-dependent for this experienced group.

Calculations

The F-ratio of 0.48 supports the research hypothesis; see Table VII-11. The variation among groups can be more clearly seen in Fig. VII-11a.

Conclusions

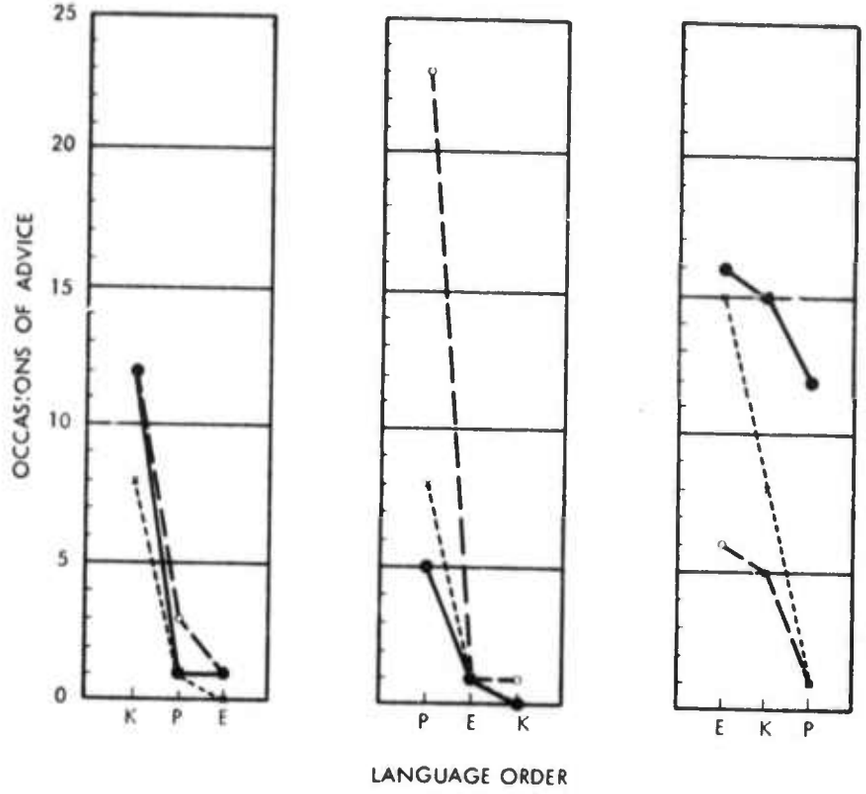
Interest lies in determining which kinds of operations are the more complex. Figure VII-11b (also compare to Fig. VII-10b) plots the occurrences of advice requests against task operations. We consider a reasonable threshold to be $2\sigma$, i.e., those operations in the top 5 percent of highest requests are assumed to be complex enough to warrant further inquiry.

OBSERVED DATA

| SUBJECTS | | | GROUP | LANGUAGE |
|---|---|---|---|---|
| 12.00 | 12.00 | 8.00 | 1 | 1 |
| 1.00 | 3.00 | 1.00 | 1 | 2 |
| 1.00 | 1.00 | 0.00 | 1 | 3 |
| 0.00 | 1.00 | 0.00 | 2 | 1 |
| 5.00 | 23.00 | 8.00 | 2 | 2 |
| 1.00 | 1.00 | 1.00 | 2 | 3 |
| 15.00 | 5.00 | 8.00 | 3 | 1 |
| 12.00 | 1.00 | 1.00 | 3 | 2 |
| 16.00 | 6.00 | 15.00 | 3 | 3 |

| SOURCE OF VARIATION | SS | DF | MS | F |
|---|---|---|---|---|
| GROUPS | 115.63 | 2 | 57.81 | 2.64* |
| LANGUAGES | 20.96 | 2 | 10.48 | 0.48 |
| ORDER | 31.63 | 2 | 15.81 | 0.72 |
| RESIDUAL | 476.52 | 2 | 238.26 | 10.87*** |
| WITHIN CELL | 394.67 | 18 | 21.93 | 1.00 |

Table VII-11 Advice from consultant -- over all commands




Legend: K = keyword
P = positional
E = English-like

Figure VII-11a Advice as a function of order

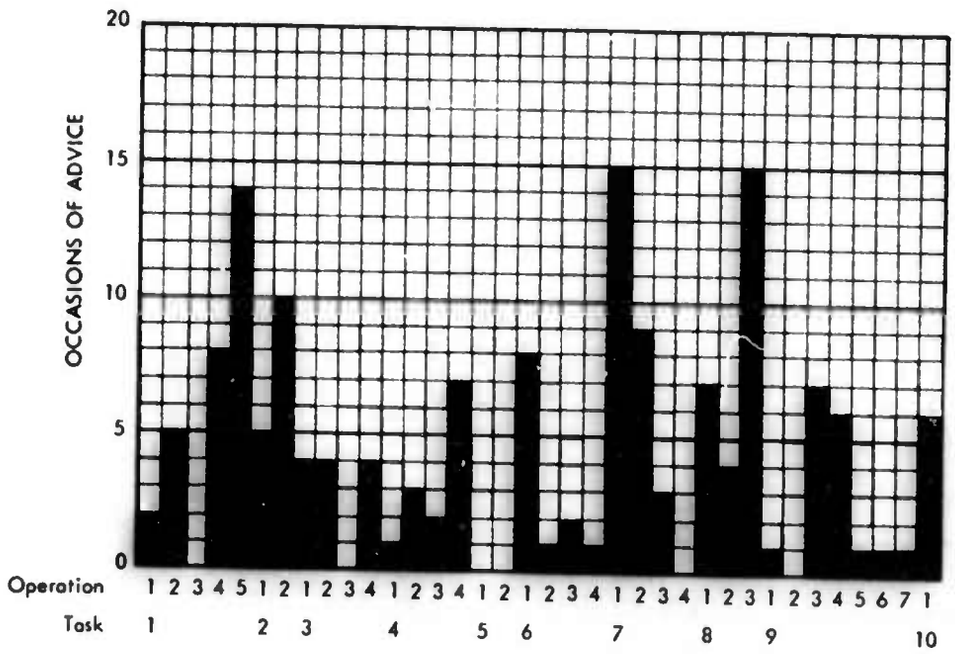



Figure VII-11b Advice per operation

## *RESEARCH HYPOTHESIS 12: CONTEXTUAL DEFAULTS*

***Contextual defaults*** will not be significant over languages.

Assumption
No reason to suspect language dependencies.

Calculations
Table VII-12 shows that the research hypothesis is supported (F = 0.41).

Conclusions
Figure VII-12 shows that many contextual defaults were used. It is suggested that the designer construct a frequency distribution of nonterminals defaulted from the commands in the upper 16 percent (1$\sigma$ level) of frequency. By far the most commonly defaulted nonterminal was the message selector.

OBSERVED DATA

| SUBJECTS | | | GROUP | LANGUAGE |
|---|---|---|---|---|
| 16.00 | 23.00 | 14.00 | 1 | 1 |
| 12.00 | 27.00 | 13.00 | 1 | 2 |
| 7.00 | 27.00 | 24.00 | 1 | 3 |
| 17.00 | 28.00 | 14.00 | 2 | 1 |
| 30.00 | 27.00 | 10.00 | 2 | 2 |
| 24.00 | 29.00 | 15.00 | 2 | 3 |
| 13.00 | 21.00 | 24.00 | 3 | 1 |
| 20.00 | 21.00 | 28.00 | 3 | 2 |
| 27.00 | 24.00 | 21.00 | 3 | 3 |

| SOURCE OF VARIATION | SS | DF | MS | F |
|---|---|---|---|---|
| GROUPS | 84.52 | 2 | 42.26 | 0.78 |
| LANGUAGES | 44.74 | 2 | 22.37 | 0.41 |
| ORDER | 3.63 | 2 | 1.81 | 0.03 |
| RESIDUAL | 10.96 | 2 | 5.48 | 0.10 |
| WITHIN CELL | 980.67 | 18 | 54.48 | 1.00 |

Table VII-12 Contextual defaults -- over all commands

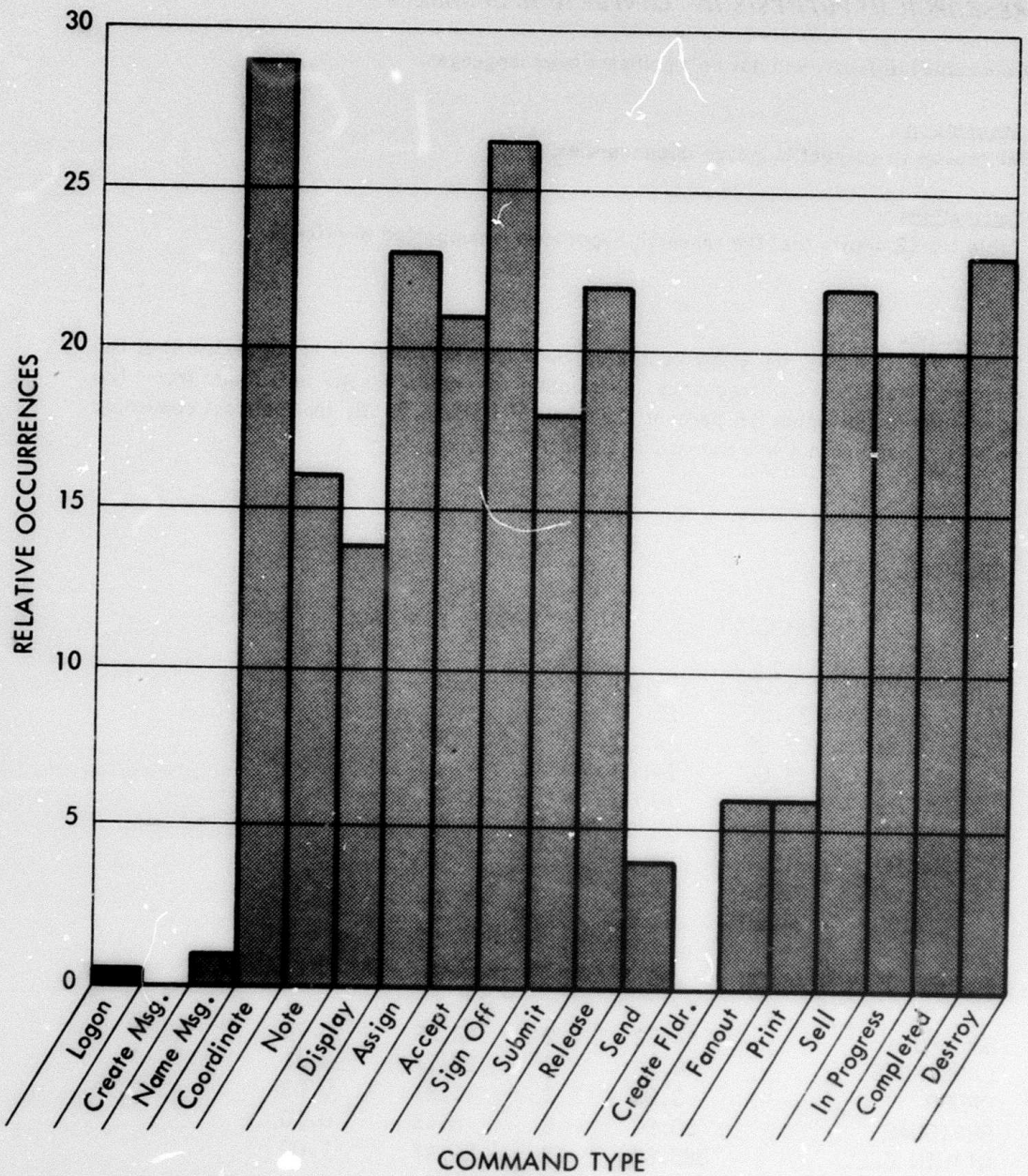



Figure VII-12 Contextual defaults per command type

## *RESEARCH HYPOTHESIS 13: PROGRAMMABLE DEFAULTS*

***Programmable defaults*** will not be significant over languages.

Assumption
No reason to suspect language dependencies.

Calculations
Table VII-13 shows that the research hypothesis is supported (F = 1.04).

Conclusions
Figure VII-13 shows a large disparity among the operations with respect to programmed defaults. It is suggested that the service function planner examine those parameters defaulted in the operations in the upper 16 percent (1$\sigma$ level) in order to determine which parameters should be settable (programmable) by the user.

OBSERVED DATA

| SUBJECTS | | | GROUP | LANGUAGE |
|---|---|---|---|---|
| 5.00 | 17.00 | 9.00 | 1 | 1 |
| 3.00 | 11.00 | 13.00 | 1 | 2 |
| 4.00 | 17.00 | 21.00 | 1 | 3 |
| 30.00 | 12.00 | 20.00 | 2 | 1 |
| 15.00 | 8.00 | 15.00 | 2 | 2 |
| 23.00 | 12.00 | 23.00 | 2 | 3 |
| 10.00 | 10.00 | 16.00 | 3 | 1 |
| 12.00 | 12.00 | 17.00 | 3 | 2 |
| 14.00 | 14.00 | 13.00 | 3 | 3 |

| SOURCE OF VARIATION | SS | OF | MS | F |
|---|---|---|---|---|
| GROUPS | 195.85 | 2 | 97.93 | 2.91* |
| LANGUAGES | 70.30 | 2 | 35.15 | 1.04 |
| ORDER | 14.52 | 2 | 7.26 | 0.22 |
| RESIDUAL | 71.19 | 2 | 35.59 | 1.06 |
| WITHIN CELL | 606.00 | 18 | 33.67 | 1.00 |

Table VII-13 Programmable defaults -- over all commands

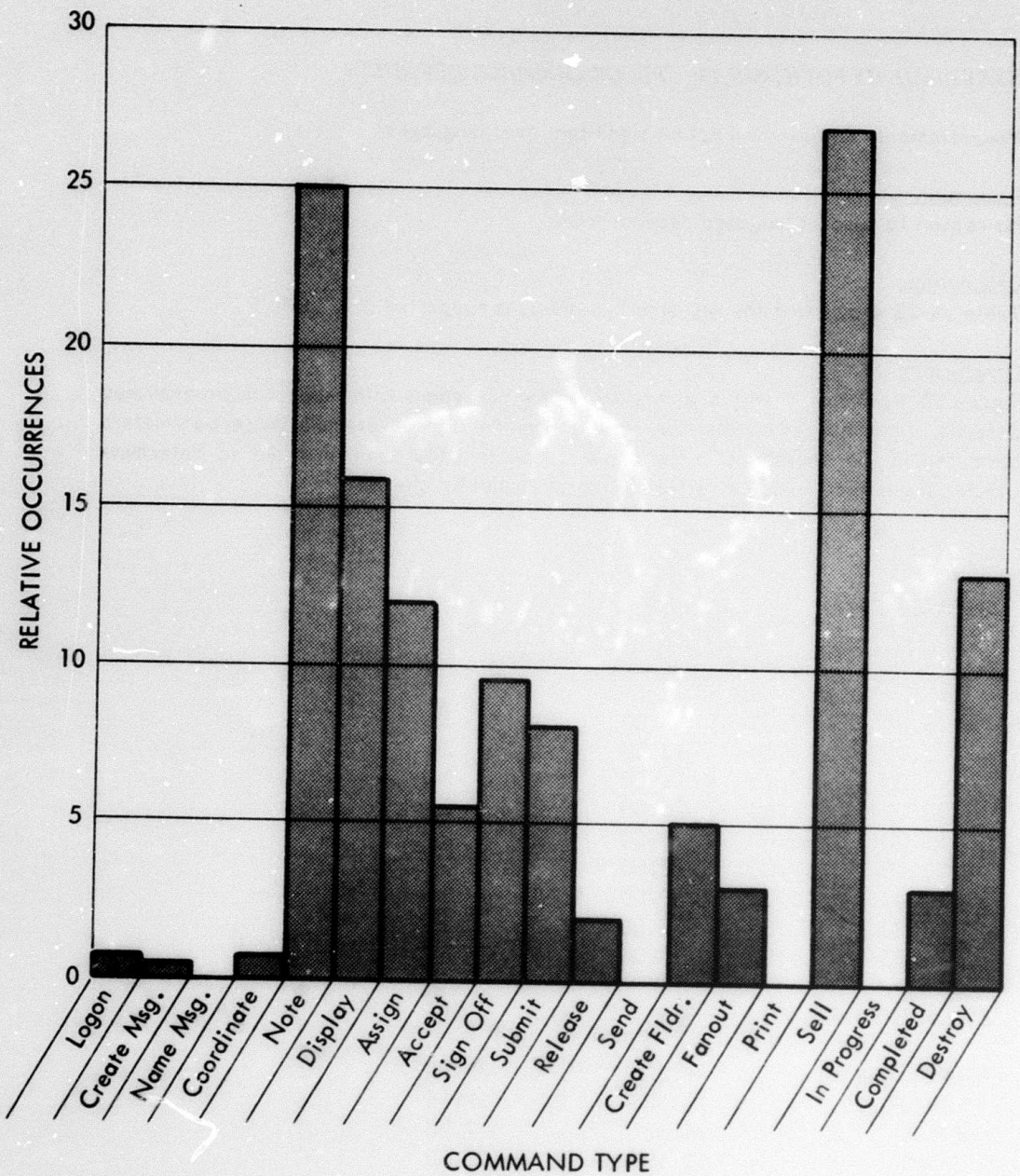



Figure VII-13 Programmable defaults per command type

## *RESEARCH HYPOTHESIS 14: COMPOSITIONS*

There will be no significant difference in the average number of ***command compositions*** with respect to languages.

Assumptions

We would anticipate some composition, but there is no reason to believe that it will be language-dependent.

Calculations

Table VII-14 supports the research hypothesis (which is the Null hypothesis) with an F-value of 0.12. Figures VII-14a, b, c show some indication of language dependency at the command level.

Conclusions

There is no significant difference among languages over all commands. The notion of composition, however, is indigenous to specific, short, logically related sequences of commands rather than to the language as a whole. If there are natural composites, then they should be included in the basic language and available to every subject, even though a repeated sequence detector [HEAFNER 74] would soon discover them on an individual basis for each user. If we look at composition at the level of task operation sequences (Fig. VII-14a, b, and c) (refer also to Appendix A or B), and also break it down according to languages, the bar charts give us two interesting kinds of information. Three two-command sequences were suggested to the subjects as potential composites. In two of the three instances it was apparently natural, since the subjects did indeed carry out the operations by issuing a single command. In the third instance the sequence was combined less than half of the time, which suggests that perhaps it was a poor candidate for composition. The other information of interest, and a more striking observation, is that the sequences task 9 operations 6 and 7, and task 3 operations 1 and 2 were combined one-third of the time in positional and English-like, but not so in keyword. This suggests that the language designer should consider these sequences in the two languages and attempt to infer some general principle concerning the kinds of service functions for which similar operation sequences might arise. These then should be composites in positional and English-like. And, lastly, the sequence task 9 operations 1 and 2 should be examined in the same way and applied to all three languages.

OBSERVED DATA

| SUBJECTS | | | GROUP | LANGUAGE |
|---|---|---|---|---|
| 10.00 | 10.00 | 8.00 | 1 | 1 |
| 4.00 | 10.00 | 12.00 | 1 | 2 |
| 4.00 | 10.00 | 16.00 | 1 | 3 |
| 6.00 | 6.00 | 15.00 | 2 | 1 |
| 7.00 | 6.00 | 20.00 | 2 | 2 |
| 7.00 | 6.00 | 15.00 | 2 | 3 |
| 2.00 | 8.00 | 6.00 | 3 | 1 |
| 2.00 | 12.00 | 8.00 | 3 | 2 |
| 2.00 | 8.00 | 6.00 | 3 | 3 |

| SOURCE OF VARIATION | SS | DF | MS | F |
|---|---|---|---|---|
| GROUPS | 76.74 | 2 | 38.37 | 1.6 |
| LANGUAGES | 5.85 | 2 | 2.93 | 0.1 |
| ORDER | 6.74 | 2 | 3.37 | 0.14 |
| RESIDUAL | 4.36 | 2 | 2.48 | 0.11 |
| WITHIN CELL | 422.00 | 18 | 23.44 | 1.00 |

Table VII-14 Composition of commands -- over all commands

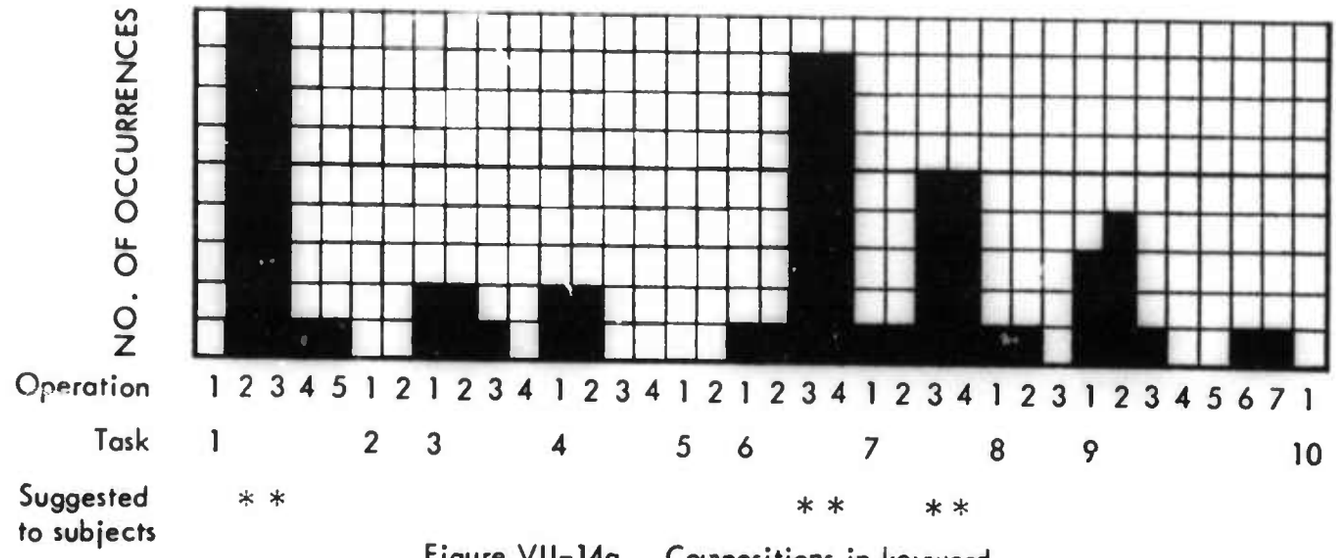



Figure VII-14a Compositions in keyword

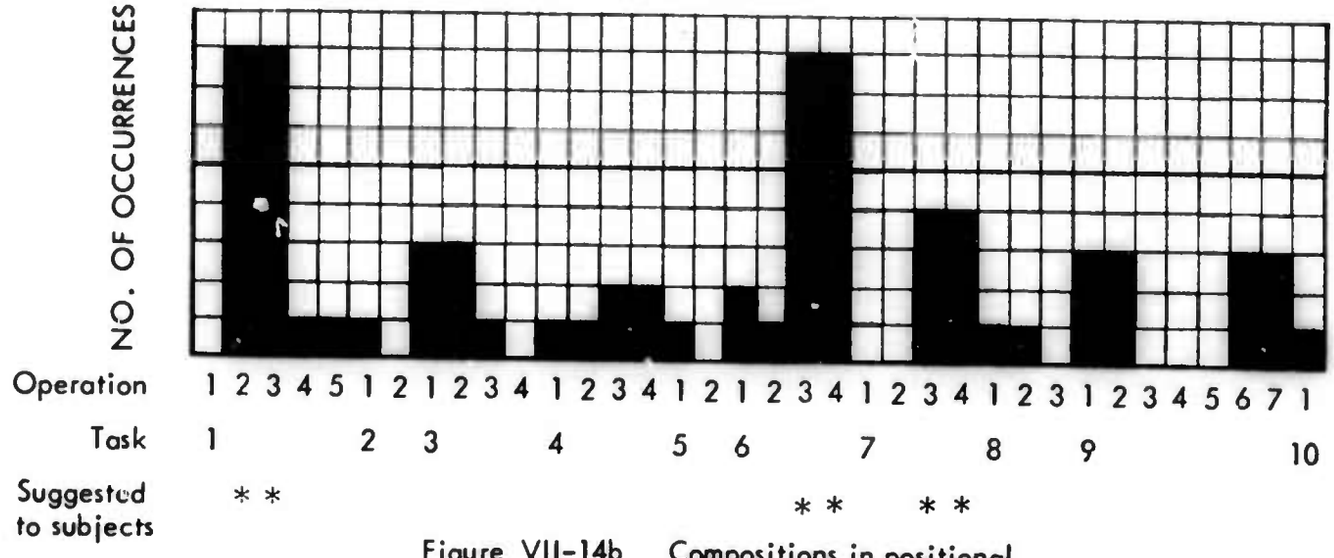



Figure VII-14b Compositions in positional

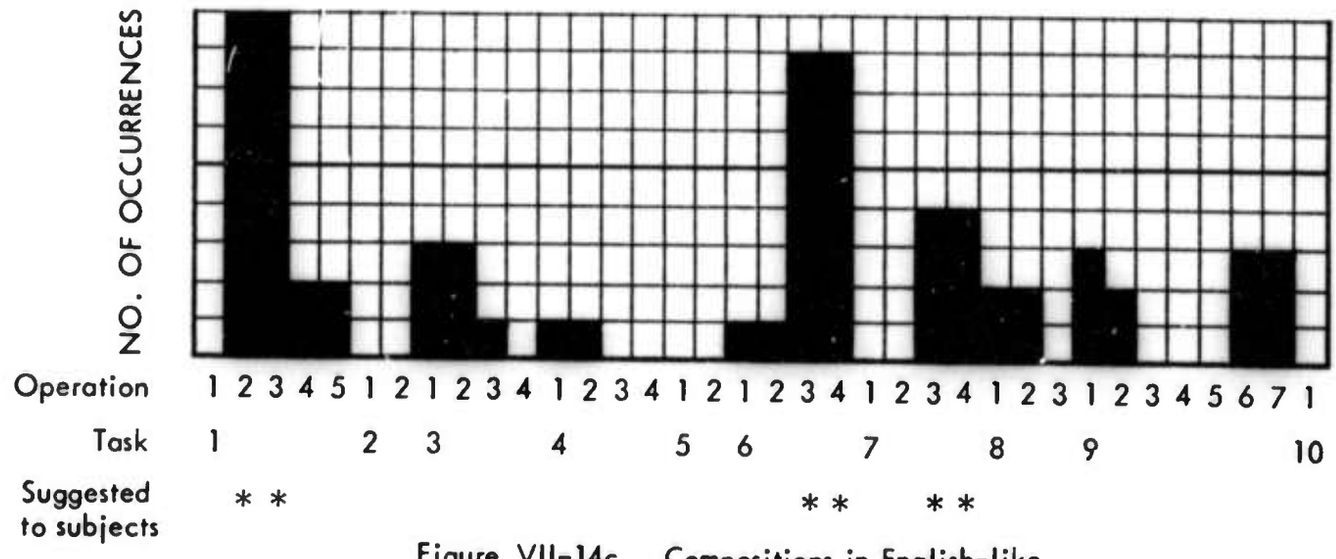



Figure VII-14c Compositions in English-like

## *CORRELATION FINDINGS*

### *Purpose of Reporting Correlations*

Correlation expresses the relationship between two variables. They are reported here to augment the analysis of variance findings in the following sense. If the language designer has determined from the analysis of variance results that certain kinds of improvements are needed, then he may use the correlation results to further assist him in determining specific changes to be made. Exactly how correlation can be useful in this way is illustrated below.

Two kinds of correlations are given: (1) the correlation between the languages (i.e., independent variable levels) in reference to a given dependent variable, and (2) the correlation between two dependent variables with respect to each language.

Let us first consider the correlations between pairs of languages. A strong relationship between two languages (with respect to some dependent variable) indicates that the subjects who tended to make many changes to one language also tended to make many changes to the other. As to the practicality of this observation, it may be possible for the designer to study these two languages together and to apply similar improvements to both of them, rather than considering them independently. Note, however, that the task inputs must be checked to verify that (where a strong relationship exists) the changes made by the subjects were of the same nature in each of the two languages. Conversely, a weak correlation between two languages suggests that the subjects' changes differ with respect to the dependent variable in question, which implies that potential modifications must be considered independently.

Now we consider the practical significance of the correlation between two dependent variables. A strong correlation gives the designer a different perspective or dimension from which to consider, then determine, improvements. For example, if reordering were related to contextual defaults, we might assume that this is really a causal relationship,* that the parameters were reordered to allow contextual defaults. Thus, rather than arranging the order of parameters simply from the standpoint of how the command "sounds", the designer would also take into account the likelihood of a contextual default.

To summarize, the correlations supplement the analysis of variance findings. Once the designer knows that an improvement is required, the correlation results may allow him to think about the languages simultaneously or they may give him a new point of view or provide more insight into the nature of the needed modifications.

---

*Correlation shows a relationship between variables. This does not automatically mean that it is a cause-and-effect relationship.

### *Correlations and Their Interpretations*

The particular coefficient of correlation that is calculated for our data is called the Pearson r [GUILFORD 73]. There are various correlation coefficients aimed at providing more relevancy for given kinds of data. The choice of the Pearson r is not that it yields the highest accuracy in the case of our data but rather that it is perhaps the most commonly used coefficient. At this stage of development of our protocol analysis, we are still concerned with gross trends and not a few percent accuracy one way or the other.

It should be pointed out that r is an index (not on a linear scale) of the relationship between two variables. It ranges in value from -1 (perfect inverse relationship) through 0 (no relationship) to +1 (perfect direct relationship). Since the index is not linear (e.g., 0.40 is not twice as strong as 0.20), we shall point out a (perhaps) more meaningful interpretation. The quantity $r^2$ is known as the coefficient of determination. Further, $100r^2$ expresses the percentage of variance in one variable that is accounted for by the variance in the other variable, or vice versa.

Somewhat arbitrarily, we discuss only those variables whose r value is 0.70 or greater for one or more of the languages, i.e., where about half or more of the variance in one variable is associated with the variance in the other. With our sample size of 9 the r=0.70 corresponds to a z value significant at the 0.95 level (in a two-tailed test). Specifically, for the two-tailed test (which means that we are not predicting the ***direction*** of the relationship), Z.95 = 1.96 and Z.99 = 2.58.

The correlation coefficient and the Z value are supplemented by a scatter diagram in each case to facilitate understanding.

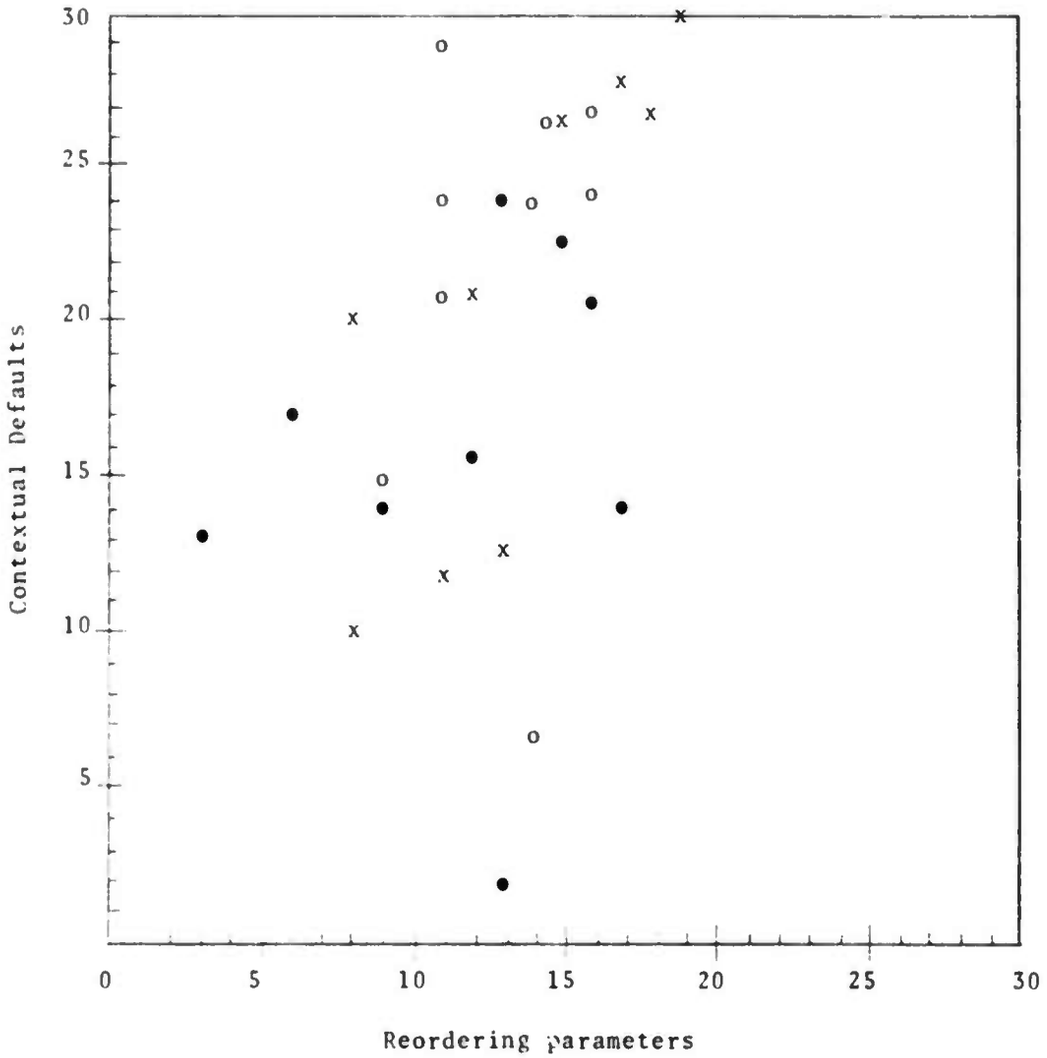
### *Reordering versus Contextual Defaults*

Reordering and contextual defaults are strongly correlated (r=0.81) in the positional language, where about 64 percent of the variance in reordering is accounted for by the variance in contextual defaults (see Fig. VII-15). This is significant beyond the 0.95 level (z=2.28).

The task inputs show that often when parameters were reordered, an originally embedded parameter was relocated at the right end and then defaulted. Therefore, we believe this to be a causal relationship. As a consequence, when determining the order of parameters in the positional language, the designer should consider the probability that a parameter could be established contextually.

This finding is even more interesting in light of the instructions given to the subjects. They were told that we were interested in determining the appropriate order of parameters for ***each*** language so that users might be taught the sequence most natural to them. [The "right" initial presentation to users should result in the most efficient use

of the language, be it keyword, positional, or English-like.] Thus each language was scored where there were permutations of parameters from the order appearing in the strawman languages. It is also evident from Fig. VII-15 that parameter arrangement was very important to the subjects in keyword and English-like; it simply is not related to contextual defaults in these cases.




| | | | |
|---|---|---|---|
| • | keyword | r = 0.45 | Z = 1.27 |
| x | positional | r = 0.81 | Z = 2.28 |
| o | English-like | r = 0.21 | Z = 0.60 |

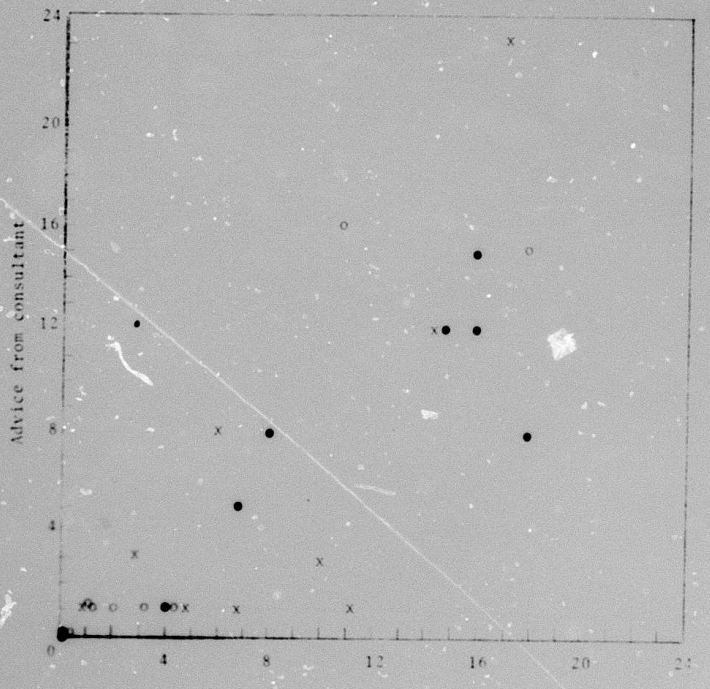
Fig. VII-15 - Reordering vs. contextual defaults

***References versus Advice***

References and advice are strongly correlated in each of the languages. In each case the result is significant between the 0.95 and the 0.99 levels; see Fig. VII-16.

This result is to be expected from experienced users, who, by virtue of their experience, are accustomed to using all channels of help at their disposal, i.e., both manuals and consultants. Evidently, neither source of aid sufficed, by itself, for these respondents. The practical inference is that the message system should sponsor both manuals and consultants.

Notice the slight tendency to use manuals more heavily than the consultant even though occasional unsolicited advice was given. This circumstance (prepared materials preferred to human tutor) is favorable to the projected use of on-line tutoring.




● keyword r = 0.90 Z = 2.55
x positional r = 0.72 Z = 2.04
o English-like r = 0.90 Z = 2.54

Fig. VII-16 - References vs. Advice

*References versus Errors*

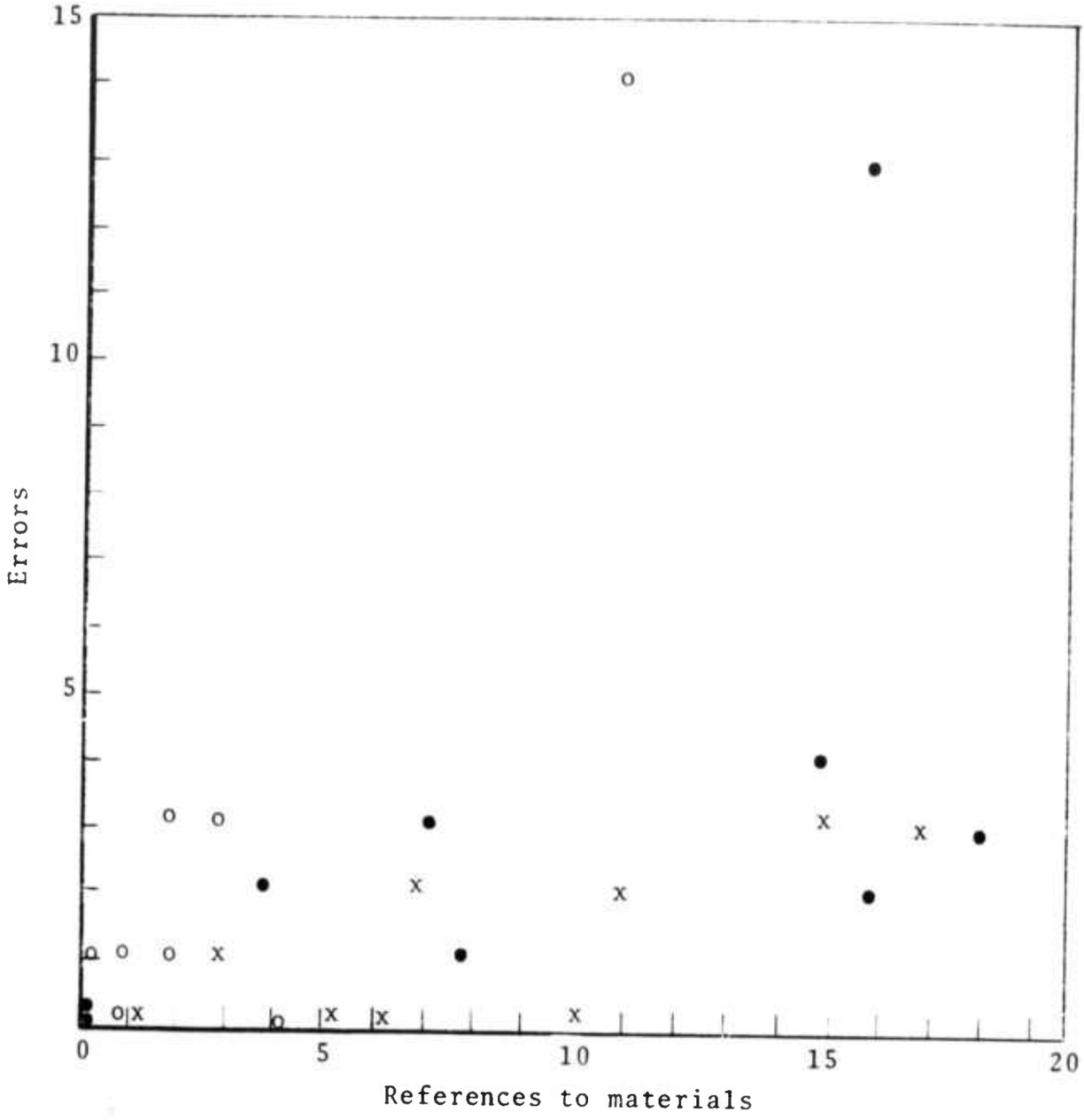
These two variables correlate in the positional language (See Fig. VII-17).

Had we hypothesized the outcome of this correlation, two opposing hypotheses would seem *a priori* plausible. One might suspect a negative correlation under the assumption that if the subject looked up a command then he would be likely to enter it correctly. Conversely, and in keeping with the finding, one might suppose that subjects would rely on the manual for operations of greatest difficulty. Even though the reference is consulted, there is a higher propensity to err simply because of the difficulty level of the operation. [This may, of course, be due in part to an inadequate exposition in the manual.]

A check of the task inputs validates that the errors did occur in those operations that were referenced in the manuals. Thus, if the manuals are to be effective they must be expanded. [They were intended to be reference manuals and not tutorials.] Many of the errors were semantic (not syntactic). An expanded manual that gives a more complete semantic description is advised for an operational environment. Furthermore, according to the subjects' remarks, the problem was due not to the *form* of the manual but to its *brevity*, and this in conjunction with a very concise indoctrination lecture led to errors. We do not view this as a failure of our objective in any sense, but rather as a clear indication of requirements for a potentially useful aid.

Two outliers are readily apparent in Fig. VII-17. These high error counts stem from the same individual. This particular subject stated that he could not work with an English-like language and would not use a keyword language. Ignoring these extreme deviants, it is clear from Fig. VII-17 that a high incidence of reference is associated with occurrence of errors in the keyword language as well. Thus the preceding comments apply to both positional and keyword.

Notice in Fig. VII-17 that references to materials in the English-like conditions tend to cluster in the 0 to 5 range. One interpretation of this observation (even though several subjects did not believe that the English-like language could ever be realized for a limited application such as the message service, over a homogeneous population of 7000 users) is that the English-like language is more intuitive than either of the others. Apparently the number of errors were about the same for each language (discarding the one subject) yet minimal references were made under the English-like conditions.

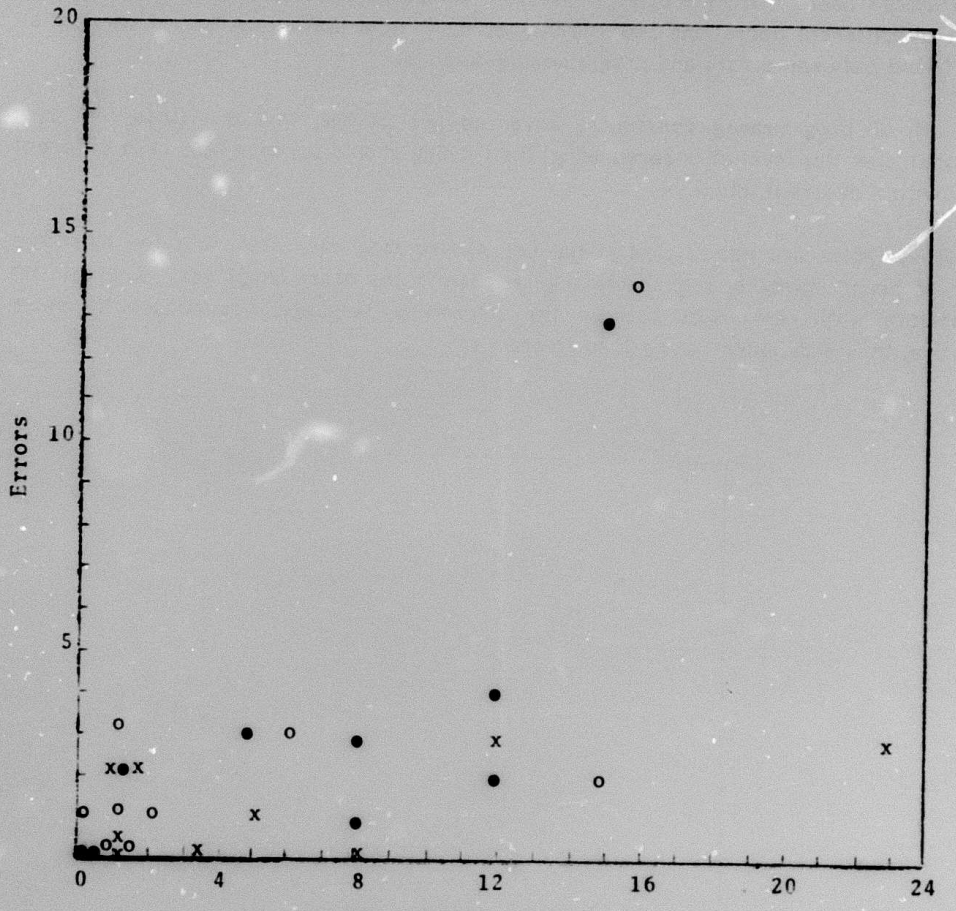



| | | | |
|---|---|---|---|
| • | keyword | r = 0.58 | Z = 1.63 |
| x | positional | r = 0.78 | Z = 2.20 |
| o | English-like | r = 0.44 | Z = 1.25 |

Fig. VII-17 - References vs. errors

***Advice versus Errors***

Advice and errors are correlated roughly equally in each language. Ignoring the two outliers, as before, Fig. VII-18 indicates that the relationship is most significant in keyword. Notice, again, the low clustering in English-like. From Figs. 16,17, and 18 we assume that advice is just another form of aid, used like references; thus previous arguments hold.




| | | | |
|---|---|---|---|
| ● | keyword | r = 0.71 | Z = 2.01 |
| x | positional | r = 0.61 | Z = 1.72 |
| o | English-like | r = 0.72 | Z = 2.04 |

Fig. VII-18 - Advice vs. errors

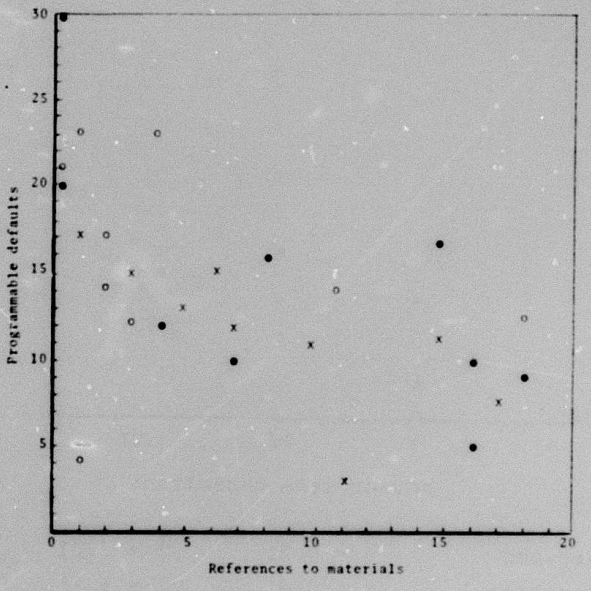
***References versus Programmable Defaults***

Programmable defaults are inversely related to references at approximately the same level in keyword and positional (See Fig. VII-19).

Subjects who took the initiative in preprogramming conditions of certain operations did not bother to look up the syntax of the applicable commands. These were the more experienced users. In view of Fig. VII-19, it should be pointed out that these scores occur on different operations and it is interesting to note that no significant correlation was found between errors and programmable defaults.

The art of programming constitutes advanced use of the service. Once the user demonstrates this level of understanding, the training should perhaps take on a different form with a different intent.

These variables are not related in the English-like language. Yet note the intensive user of programming in English-like, higher than in the other languages. There is no association with references because the natural-like language was apparently more intuitive, thus eliminating the need to reference.




| | | | |
|---|---|---|---|
| ● keyword | r = -0.71 | Z = -2.00 |
| x positional | r = -0.68 | Z = -1.93 |
| o English-like | r = -0.18 | Z = -0.51 |

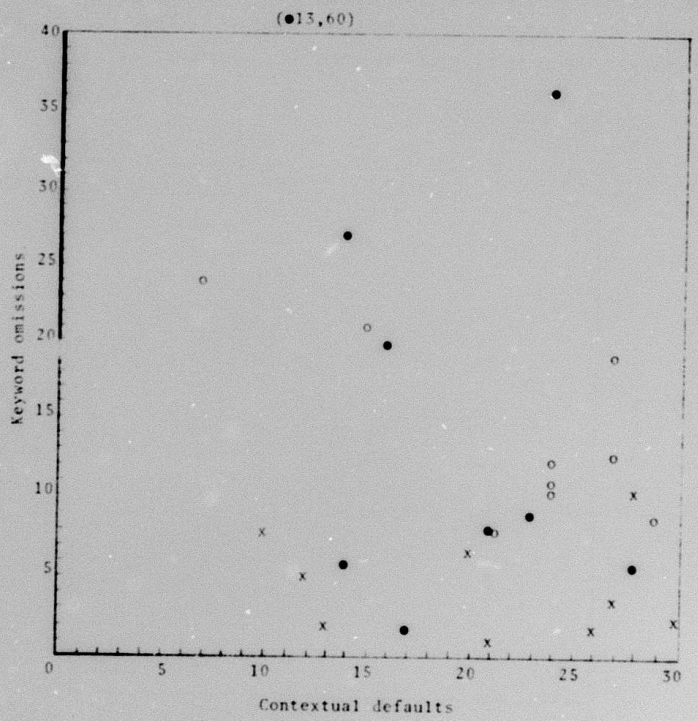
Fig. VII-19 - References vs. Programmable defaults

***Contextual Defaults versus Keyword Omissions***

These variables show a reciprocal relationship in English-like (See Fig. VII-20).

Fewer keywords appear in positional than in the two other languages. One might expect, then, as many omissions in keyword as in English-like. (Refer to the analysis of variance finding of keyword omissions.) Our interpretation is that in the more abstract language (keyword) users tend to follow the suggested form in contrast to the more natural language (English-like) where keywords were more often omitted since they would seem redundant in conversation.

If these assumptions are basically correct, then we attribute the correlation in English-like to the fact that if one defaults many parameters then this reduces the number available for potential keyword omission. Therefore, in English-like for parameters not commonly defaulted, the language designer might employ common noise words in recognition, in lieu of, or in addition to keywords.



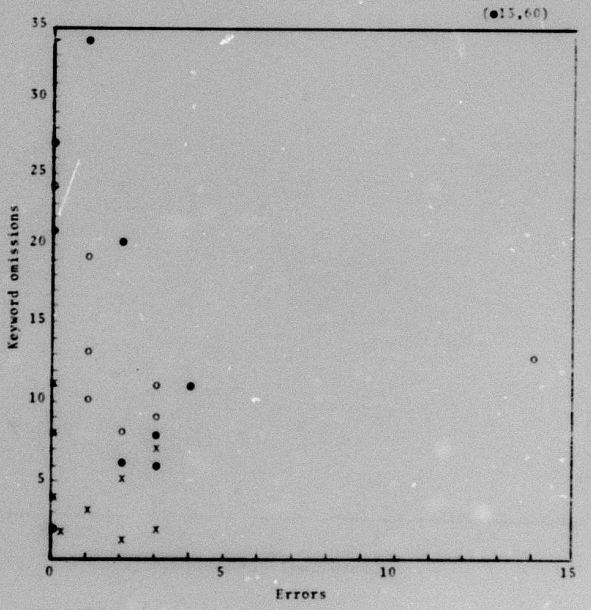

| | | | |
|---|---|---|---|
| ● keyword | r = -0.36 | Z = -1.02 |
| x positional | r = -0.06 | Z = -0.16 |
| o English-like | r = -0.70 | Z = -1.99 |

Fig. VII-20- Contextual defaults vs. keyword omissions

***Errors versus Keyword Omissions***

According to the calculations these variables are related in keyword, see Fig. VII-21. Ignoring the outlier (which is not plotted in the scatter diagram) the data for keyword does not satisfy the linear regression prerequisite for correlation. From Fig. VII-21 it is apparent that the data would be more accurately described by a polynomial of degree two.

Examination of the task inputs shows that we can think of the subjects as falling in one of three categories. In the first category subjects do not have a strong feeling for what parameters should be required for certain operations. To insure correctness they look up the command and mimic the example. Subjects in the second category were bolder but not more knowledgeable. Part of their (more often) incorrect syntax was the omission of keywords. The third category contains those subjects who omit many keywords while maintaining low error rates. The first category should not concern the designer since these diligent and conscientious users will, by virtue of their approach, develop into category 3 users. Subjects in the second category display deficiencies of the language. Analysis appropriate to improving the language is discussed elsewhere in this report (see analysis of variance of errors). The third category houses users who omitted keywords in a regular or consistent fashion. The particulars of these methodical changes are discussed under the analysis of variance earlier.




| | | | |
|---|---|---|---|
| ● keyword | r = 0.70 | Z = 1.97 |
| x positional | r = -0.31 | Z = -0.87 |
| o English-like | r = -0.28 | Z = -0.79 |

Fig. VII-21 - Errors vs. keyword omissions

***Programmable Defaults versus Delimiters***

These variables are directly proportional in English-like and inversely proportional in positional (see Fig. VII-22). [ Notice also the language-dependent separation of ranges. This is an interesting study in contrast of the language forms.]

In English-like, more advanced use (programming) accompanied substitutions of delimiters. These were not replacements for blank spaces but were more sophisticated replacements allied with a knowledge of parsing schemes. These subjects knew when and where such changes could be made, from the standpoint of parsing the input. It would be instructive for the designer to study the task inputs of punctuation changes in the English-like language as to alternate acceptable forms in that language. These forms may also be useful to the novice.

The reciprocal relationship in positional is a natural phenomenon. As discussed earlier, consistent changes to delimiters in positional were made to simplify typing. Where programming alleviated the necessity of parameters, fewer delimiters appeared in the commands in question. In this instance the designer should concentrate on fixing the punctuation in positional; the scatter diagram merely supports our earlier contentions.

In keyword there was not only more variety exhibited in the symbols substituted but also a large disparity in the instances of replacement. There is no significant correlation with programming.

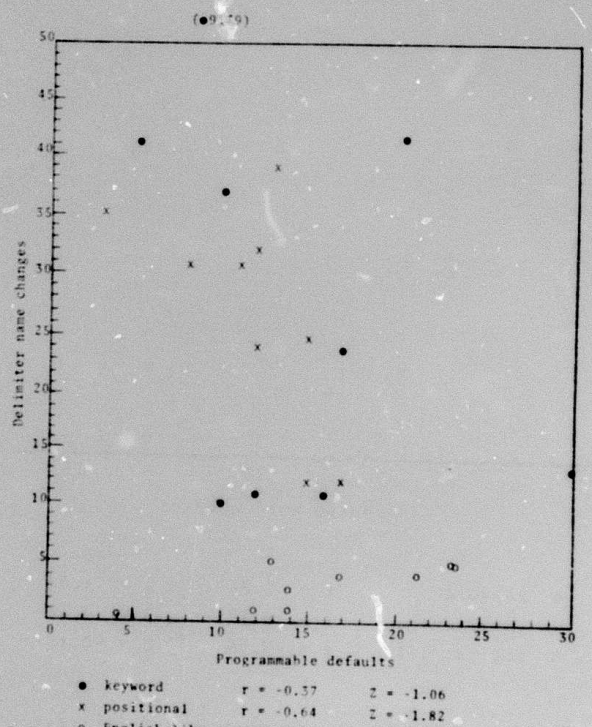
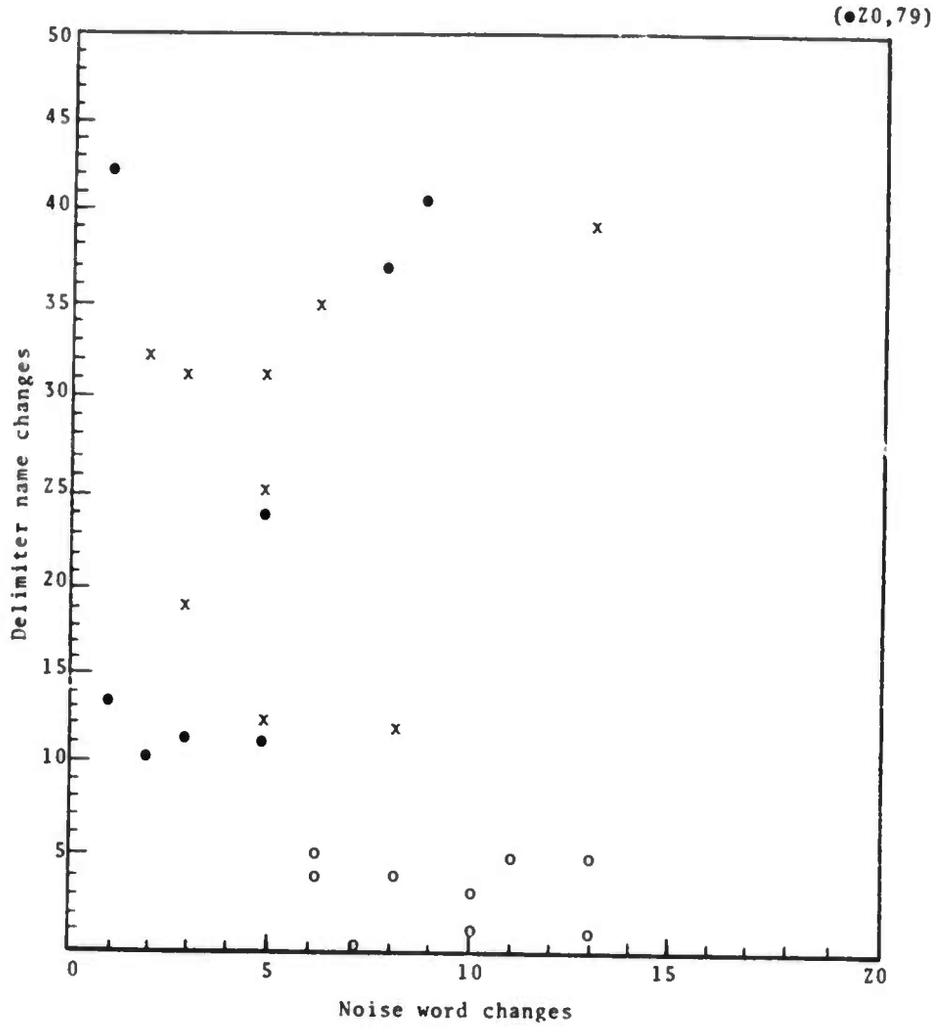



Fig. VII-22 - Programmable defaults vs. delimiters

***Noise Words versus Delimiters***

These variables are related in keyword. (See Fig. VII-23). Our best guess, and it is only a hunch, is that subjects were more aware of delimiters in the keyword language. As a result, when noise words were added they were frequently set off by special punctuation as if the subject felt it was essential for the parse. If this guess is correct, then it points to a need in training in the use of the keyword language.

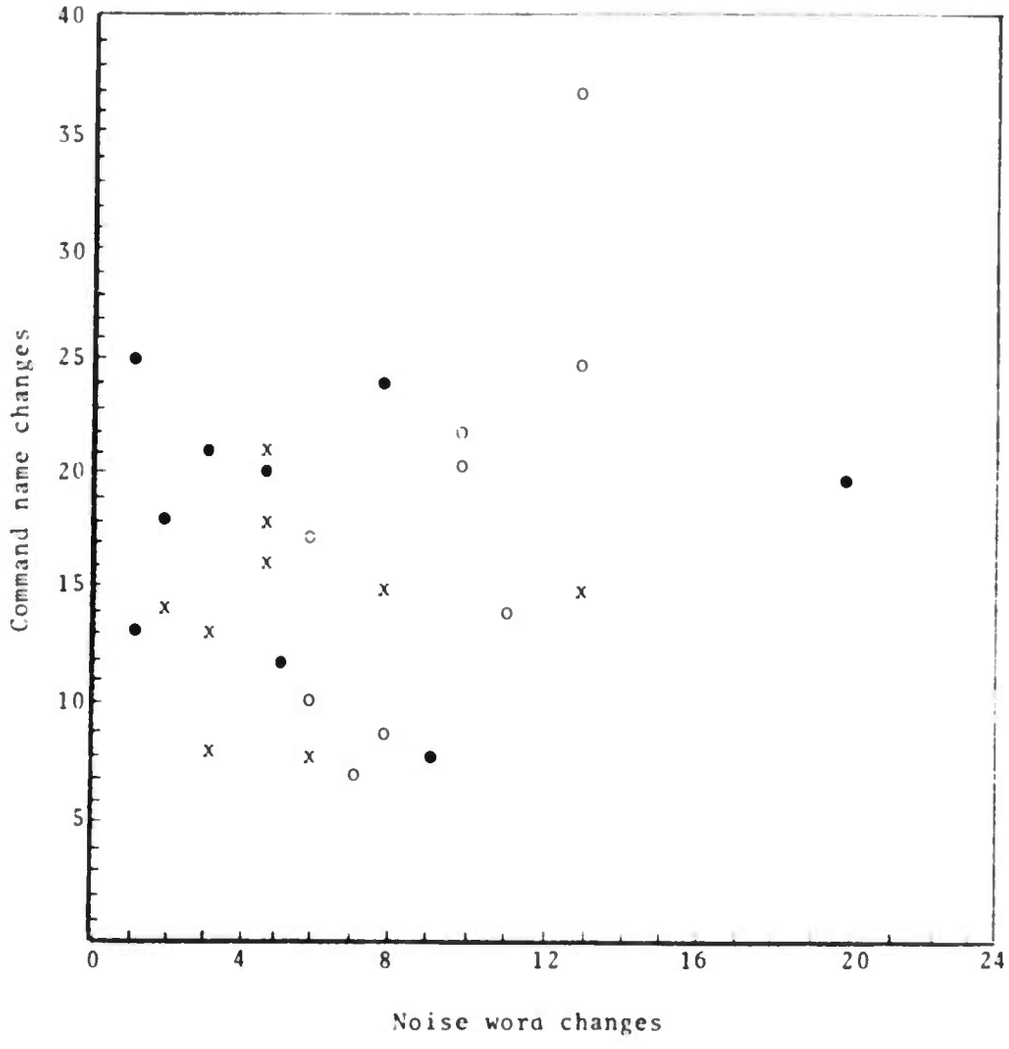



| | | | |
|---|---|---|---|
| • | yword | r = 0.84 | Z = 2.39 |
| x | positional | r = 0.17 | Z = 0.48 |
| o | English-like | r = -0.05 | Z = -0.15 |

Fig. VII-23 - Noise words vs. delimiters

*Noise Words versus Command Names*

Noise words are related to command name changes in the English-like language (see Fig. VII-24). Here new command names were often given as a phrase, that is, a new verb with attendant noise words. Noise words appeared less frequently and more randomly placed throughout the command in the other languages.

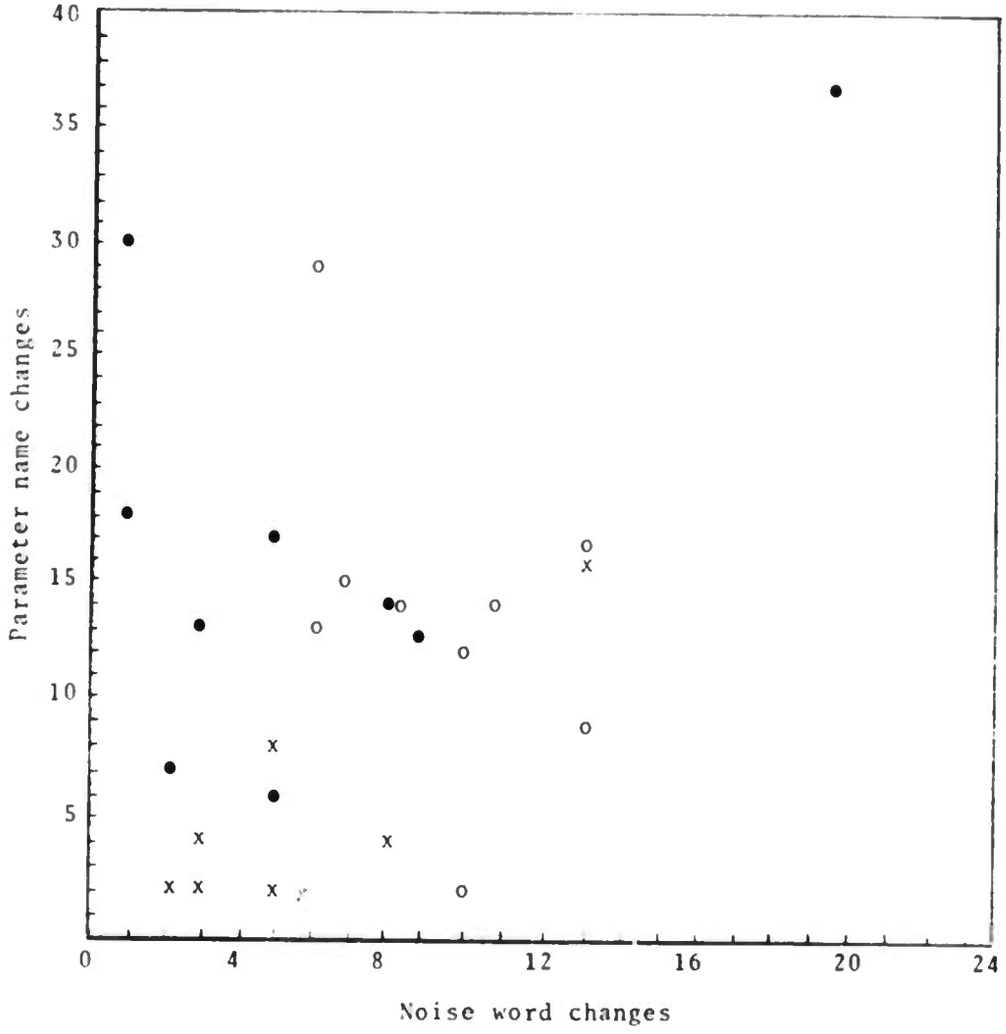



| | | | |
|---|---|---|---|
| ● | keyword | r = -0.02 | Z = -0.05 |
| x | positional | r = 0.16 | Z = 0.45 |
| o | English-like | r = 0.78 | Z = 2.19 |

Fig. VII-24 - Noise words vs. command names

*Noise Words versus Parameter Name Changes*

These variables are related in positional (see Fig. VII-25). As in noise words versus delimiters, we are not emphatic in explaining this relationship. Few parameter names were changed in positional. Most of them occurred where further qualification of an item was required, that is, in the more difficult operations. Consequently, subjects tended to add noise words in relation to their uncertainty in order to make their meaning clearer.

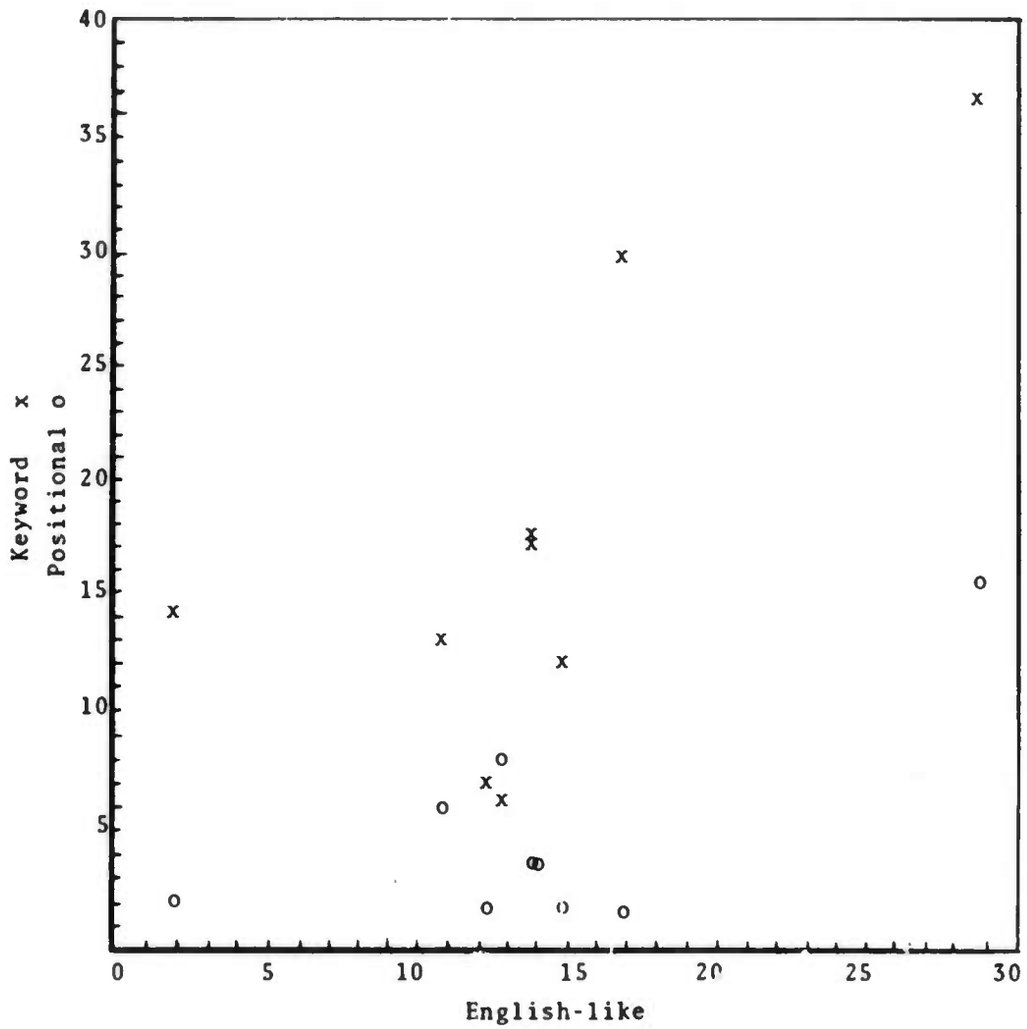



| | | | |
|---|---|---|---|
| ● | keyword | r = 0.52 | Z = 1.48 |
| x | positional | r = 0.81 | Z = 2.28 |
| o | English-like | r = -0.38 | Z = -1.07 |

Fig. VII-25 - Noise words vs. parameter names

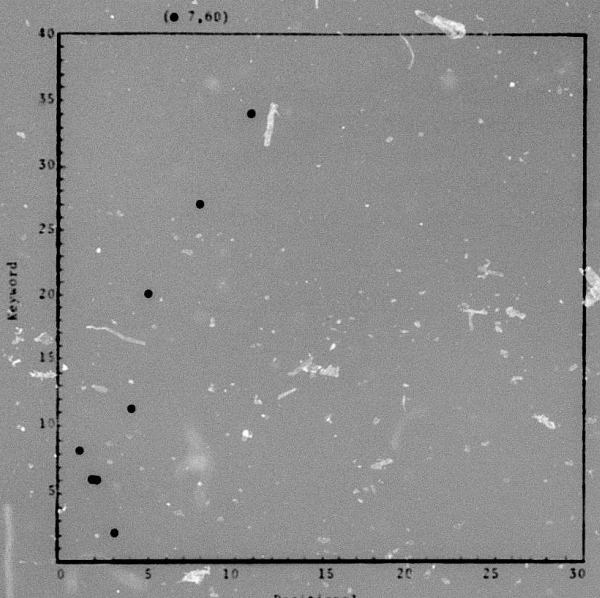
***Correlations of Levels of the Independent Variable***

Figures VII-26 through 32 show significant correlations of pairs of languages with respect to certain dependent variables. Where modifications are fostered by the analysis of variance, these correlation findings can be helpful in allowing the designer to concurrently consider several languages with respect to the planned changes. Again, the designer is cautioned to verify from the task inputs that the same ***kind*** of change was made in each of the correlated languages.



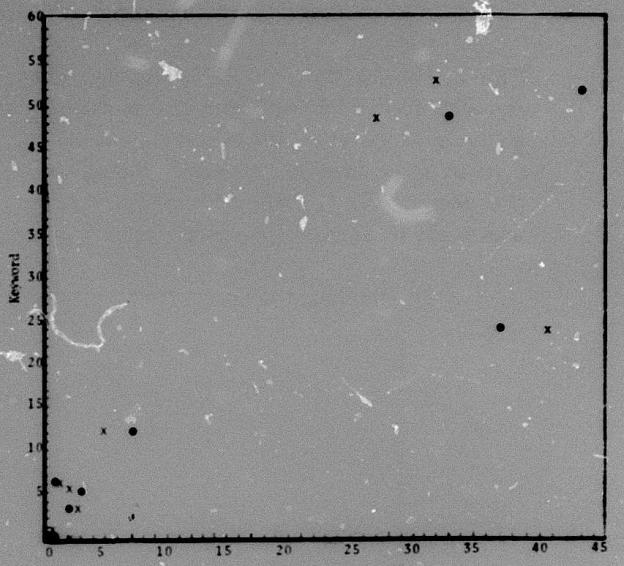

English-like/Keyword r = 0.72 Z = 2.17
English-like/Positional r = 0.73 Z = 2.20

Fig. VII-26 - Parameter name changes



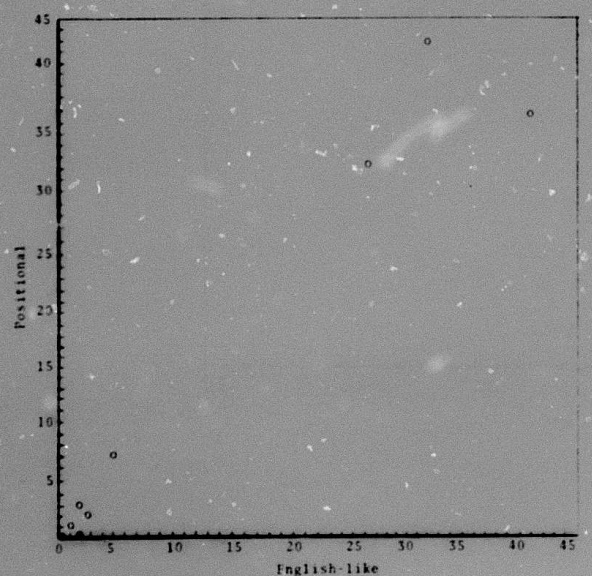

r = 0.73 Z = 2.19

Fig. VII-27 - Keyword omissions



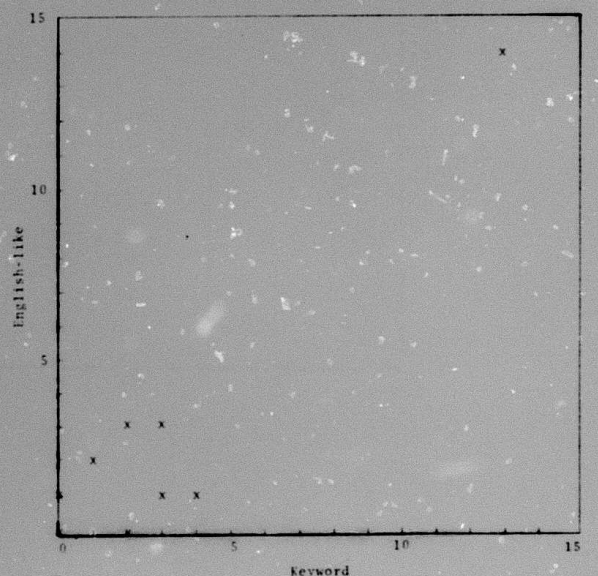

Keyword/Positional r = 0.93 Z = 2.78
Keyword/English-like r = 0.82 Z = 2.46

Fig. VII-28 - Abbreviations



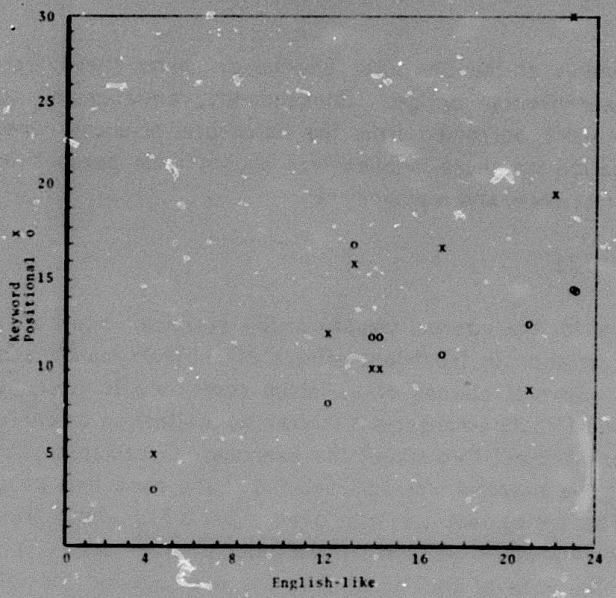

r = 0.97 Z = 2.91

Fig. VII-29 - Abbreviations



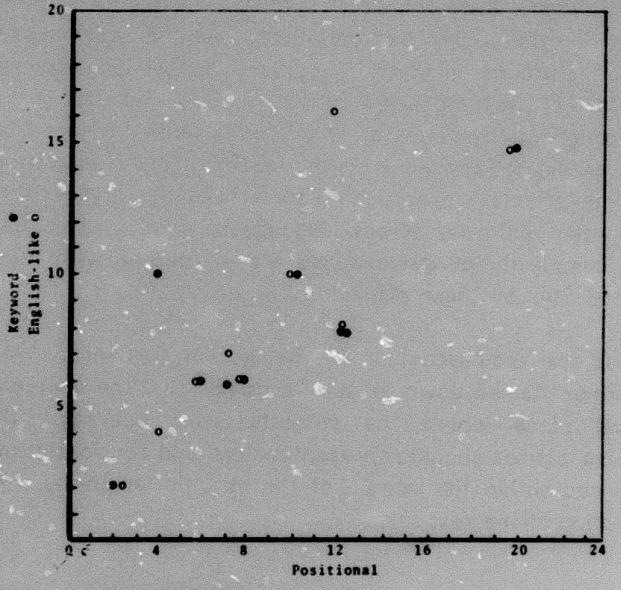

r = 0.93 Z = 2.80

Fig. VII-30 - Errors




English-like/Keyword r = 0.71 Z = 2.12
English-like/Positional r = 0.75 Z = 2.25

Fig. VII-31 - Programmable defaults




Positional/Keyword r = 0.80 Z = 2.39
Positional/English-like r = 0.87 Z = 2.61

Fig. VII-32 - Composition of commands

# VIII. EXPERIMENT VALIDITY AND GENERALITY

Owing to the newness of the Computer Science discipline, there are few established guidelines for experimental design. Consequently, environment controls for this protocol exercise were borrowed from the behavioral sciences, where they seemed suitable. Consultation on those applied was sought from persons in the behavioral sciences, computer science, and management.

## *INTERNAL VALIDITY*

Our first concern is for the internal validity of the exercise. Internal validity means the extent to which variation in treatment effects of subjects makes a difference in the results obtained. Several classes of variables contribute to or contaminate internal validity. An identical room arrangement (known as a situation or context variable) was maintained for each subject throughout the exercise. Circadian rhythm was controlled to the extent that the exercise was administered at the same time of day (1:30 P.M.) to each subject. [The entire test was run over a period of about three weeks.] Since there was a longer elapsed time between the indoctrination lecture (given to the group as a whole) and the time of participating in the exercise for the later subjects, the lecture was highlighted for each subject at the beginning of his exercise session.* That overview reinforced the role that the subject was to play and also pointed out the areas in which the subject was supposed to make free and natural choices, such as in the re-naming of language elements. Variables relating to the internal validity, such as the role selection (where subjects were to play the role of experienced computer professionals), are known as reactive measures. Another such measure from which errors can arise is maturation, in which the subject changes his response characteristics over time due to (1) learning more about the test (hence he responds more knowledgeably throughout the series of conditions because the learning effects are not erasable), and (2) becoming fatigued when the session exceeds his attention span. Both fatigue and learning effects are accounted for inherently in the Latin square design, which distributes them uniformly across language conditions. It should be noted that the length** and complexity of the exercise are in keeping with the typical working session experienced daily by these subjects.

Another concern is the acquiescent or the belligerent response. We assert that the subjects were neither "talked down to" nor "snowed"; that the exercise was conducted on an intellectual plane where the subjects were peers of the interviewers. Consequently the responses should represent mature and considered opinions as to the subjects' preferences within the domain of the specific languages. The interviewers shared this perception of the responses.

---

*A comparison of mean scores for the first three subjects and the last three subjects showed no significant variation due to this treatment effect.

**Session duration averaged about 2 hours and 15 minutes.

Addressing now the consultant and observer,* the maturation effect is again considered (here called instrumentation) whereby changes in the presentation and responses by the consultant and observer can be manifested due to learning effects over time. For example, subtle changes in the instructions or interviews can significantly change the outcome. To reduce these effects by making instructions non-implicational (and at the same time to improve the test items) several dry runs were conducted to stabilize the consultant and observer protocols. Interviewer's fatigue was negligible, since only one session was held per day. Probably the most prevailing error source was rating errors, i.e., the interpretation and encoding of subjects' responses by the observer. By and large the classification of a response is objective, yet in certain situations it is subjective. For example, in some responses a word could be interpreted either as a parameter substitution or as a noise word with the parameter omitted. In order to score such responses consistently the checklist data were verified (for each subject) on two separate occasions by playing back the tape cassettes and checking against (reconsidering) the observations scored during the exercise.

Test variables may interact with other variables, such as age or sex of subject, which are beyond the purview of the study. There was no practical way to identify and control these variables. Our sample, which comes from a very homogeneous population, is simply described in terms of general characteristics from which more specific random variables could be inferred, should they be considered relevant in interpreting the results.

There may be an experimental bias due to the substitution of verbal interaction for the typing which normally is required for man-computer interaction. We did not compare modality effects. Since the purpose of this test was to study input syntactic forms, we feel that the advantage gained by removing terminal type bias from the data outweighs the disadvantages of this mode substitution.

### *EXTERNAL VALIDITY*

While establishing internal validity is essential, our prominent concern is with external validity. That is, to what extent in the dimensions of both the independent variable (languages) and the population of users can we generalize the results of this study? The answer, unfortunately is "not very far". Looking first at the languages under study, the forms chosen are realistic and relevant in the sense that they commonly appear as problem-oriented command languages for non-programmers. Yet, care must be taken not to extrapolate the results to a different man-machine interaction setting with the same or hybrid language forms. For example, use of a graphic I/O device or, as another example, use of a purely demand/response, user-driven system, might inject confounding or cancelling effects.

---

*Two analysts played the roles of simulator and observer throughout the exercise. Both of them acted as consultants at all times.

The largest population to which generalizations could at all safely be made is that of experienced TENEX [BOBROW 72] *programmers*. We have definitely not treated the population of military Action Officers. Absolutely no parallels should be drawn between these two populations. The subjects of the exercise were all experienced computer professionals. Their strongest common characteristic is knowledge of TENEX, though the breadth and depth of their computer training varies. Each subject uses TENEX daily, and has for several years. Rudimentary examination of the session tapes points out that, although subjects were instructed to loosely stay within the syntactic framework of each of the languages yet be as inventive as they desired in tailoring each language to their personal disposition, those subjects with a broader exposure to other on-line systems tended to do just that, whereas those with a lesser working knowledge of other systems tended to revert to TENEX-like commands for difficult-to-compose operations, regardless of the language they were using. To put it another way, it is quite possible that if the experiment were repeated using subjects whose strongest common denominator were MULTICS [ORGANICK 72], that different results might obtain. Hence, we confine the population, of which we believe this sample group to be representative, to be experienced TENEX programmers. There is even considerable danger in so large a population as this, as witnessed by the concordance test for experienced users (see Appendix G). That result, however, tends toward the anticipated result (positional) and with so small a sample size it would be unwise to state a strong conclusion either way.

Lastly, one might justifiably ask if there are alternate plausible hypotheses, in some cases, to the research hypotheses set forth. As a safeguard against rivals, the research hypotheses were reviewed by several people. Before the protocol test six persons were asked to analyze the hypotheses: a behavioral scientist, a manager, and four computer scientists. Their task was to agree, disagree, or offer no opinion on each hypothesis. Where disagreement arose, they were to state an alternate hypothesis for consideration. Responses were obtained from the manager and three of the computer scientists. Essentially, disagreements were in degree or intensity of the projected significance of the outcome. In no case was there an opposing explanation offered to describe the same expected result.

# IX. CONCLUSIONS AND RECOMMENDATIONS

## *PROTOCOL ANALYSIS: AN EFFECTIVE PROCESS*

Central to this study is the question of whether or not protocol analysis is an effective practice as an aid in designing languages for people inexperienced in the use of computers. The reason for choosing the protocol analysis form of exercise and interview as opposed to, say, a questionnaire, is as follows. The exercise permits an in depth study and interaction with the subject which ensures that he understands what is expected of him. The task part consists of closed items which imply forced responses that can be modeled and measured. The quasi-structured interview is open-ended to possibly acquire information not anticipated. Disadvantages are that the protocol analysis is costly, time-consuming, and inconvenient.

Conclusions stated in Chapter VII and to some extent in Appendix G are incidental in the sense that the tested sample comes from a population not of primary interest. Yet the fact that significant findings were obtained is supportive of our contention that protocol analysis is a useful practice. Hence, we conclude, from the results of our example application, that it probably is effective and thus a similar experiment should be carried out with the target population.

## *IMPROVEMENTS IN THE DESIGN*

The main goal of the "dry run" was to provide an instance of use of the protocol analysis so that its merits might be judged for its possible application to the population of Action Officers. Given that it is indeed appropriate, an ancillary goal is to determine changes that should be made to the exercise. Based on the dry run, the following are suggested as improvements.

1. It is evident from some of the findings that a larger sample is needed. Both reliability and the power of the statistical tests applied to the data increase with an increase of sample size. There are, of course, questions of economy, convenience, availability of subjects, and other factors that are outside the considerations for statistical reliability of the study *per se*. However, a sample size of 20 to 30 subjects is strongly recommended.

2. Given a larger sample, to achieve some economies of scale it is advised that the task be shortened by eliminating some of the redundant operations. Total reliability would not be expected to suffer greatly by reducing the number of operations from 36 to 20. It is suggested that an amended Appendix B be discussed with the potential test subjects to determine its relevance prior to administering the exercise. This allows one more iteration of refinement.

3. Some additional dependent variables should be included, e.g., addition of parameters, addition of commands, and decomposition of commands. Inclusion of more dependent variables places no burden on the subjects and little more in the way of analysis, while the extra variables provide more information to the designers which results in more accurately specified languages.

4. A few specific commands used in the test are believed to be wrong for any population. They should be corrected.

5. Difficulties were experienced with several of the cassette tapes; two post-task interviews were lost. It was earlier prescribed that more reliable equipment be employed for the exercise, since such equipment is a very minor part of the total cost of the experiment. We reinforce that sanction.

6. Comments from the designers (on a draft of this report) indicate that, in their opinion, the interview material would be useful for planning service functions and training aids. Consequently, it is suggested that the interview portion of the exercise include some specific questions on these subjects so that they may be analyzed in a manner similar to the concordance tests of languages.

## *COST AND TIME FACTORS*

A necessary consideration for the application of similar protocol analysis experiments is total cost--cost affecting the scope of the test, hence the specific research questions and subsequent analysis which impact the implications and conclusions, and effectively the study's utility. Overall costs will naturally vary according to the particulars of the test purpose and other factors. Thus, rather than listing dollar amounts, major cost factors for this example test are identified, an estimate of each factor is given in appropriate terms, conditions bearing on this cost are stated, and the expected variability of costs from experiment to experiment is judged. Table IX-1 condenses this information.

Another crucial consideration is elapsed time from test inception to final report. Table IX-2 provides time estimates. Design, administration and analysis are effectively sequential processes, whereas parts of the report preparation may proceed concurrently with each of the sequential steps.

Table IX-1
Major cost factors of protocol analysis

| FACTOR | EXPENDITURE | CONDITIONS | EXPECTED VARIABILITY |
|---|---|---|---|
| Research design<br>problem identification<br>background study<br>research questions<br>hypothesis generation<br>data collection methods<br>analysis methods<br>etc. | 2½ man-months | given familiarity with research methods & overall objectives of study | high |
| Language specification | 1½ man-months | given description of service functions | moderate |
| Statistical computation<br>program development<br>production | ½ man-month<br>30 min. CPU | | low |
| Analysis | 2 man-months | | low |
| Printed materials used in test | ≈$100 | | low |
| Recording equipment | $50 - $500 | range of equipment from single cassette to rental multi-channel recording | moderate |
| People<br>subjects of interviews | ½ man-month | | low |
| Report preparation<br>secretarial | 6 man-months | given on-line text editing facilities | moderate to high |

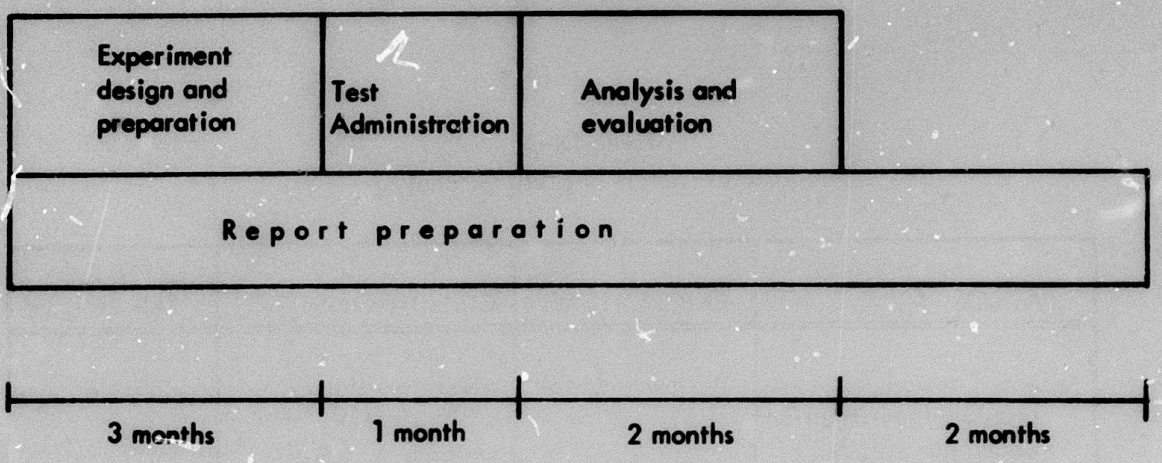



**Table IX-2 - Example elapsed time for protocol analysis**

## *SUGGESTIONS FOR ADDITIONAL STUDIES*

Chapter I alludes to a thesis--that models of users and models of services can be accurately used as predictors in selecting a language form for a population of individuals using an application, and that this language form will result in high users' performance. This thesis is suggested in the introduction by stating that we wish to develop representative languages, via protocol analysis, for later tests of users' performance, and further that we restrict the study to a specific user type and service kind. We wish to extend that thesis in that even with the development of such empirical laws as a method of language selection, a language thus chosen will still be only an approximation (in terms of yielding optimal users' performance) due to variations within the population of users and deviations from the mean of the service category. Thus, in addition to selection methods based on user and service, there is need for on-line regulation methods which, can be employed by the user to modify language constructs to further improve performance.

A framework for models of users and services is given in [HEAFNER 74] which may lead to a ***selection methodology*** for choosing tractable languages. Likewise, several language refinement algorithms (***dialogue regulation methods***) are proposed in that report. These two components presuppose protocol analysis pilot studies.

We state as a tautology that as the use of on-line computers by non-computer professionals increases, so does the need for efficient and effective man-computer communications. To be useful, then, a theory (of man-machine dialogue) should be well founded and time-tested. Thus, we recommend additional protocol analysis studies and also performance studies in order to both improve the methods and to provide a baseline for language selection from a number of language families covering various user populations and application categories.

# APPENDIXES

*NOTICE*

*Due to the large size of the appendixes for this document, they have been removed and a limited number printed as a supplement. If you did not receive the supplement, but need one, a copy may be requested from ISI while the supply lasts after which it will be available from National Technical Information Service, Springfield, Virginia 22151.*

## REFERENCES

[ARPA]
Session entitled *The ARPA Network*, AFIPS Conference Proceedings, Vol. 40, 1972.

[BABBIE 73]
Babbie, E. R., *Survey Research Methods*, Wadsworth Publishing Company, Inc , 1973.

[BOIES 74]
Boies, S. J., *User Behavior on an Interactive Computer System*, IBM Systems Journal, Vol. 13, No. 1, 1974.

[BOBROW 72]
Bobrow, D. G., J. D. Burchfiel, D. L. Murphy and R. S. Tomlinson, *TENEX, A Paged Time Sharing System for the PDP-10*, Communications of the ACM, March 1972.

[CAMPBELL and STANLEY 63]
Campbell, D. T. and J. C. Stanley, *Experimental and Quasi-experimental Designs for Research*, Rand McNally and Company, 1963.

[DEC]
*DEC System 10 Users Handbook*, Digital Equipment Corporation, Maynard, Massachusetts 01754.

[DED]
DED, Diminunitive Editor, USC/Information Sciences Institute internal documentation. Contact Donald Oestreicher.

[DOWNIE 59]
Downie, N. M., and R. W. Heath, *Basic Statistical Methods*, Harper and Brothers, New York, 1959.

[ELLIS 73]
Ellis, T., L. Gallenson, J. F. Heafner, and J. T. Melvin, *A Plan for Consolidation and Automation of Military Telecommunications on Oahu*, USC/Information Sciences Institute, ISI/RR-73-12, May 1973.

[ELLIS 70]
Ellis, T. O., J. F. Heafner and W. L. Sibley, *The GRAIL Project*, Proceedings of the Society for Information Display, Vol. 11, No. 3, 1970.

[GUILFORD 73]
Guilford, J. P., and Benjamin Fruchter, ***Fundamental Statistics in Psychology and Education***, McGraw-Hill Book Company, Inc., 1973.

[HEAFNER 74]
Heafner, J. F., ***A Methodology for Selecting and Refining Man-Computer Languages to Improve Users' Performance***, USC/Information Sciences Institute, ISI/RR-74-21, September 1974.

[LEAVITT 73]
Leavitt, E., T. Strollo and S. Shapiro, ***TENEX User's Guide***, Bolt Beranek and Newman Inc., 50 Moulton Street, Cambridge, Massachusetts 02138.

[MICHAEL]
Michael, W. B., Prof. Ed. Psychology, University of Southern Calif., ***Conceptual Steps in the Statement of the Null Hypothesis in the Instance of Two-tailed Test of the Significance of a Difference Between Means (Large Sample Approach)***, unpublished paper.

[MILLER 73]
Miller, E. F. (editor), ***Bibliography and KWIC Index on Computer Performance Measurement***, General Research Corporation, RM-1809, June 1973.

[MYER 73]
Myer, T. H., J. R. Barnaby and W. W. Plummer, ***TENEX Executive Manual***, Bolt Beranek and Newman Inc., 50 Moulton Street, Cambridge, Massachusetts 02138.

[OESTREICHER 74]
Oestreicher, D. R., J. F. Heafner, and J. G. Rothenberg, ***CONNECT: A User-Oriented Communications Service***, ACM Annual Conference, November 1974.

[ORGANICK 72]
Organick, E. I., ***The MULTICS System***, MIT Press, OGM-ISIBN 0-262-150-3, 1972.

[RICHARDSON]
***FILE/WATCH: User File Monitor***, USC/Information Sciences Institute internal documentation. Contact Lee Richardson.

[SRI/ARC]
***ARPA Network, Current Network Protocols***, ARPA Network Information Center, Augmentation Research Center, Stanford Research Institute, Menlo Park, California 94025.

[TUGENDER 74]
Tugender, R., and D. R. Oestreicher, ***Basic Functional Capabilities for a Military Message Processing Service***, ISI/RR-74-23, May 1975.

[WINER 71]
Winer, B. J., ***Statistical Principles in Experimental Design***, McGraw-Hill Book Company, Inc., 1971.

[XED]
***XED - Experimental Editor***, Short Reference Manual, USC/Information Sciences Institute internal documentation. Contact Donald Oestreicher or Ronald Tugender.

[YONKE 75]
Yonke, M.D., ***Bananard Reference Manual***, USC/Information Sciences Institute, ISI/TM-75-2. in progress.