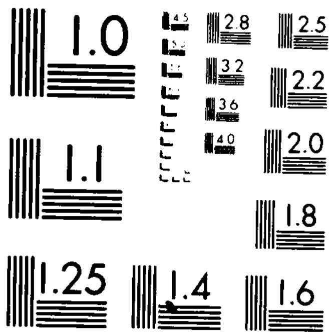
AD-A170 660 HUMAN PERFORMANCE ASPECTS OF SMALL SCREEN DISPLAYS: A LITERATURE REVIEW R..(U) COMPUTER TECHNOLOGY ASSOCIATES INC MCLEAN VA R L SHANNON ET AL. MAY 86 HEL-TN-5-86 1/1

UNCLASSIFIED DAAK11-84-D-0008 F/G 5/10 NL


1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0

12

AD

Technical Note 5-86

HUMAN PERFORMANCE ASPECTS OF SMALL SCREEN DISPLAYS: A LITERATURE REVIEW REVEALING THE LACK OF SPECIFIC RESEARCH

Rebecca L. Shannon
Lisa J. Stewart

May 1986

Approved for public release;
distribution unlimited.

U. S. ARMY HUMAN ENGINEERING LABORATORY
Aberdeen Proving Ground, Maryland

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER Technical Note 5-86	2. GOVT ACCESSION NO. ADA 170660	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) HUMAN PERFORMANCE ASPECTS OF SMALL SCREEN DISPLAYS: A LITERATURE REVIEW REVEALING THE LACK OF SPECIFIC RESEARCH		5. TYPE OF REPORT & PERIOD COVERED Final 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Rebecca L. Shannon Lisa J. Stewart		8. CONTRACT OR GRANT NUMBER(s) DAAK11-84-D-0008
9. PERFORMING ORGANIZATION NAME AND ADDRESS Computer Technology Associates, Inc. 7927 Jones Branch Drive, Suite 600 W McLean, VA 22102		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Human Engineering Laboratory Aberdeen Proving Ground, MD 21005-5001		12. REPORT DATE May 1986 13. NUMBER OF PAGES 55
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release;
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)

Human Factors Engineering	Protocols	Standards
Human Factors	Text Editing	Liquid Crystal Displays (LCD)
Small Screen Displays	Abbreviations	Guidelines
Displays	Command Naming	CRT Displays

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

This report contains the results of a literature review conducted for the Human Engineering Laboratory. Literature from diverse disciplines pertinent to the human performance aspects of small screen displays was examined. In particular, four subtopics were focused on. They include: display size, defined as number of characters per number of lines; protocols, defined as ways users interact with a system to achieve their goals; abbreviations; and vocabulary. The results indicate the lack of

DD FORM 1473 1 JAN 73 EDITION OF 1 NOV 65 IS OBSOLETE

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered) specific research on small screen displays but also indicate a trend in current research interest and towards interest in the near future. The report points out other research trends and voids and contains an annotated bibliography of more than 100 source documents.

Technical Note 5-86

HUMAN PERFORMANCE ASPECTS OF SMALL SCREEN DISPLAYS: A LITERATURE REVIEW REVEALING THE LACK OF SPECIFIC RESEARCH

Rebecca L. Shannon
Lisa J. Stewart

Computer Technology Associates, Inc.
7927 Jones Branch Drive, Suite 600 W
McLean, VA 22102

May 1986

APPROVED: ______________________
JOHN D. WEISZ
Director
Human Engineering Laboratory

Approved for public release;
distribution unlimited.

Human Engineering Laboratory
Aberdeen Proving Ground, Maryland 21005-5001

PREFACE

This report represents the findings of a literature review conducted under contract number DAAK11-84-D-0008 for the Human Engineering Laboratory.

The authors wish to thank John Mays and Ed Harras of CALCULON Corporation for their assistance in preparing this document.

They also wish to thank Salvatore P. Schipani of the Human Engineering Laboratory for his efforts in overseeing the report through to publication.

CONTENTS

EXECUTIVE SUMMARY

A representative sample of documentation pertaining to the use of small screen displays on computers was obtained from academic and professional journals, computer-aided listing services, and through contacts with human factors consultants, industry representatives, and authors of relevant research. Literature from commercial as well as military applications was included.

For the purpose of this report, any screen displaying fewer than 80 characters per line and displaying fewer than 24 lines per screen, or a combination of both, was considered to be a small screen.

Interest in the topic of small screen displays appears to have been prompted by an interaction between available display technology and changing display application. Research indicates that various display technologies (e.g., *LED, CRT, LCD, EL) were developed for the display of* alphanumerics or numerics on equipment with limited or no interactive capabilities, like calculators, appliances, watches, and clocks (Snyder, 1980). In interactive environments like airplane cockpits, power plant control centers, or a ship's bridge, small screens have also been used primarily to read or enter alphanumeric character strings, not for the broader and more consistently interactive task applications. However, the advent of portable computers and other hand-held interactive electronic devices necessitated the use of small screens that could support highly interactive task applications. As a result, there has been increased attention to the human performance issues.

In contrast to display size, protocols were addressed in an abundance of literature, although indirectly. The topic of protocols, a complex one, is affected by factors involving the user, task, and system. Relevant information was often buried within discussions of system comparisons, designed case studies, and modeling efforts. Because this is a broad topic researched from several perspectives, few commonly accepted guidelines and standards exist.

Studies investigating abbreviation strategies and human performance generally reported truncation as the best method overall. Several variables like task application appear to interact, affecting optimum strategy. Though some standards on abbreviations exist, they are vague and may need updating. Abbreviation methods for small screens are not addressed, although the need for such research is alluded to by only one set of guidelines (Hendricks, Kilduff, Brooks, Marshak, & Doyle, 1983).

The majority of the vocabulary "standards" may be more accurately described as general rules of thumb for determining human-computer interface vocabularies. One exception, a guidelines document, presents a list of recommended command names (Hendricks, Kilduff, Brooks, Marshak, & Doyle, 1983).

Overall, research pertaining to the relationship between human performance and the size of small screen displays is sparse. The majority of what exists is very recent, creating the presumption that this topic is under current ongoing research that has not yet been published (Rogers, 1983, 1984).

This literature review revealed a large number of articles about human performance and vocabulary. Research indicated that vocabulary is as broad a topic as protocols; as a result, vocabulary was investigated within several frameworks. Studies addressing command languages, menus, natural language, and icons all contributed relevant but diverse results to the topic of vocabulary. There is a void in the research concerning vocabulary schemes for small screen displays, indicating a need for further investigation.

Further research on the effects of small screen computer displays on human performance is strongly recommended.

HUMAN PERFORMANCE ASPECTS OF SMALL SCREEN DISPLAYS: A LITERATURE REVIEW REVEALING THE LACK OF SPECIFIC RESEARCH

INTRODUCTION

As electronic display technologies become more advanced, and as they are integrated more into various products and systems, the demand for human factors considerations increases. More people are working with displays, increasing the need for better design. As a result, measuring human performance becomes important. This document reviews the literature pertaining to the human performance aspects of displays, including those few found on small screen displays.

Four subtopics are discussed in this report. The first is display size which addresses the number of characters per line and the number of lines per screen, and the effect on the physical size of the display. The second subtopic is protocols. Protocols are the ways individuals interact with a system to accomplish their objectives. This report focuses on text editing tasks and results of the pertinent human performance research. The third subtopic, abbreviations, arises as an extension of display size. As display size decreases, text must often become abbreviated. The human performance study addressing optimum ways to abbreviate text is discussed. Vocabulary is the fourth subtopic, defined as ways to name methods through which users interact with a system. The three subtopics, protocols, abbreviations, and vocabulary, are included in greater detail under the subheadings of small screen specific and guidelines or standards. Small screen specific is a term used for those screens which have been considerably reduced in size, allowing them to fit into confined spaces or to be easily portable. Guidelines or standards are generally accepted rules and ones that are in use for the purpose of this literature search.

OBJECTIVE

The purpose of this report was to review literature of the past 8 to 10 years in both commercial and military applications. The display technologies reviewed include cathode ray tube (CRT), light emitting diode (LED), liquid crystal display (LCD), electroluminescent (EL), and plasma displays. In addition to compiling research results, this literature review attempts to identify areas where further research is needed to answer human performance questions.

METHOD

The literature search was conducted with an emphasis on locating and obtaining a representative sample of pertinent documentation. Many academic and professional journals and magazines were reviewed. Computer-aided listing services were queried to cover the technical report literature. To ensure a review of the most current literature possible, human factors consultants, industry representatives, and authors of relevant research (Granaas, McKay, Lanham, Hurt, and Juola, at the University of Kansas, and Willinges and Willinges, at Virginia Polytechnic Institute and State University) were contacted for input. Using this methodology, more than 100 primary sources were identified for inclusion in this review.

DISCUSSION

Display Size

A review of the literature reveals that small displays of various technologies have been employed in limited interaction capacities for 10 to 20 years. (Adam, 1983; Myers, 1984; National Technical Information Service (NTIS), 1982; Snyder, 1980). To date however, there are at least 69 portable computers on the market (Wszola, 1983). Some contain smaller screens than the standard 24-line by 80-character screen and some run programs comparable to those available for desktop microcomputers. Specifically, of the 69 products available, 29 have CRT, LCD, or EL displays that are smaller than the standard, with actual size ranging from 54 characters by 24 lines down to 20 characters by 1 line. Available software for these portables includes familiar programs like Microsoft BASIC, JRT Pascal, Visicalc, Multiplan, Mailmerge, Spellstar, and Wordstar, as well as other languages and programs designed to produce similar spreadsheet, programming, and word processing results.

Given limited screen size and large, highly interactive programs, what are the human factors concerns? A telephone survey was conducted to focus on areas of specific interest. (Contacted were Cherry Display Products, Sharp Corporation, PCI Incorporated, Industrial Electronics Engineers Incorporated, IBM, and Waveteck). Though not intended to be exhaustive, this survey of screen manufacturers and system integrators indicates that legibility, including size of characters or targets in proportion to size of the screen, ambient lighting, lowercase versus uppercase and lowercase letters, font styles, screen proportions, and segmented versus dot matrix letters are concerns. The literature reviewed also indicates that to date, legibility is one of the primary concerns of the human factors specialists (Department of Defense (DoD), 1981b; Engel & Granda, 1975; Hendricks, Kilduff, Brooks, Marshak & Doyle, 1983; NTIS, 1982; Nuclear Regulatory Commission, 1981; Waern & Rollenhagen, 1983). Seminal research in that area, the studies that established that users can indeed read small screens, was carried out primarily in the early sixties (NTIS, 1982). A few studies of legibility specific to limited screen applications have also been performed over the past 10 years. Those discussed below represent a sampling of this type of research.

Ellis (1978), for example, compares segmented numerics, often used with small screen technologies, to conventional numerals, finding that certain segmented numerals are predictably difficult to read, take longer to read correctly, and that users can be trained to read segmented numerics. Users lose the effects of their training when they do not practice.

A study by Payne (1983) finds that backlighting is a problem when reading LCDs, especially when viewed at an angle. The author concludes that where legibility under low ambient light levels is required, LCDs should not be used, or the user should be able to control backlighting.

Angles and multiple images on LEDs are the subject of research done by Riley (1977). Riley finds that when the user or display is moved so that the display cannot be fixated during the movement, refresh of the display diodes must be increased relative to the velocity of movement.

System designers also appear to be concerned with other human performance questions but indicate that controlled experimentation has not been a part of the development process. Popular magazine articles raise some questions about the practicality of small screens for certain applications but provide few answers and no human performance data. Wszola (1983), for example, touches on a few aspects of current small screen display technologies that have human factors implications. He notes that CRTs are still the most popular of displays, and that their disadvantages (bulky shape and high power consumption) are outweighed by advantages (they produce light and can be used under poor lighting conditions, can display alphanumerics and graphics, and are already supported by sophisticated software). LCDs, on the other hand, have slower response times, poor graphics, and can stop functioning completely in subfreezing temperatures. However, LCDs are small, lightweight, require little power, and are "relatively immune" to damage. The largest LCD now available is 8 lines by 80 characters, and Wszola (1983) expresses some hesitation about size:

> Whether such a display is large enough for effective work is a matter of personal taste... Portable computers with a 1-line display are minimally usable. You can work with all of these displays, but they are only a fraction of the size of a standard CRT display of 80 characters by 24 lines. You should carefully consider the display size in terms of your particular application and choose the portable that best fits your needs. Word processing on a 4-line display is awkward at best. (p. 44)

Others voice the same concern. In discussing the Gavilan portable, Zepecki (1983) says that the "8-line character display restricted our ability to process lengthy files" (p. 81). Of the Hewlett-Packard's portable, which has a 32-character LCD, Morgan (1982) comments, "I wouldn't want to use the HP-75C for serious word processing. But then, it wasn't designed for that purpose" (p. 8). Convergent Technologies designed its Workslate spreadsheet with the help of 50 to 100 potential users. That system has a 16 x 46 character LCD. Robertson (1983) suggests that "writers will probably decide against this machine because the keyboard and software were obviously not designed with them in mind" (p. 53). Apparently limited experience rather than performance data support these statements.

Nevertheless, the comments seem to indicate that reviewers and system designers alike feel there are trade-offs to be made with small screens. A tendency to believe that word processing should be available with portable systems--the fact that most systems have text entry or manipulation software at some level of sophistication would support this assumption (Wszola, 1983)--is tempered by the notion that these systems, screens, and occasionally keyboards, are too small to support such software in any but a minimal manner.

Where does the trade-off between performance and screen size occur? This is a question that has not yet been answered. The work of Darnell and Neal (1983) is promising. Experimenting with editing of partial and full page displays, these researchers find that productivity is the same for partial and full pages for "typical manuscript editing." Duchnicky and Kolers (1983), who work with scrolled text, concluded that while line length, character density and window height all affect the reading rate for scrolled text, people can read and comprehend text in windows as small as 15 characters by one line. Elkerton and Williges (1984) add to research in this area by investigating search strategies in respect to user experience, file type, file length, target type, and window size (1, 7, 13, or 19 lines). They find that a one-line window generally inhibits file search, and this may indicate a loss of file context. The authors see evidence mounting against one-line windows in retrieval systems and interactive editors, and suggest research be directed toward more accurate identification of the optimal window.

Rosinski, Chiesi, and Debons (1980) investigate the performance of typists entering text, computer programs or numeric data where 0, 1, 2, 3, 5, 9, 15, 26, 46 and 79 characters can be seen. This study finds that visual feedback has no ultimate effect on input performance, but it does affect the number of corrections made by the typists, a point that might be considered in system modeling.

However, the conclusion of Barfield, West, Robertson, Taylor, and Tamplin (1983) indicates that characters on the screen should be increased and scrolling decreased for maximum performance. In addition, Foster and Bruce (1982), who experimented with the Viewdata system in England, cite a 1978 study by Reynolds, Spencer, and Glaze, and a 1980 study by Sutherland to support their statement that a "text frame should include 70-100 words."

Granaas, McKay, Lanham, Hurt, and Juola (1984) look at the question of text presentation on a small screen, comparing rapid, serial visual presentation (RSVP), and leading, both of which are alternatives to the scrolling, paging, or windowing most often used on standard CRTs. Comprehension of text presented using the RSVP or paging methods was found to be superior to the leading method.

Protocols

Protocols, are defined as interactive techniques used to perform a particular operation. They are the steps the user takes to "do" something, e.g., type control-k-y to delete a block, or press function key 2 to undo. While the definition is rather straightforward, this survey uncovered only a few studies that deal specifically with this topic.

Both popular magazines and scholarly journals contain articles discussing particular systems and their relative "user-friendliness," as well as articles that provide broad overviews of the user interface (Cooper, Marston, Durrett, & Stimmel, 1982; Dean, 1982; Dean, 1983; Edwards, 1983; Fong, Collica, & Marron, 1975; Good, 1982; Hammond, Jorgensen, MacLean, Barnard, & Long, 1983; Heckel 1983; Jayaraman, Lee, & Konopasek, 1982; Johnson, 1983; Lemmons & Robertson, 1983; Marcus, 1982; Roberts, 1983; Simpson, 1982; Smith, Irby, Kimball, & Verplank, 1982; Yavelberg, 1982). These articles usually present, as general recommendations, the lessons learned from the design and testing of a particular machine, intuitive or experientially-based comments, or both. There is occasional discussion of keystroke level operations, but human performance data are a rare inclusion. These articles frequently include suggestions on language options pertinent to the vocabulary section of this review.

Research and opinions that speak to the subject call attention to the relationships among all aspects of the human-computer interface and the fact that this interface is not easily dissected into component parts. In addition, some literature would support the conclusion that keystroke analyses have little meaning apart from the user and the task. Cooper, Marstan, Durrett, and Stimmel (1982) and Smith, Irby, Kimball, and Verplank (1982) hesitate to confront the question of operations techniques without first defining the task. Referring to the IBM personal computer's reset procedure of holding down the Control and Alternate keys while pressing the Delete key, Cooper, Marstan, Durrett, and Stimmel (1982) comment, "Without considering a specific context we cannot answer these questions [whether this is a 'good' technique], but we can consider the kind of criteria a human-factors specialist would use in answering them for some particular context" (p. 70).

However, throughout the literature cited above, there also appears to be a general consensus on the primitive operations a "generic" system requires, if not the exact technique to perform those operations. Rutkowski (1982), writing of the Human Applications Standard Computer Interface (HASCI) he designed, appears to be in the forefront of those incorporating these tacit "standards" into design. He moves a number of operations formerly controlled by commands or multi-key operations to special keys. HASCI also places "the most desirable" functions on dedicated function keys and advanced functions are accessed by control letters. Additionally, the interface provides the capability to see formatted text rather than symbols on the screen, and menus are organized so that common choices come first while destructive ones come last.

Primitives for text editors are also discussed. Finseth (1982) is very specific in an article on the capabilities that a text editor should possess. Among those of interest are: at least a 24 x 80 character screen, commands that don't require hands to leave the keyboard, mnemonically arranged commands, commands that match the material being edited, use of control characters for commands, and easily modified defaults. Jong (1982) also makes suggestions for a user-oriented text editor. The author comments on a full range of editing features including cursor movement and cursor movement commands, key/command relationships, text editing commands, and function/control keys. Jong (1982) states that function keys should not be used for general purpose editing. Miller and Thomas (1977) also review preferred capabilities of a well-designed text editor. Among their concerns are computer program editors, problems with symbolic editing, line-oriented representation, and multiple targets.

A number of what might be considered higher level operations techniques are also studied. Poller and Garter (1983) compare the EMACS and VI text editor in respect to mode. Data on speed and quality of editing and user opinions were collected. The authors conclude that features of mode and modeless editors should be combined in the ideal editor. They also propose that basic cursor move and delete control commands be added to the insert mode. Windowing versus scrolling is the topic of Bury, Boyle, Evey, and Neal (1982), who find that novice users preferred windowing and review a number of questions relevant to that choice. Light pens, light guns, and the keyboard are compared in Goodwin (1975). Light pens and guns were found superior to the keyboard in respect to positioning time. Further analyses regarding several tasks are made.

Modeling of the human-computer interface also appears to be a significant direction in current research. This area of study emphasizes a top-down approach and frequently considers one or more variables in the user/task/system relationship. Embley, Lan, Leinbaugh, and Nagy (1978), for instance, recommend a mathematical procedure for comparing program editors with respect to time. Methods for selecting editing tasks and command sets, recording performance, and calculating terminal session time are clearly defined. Roberts and Moran (1982) provide a model which considers a greater number of variables, including time, errors made, ease of learning, and range of system functionality.

Card (1979) also ties protocol to task and user in his analysis and comparisons of performance times. In introducing the Command Language Grammar, Moran likewise (1981) speaks to the need for top-down, as opposed to keystroke-level up, system design based on integrated user/task/system modeling. In The Psychology of Human-Computer Interaction, Card, Moran, and Newell (1983) complete and synthesize these concepts.

Foley (1982) has developed a task-dependent model of interaction techniques. This model is graphics oriented, but basic elements can be transferred to a text editing environment.

Another approach to text editing analysis is the collection and review of large quantities of data on actual editing practices. Among those espousing this method are Embley and Nagy (1982), who include in their effort a look at suboptimal performance, and Davis (1983a, 1983b), who couples data analysis with application of Moran's (1981) Command Language Grammar. Tyler, Roth, and Post (1982) concentrate on exploring the discrepancies in novice, intermediate, and expert text editing performance. Free-form text editing behavior is the subject of Hammer and Rouse (1979) who report on initial stages of a project designed to collect editing data for the TECO and SOS systems. A study by Chafin and Martin (1979) has similar objectives.

Small Screen Specific

No discussion of protocols with respect to small displays was found in the literature reviewed, with the exception of a few comments on how specific portable systems handle text manipulation. No human performance data were included in such reviews.

Guidelines or Standards

Thimbleby's article, "Guidelines for 'Manipulative' Text Editing" (1983), is the most thorough review and the only set of guidelines found on this topic. This author breaks text editing into three categories--symbolic editing (editing with keyboard commands), manipulative editing (special keys), and gesture editing (using pointing devices like light pens)--and covers major "manipulative" design issues at the keystroke-specific level, e.g., deleting: off- and on-line methods; inserting: expand and insert.

Williges and Williges (1984) present a rudimentary survey of established human performance guidelines as they relate to data entry procedures, selection of input device, keyboards, special-function keys, cursor control, direct pointing controls, and continuous controls. These guidelines pertain mostly to what has been called above high keystroke level operations. Human Engineering Guidelines for Management Information Systems (Hendricks, Kilduff, Brooks, Marshak, & Doyle, 1983) also presents a cursory look at protocols in Chapter 5, "Working in the File." Applicable information on devices, including the types of devices most suited to a particular task, is contained in the chapter entitled "Keyboard and Input Devices." The treatment of this subject by MIL-STD-1472C (DoD, 1981b) under Section 5.15 "Personnel-Computer Interface" is similar and quite brief.

Engel and Granda (1975) review devices available for data entry in a section on data entry techniques, with emphasis on control on keyboard, fast movement, accurate positioning, freedom to displace from screen, and extendability over a wide range of applications and environments.

Computer Science and Technology: Selection of Data Entry Equipment (Recicar, 1979), a National Bureau of Standards document, provides information to Federal agencies on selection of data entry equipment. The author of the report found no information on the editing or validating features of portable keyed devices available in the sources he reviewed.

Correspondence with the American National Standards Institute indicates that a subcommittee of that organization is defining and classifying text preparation and interchange functions to identify appropriate areas of standardization. The group will look at basic system functions for text preparation, manipulation, reception, preparation for distribution and presentation, as well as other aspects of text processing. The objective of this study and possible standard is to minimize operator error and retraining costs, without placing constraints on extended system capabilities.

Abbreviations

Three articles containing human performance data relating to abbreviations indicated that there are a number of issues with human performance implications to consider in the design of an abbreviation scheme, including but not limited to, abbreviation consistency; the task involved (encoding or decoding); the experience of the user population; the size of the word; multi-word terms; and idiosyncratic words. The place of conventional abbreviations--those supplied by subject-matter experts as exemplified in a number of the guidelines reviewed below--in command languages is also a question raised by the literature discussed here.

Hirsh-Pasek, Nudelman, and Schneider (1982) investigated learnability and encoding and decoding of abbreviations in limited lexicons like those provided by computer command languages. The conclusions they reached, based on results of three experiments, were that simple truncation is the most learnable and encodable abbreviation scheme, phonics or vowel-drop (also called vowel deletion or contraction) methods are preferable for decoding.

Streeter, Ackroff, and Taylor (1983) examined methods of abbreviating words and extrapolated rules based on their observations. Learning and recall of abbreviations formed by the rules are then tested. Weighing a number of different factors, these authors recommend that truncation should be used in most instances though vowel deletion should be used for monosyllabic words and acronym formation should be used with multiple-word terms. However, if decoding is involved, vowel deletion abbreviations are suggested.

Exploring abbreviations in a sonar command and control environment, Rogers and Moeller (1984) began with the assumption that the greater the orthographic and phonological similarity between the word and abbreviation, the easier the abbreviation should be to generate, recall, or interpret. Placing their research in the context of those studies already discussed here, the authors first experimented to see whether the conclusions noted above applied to their environment. The results of that study indicated that for abbreviations matched in length, the conventional abbreviations and simple truncation were roughly equivalent and were both substantially better than the contraction/vowel-out abbreviations.

The second part of the study by Rogers and Moeller (1984) looks at response time for experienced decoding of abbreviations and addresses the effect of multiple versus single abbreviations of the same word. Conventional abbreviations, in this instance, are found to be more accurately decoded than simple truncation when the precise ending of the word must be guessed but not when the root word must be guessed. Other findings indicate that the sound of the initial segment of words has no consistent relationship with the length of decoding response times. On the other hand, visual similarity is as critical as any other factor in decoding abbreviations. This study also finds that experienced users may benefit even more than naive users from rule-based abbreviations, perhaps because the consistencies may be used to help decoding of abbreviations.

The possibility that longer words call for longer abbreviations is also suggested by Rogers and Moeller (1984). The authors, however, conclude that no rule produces the optimal abbreviation for each word, and they support the use of truncation, based on testing to date and rule-knowledge benefits of this technique to experienced users.

Small Screen Specific

Articles which address abbreviation in the context of small display screens were not found in the literature that was reviewed.

Guidelines or Standards

A selective review of abbreviation standards indicates that such standards in most instances rely on conventional abbreviations rather than rule-based techniques for generation of all abbreviations.

The Military Standard on Abbreviations for Use on Drawings, Specifications Standards, and in Technical Documents (DoD, 1981a) appears to be a compilation of abbreviations from Webster's Third New International Dictionary as well as abstracted from documents published by the American National Standards Institute (ANSI), the American Society for Testing and Materials (ASTM), and the Institute of Electrical and Electronics Engineers, Inc. (IEEE). An abbreviation procedure for words not covered in this standard is not provided in MIL-STD-12D (DoD, 1981a) and there appear to be inconsistencies in the techniques used, e.g., the word "insert" is abbreviated "insr," while delete is "del"; automatic bandwidth control is ABWC, while automatic noise limiter is ANL. Therefore, it appears that no broad abbreviation technique generalizations can be made based on this document, particularly without a thorough analysis of the abbreviations included.

MIL-STD-1472C (DoD, 1981b), Human Engineering Design Criteria for Military Systems, Equipment and Facilities, requires that abbreviations conform to MIL-STD-12D, MIL-STD-411D, or MIL-STD-783D. It also states that "if a new abbreviation is required its meaning shall be obvious to the intended reader" (p. 118). New acronyms, if required, shall be developed using the rules of abbreviation in MIL-STD-12D. However, as noted above, rules for generation of new abbreviations are not contained in MIL-STD-12D.

The American National Standard for the Abbreviation of Titles of Periodicals (American National Standards Institute, 1969) includes this general abbreviation rule in its preface:

> The recommended method of abbreviating is by truncation, that is, to drop a continuous group (at least two) of the final letters of the word. Words consisting of a single syllable or of five or fewer letters shall not be abbreviated unless they are frequently used generic words and occur in the Word Abbreviation List (page 8). (The Word Abbreviation List is issued by ANSI's National Clearinghouse for Periodical Title Word Abbreviations.)

The Human Engineering Laboratory's Human Engineering Guidelines for Management Information Systems (Hendricks, Kilduff, Brooks, Marshak, & Doyle, 1983) suggests that approved abbreviations or acronyms and display codes should be used if limited space is a consideration. This document also recommends that systems do not output abbreviations which are not input by the user, presenting the possibility that the user will not have at his or her disposal the means to decode the abbreviation. In addition, abbreviations should only be used if they are more than two letters shorter than the complete word or when most users regularly use abbreviations. Abbreviations should also be unique.

Engel and Granda (1975) also discuss abbreviations in Guidelines for Man/Display Interfaces. Truncation to the first four letters, they conclude, is usually acceptable, but exceptions are needed. They suggest that critical actions should not be dependent upon one keystroke, and consequently confirmation is needed where one-word abbreviations like "Y" and "N" are used. As mentioned in the Human Engineering Laboratory guidelines, abbreviations should not be used for output. Engel and Granda (1975) include a warning against using similar abbreviations in the same entry, which may tend to increase user errors. They also note that the inability to abbreviate/truncate can contribute to user dissatisfaction.

Guidelines for Control Room Design Reviews (Nuclear Regulatory Commission, 1981) suggests that abbreviations should be used whenever possible to minimize operator input requirements. This document also reiterates points mentioned above: If the user inputs an abbreviation, the system should output that same abbreviation; the use of abbreviations or contractions for output text should be avoided; and critical actions should not be dependent on a keystroke.

Ramsey and Atwood (1979), in reviewing pertinent human factors literature, find that the most frequently used abbreviation scheme consists of single-character abbreviations for commonly used commands. In addition, "first-k-character" algorithms are also often used. According to the authors, abbreviated input should be consistent with unabbreviated command usage in order to provide a smooth transition for the naive user.

Williges and Williges (1984) reiterate guidelines mentioned above, with a number of additions. The user should be instructed as to the method used by the system for selecting command abbreviations; abbreviations should be mnemonically meaningful; each word should have only one acceptable abbreviation; abbreviations should be permitted in text entry and expanded later by the computer; and, autocompletion of command names by the computer should not be used.

Vocabulary

Finally, the topic of vocabulary, like protocols, is multifaceted. Perhaps this is because in the truest sense of the word, the term "vocabulary" no longer covers the range of techniques available to "speak" with a computer. Rather, the topic of vocabulary might be further broken down to include current communications options like command languages, menus, natural languages, icons, query languages, form-filled communication, and perhaps a number of other more unique categories. On the other hand, however, these distinctions seem to become increasingly artificial as technological advances allow more storage of nonapplication programming, and human factors experts for expanding and diversifying system language features to suit a variety of uses.

This diversification of language features, and possibly an increased tendency to see all aspects of the human-computer interface as interrelated, places the subject of "vocabulary" within the context of user interface and human-computer dialogue, where it is discussed with numerous other topics. A number of articles that deal with the broad topic of user interface have been mentioned in the protocols section. Here, articles whose main emphasis encompasses those subcategories of vocabulary proposed above are discussed.

A command language is one of the traditional computer interfaces, and current literature attempts to take a closer look at commands to refine their design. In previewing a number of articles on this subject, Thomas and Schneider (1982) suggest that command languages should be studied to find "good" commands. They also feel, however, that computer systems give students who study behavior a unique opportunity to observe a revolution in human thinking and communication. Some of the ideas presented in their articles include: use the terminology of the user; avoid words with multiple parts of speech; be consistent; use the terms you require from the user in the messages to the user; and, use synonyms in the index to manuals.

Some authors bring to the study of computer commands principles and practices developed in other academic disciplines. For example, Arblaster's (1982) approach to the study of computing languages is based on psycholinguistics. Featural analysis, explored by Rosenberg (1982, 1983) as a theoretical framework for analyzing the suggestiveness of command names and for designing command names, is also borrowed from psycholinguistics. A discussion of these works is beyond the scope of this review.

The preponderance of work in the field of command naming and other aspects of vocabulary is again, however, rooted in behavioral and cognitive psychologies. Carroll's (1982) analysis of naming behavior is intended to outline broad directions for naming research. His analysis of 2,500 file names led him to conclude that names are not meaningless in a nontechnical sense and to suggest tentatively that congruence and hierarchical consistency as principles for naming appear promising. Naming strategies and recall were also the subjects of Scapin (1982), who concluded that a naming method and structuring rule should be applied for command naming.

Among the behavioral issues in the use of interactive systems identified by Miller and Thomas (1977) are command organization, argument formats, and prompting and defaults within and between commands. Bernard, Hammond, Morton, Long, and Clark (1981) investigated command language in respect to positioning of recurrent arguments. Their studies show that positionally consistent systems were more quickly learned. However, naive and experienced users show differences in preferred position of recurrent arguments. These researchers emphasize the consideration of interrelationships in a total system environment.

The differences between naive and experienced users are also noted by others. Scapin (1981) finds that the meanings of computer commands are recalled better by experienced subjects and that experienced subjects can more easily transform command definitions into their own wording. However, there is also competition in the experienced user's memory between command word synonyms. The author suggests teaching commands with operational definitions followed by functional definitions, command standardization and prompting as possible solutions to problems arising from experiential differences. Folley and Williges (1982) also find that naive users have smaller and simpler sets of commands than do experts; in other terms, the expert's model is the naive model set of commands plus more powerful commands. The suggestion which follows is that the resultant user models be used in system development. Carroll (1983) carried out an experiment based on the type of user model suggested by Folley and Williges (1982) and a menu-based system. His results indicate that naive users who are blocked from more sophisticated functions learn more quickly and with less frustration. If the system is heirarchically structured, expert-level functions can be made available when the user is ready.

Artificial command languages and their relationship to natural English is the underlying theme in Black and Moran (1982). Experiments they conducted showed that subject-supplied command names differ significantly from designer-supplied names, but the difference in recall between the two categories is negligible. Subject-generated names were also more general, natural English terms. A further study of command names, nonsense syllables, and icons reveals that learning and recall performance was best with infrequently used, discriminating words. Bernard, Hammond, MacLean, and Morton (1982) also compare general, or what they call generic, terms to specific command names with the thought that generic terms are perhaps more memorable. They find, however, that neither recall nor recognition accuracy was reliably different between generic and specific, though recall is better with specific commands. An examination of individual differences in relationship to cognitive strategies leads the authors to recommend that command naming should be studied within the context of the task to be performed.

Broad suggestions for vocabulary based on the design of the Topological Operations Language are provided in Parthasarethy (1982). Facility and safety are among the main concerns of this system designer. More specific features that lead to that end are included.

A section on command language is also included in Botterill's (1982) description of the System/38 user interface. This author describes the System/38's syntax, command names (verb-object pairs), abbreviations (vowel deletion, three letters total), keyword and value naming, the Command Creation function, validity checking, and parameter defaulting and prompting.

Collecting data on command usage was the object of a project reported by Kraut, Hanson, and Farber (1983). Part of their data is 11,000 commands issued over a period of 2 weeks from 16 subjects using the UNIX operating system. An interesting finding is that of 400 available commands, 20 commands accounted for about 70 percent of the usage. Commands were used in this order of frequency: generic editing commands that shape text and other objects were the most frequent (36 percent); orienting commands which give a variety of status information (21 percent); task-specific commands (3 percent); and mail or news exchange utilities (3 percent). Users made errors on 10 percent of the commands they issued. Error patterns indicate that users were more likely to make an error when reexecuting a command, on programming commands, on commands that rely on their knowledge of status and orienting information, and on commands that are inconsistent with the modal command syntax. Suggestions for interface design are extrapolated from this information.

Reed (1982) claims that humans use error-correcting rather than error-preventing strategies, and consequently suggests the appropriate reorganization of programming languages to accommodate this tendency. Using concepts developed in the realm of artificial intelligence, the author presents a framework to design an error-correcting system. This concept is perhaps also applicable to all types of system languages.

Readability was the subject of Roemer and Chapanis (1982). Based on an experiment where subjects used a tutorial for the IBM 3277, general suggestions regarding language are made. According to the authors, computer dialogues should use the simplest language, the user population's reading ability should be considered in system design, and well-designed, engineered dialogues can improve user attitudes toward computers.

Query languages are sometimes discussed as separate entities; however, in many instances design suggestions can apply to all language varieties. Ehrenreich (1981) reviews literature on query languages, and makes recommendations on data organization, quantifiers, language options, feedback of the query, abbreviations, and dialogue transactions, as well as on guidelines more specific to natural and formal query languages. A comparison of natural and structured query languages is made in Small and Weldon (1983), with focus on the Structured English Query Language (SEQUEL). An experiment, where people played the part of the natural language system to provide optimal sophistication, found no differences in accuracy of queries for the two languages, but found some superiority for SEQUEL in other respects.

Menus sometimes take the place of a command language or may be used in combination with other language options. Card (1982) suggests that a greater understanding of information-processing mechanisms, which people use in making menu selections, will lead to optimal menu design. His research finds that of alphabetically, functionally, and randomly arranged menus, initial search is faster for alphabetically arranged. This information is used to extend theories of cognitive psychology. Savage, Habinek, and Barnhart (1982) present a more practical approach to the design, simulation, and evaluation of a menu-driven interface. This article discusses the errors people made with a particular menu-driven system and suggests how those errors might be avoided through menu redesign.

The idea of excursions is also pertinent to the subject of menus. Darlington, Dzida, and Herda (1983) develop the concept of excursions as information-gathering sequences which permit the user to learn about a system. Interfaces, the authors suggest, should be designed hierarchically to allow excursions for naive users while permitting experienced users to take a more direct route.

A number of systems on the market today, including the Apple Lisa and Xerox Star, currently make use of iconography. It may be that systems based on menu-icon combinations are in the offing for other companies as well (Lemmons & Robertson, 1983). The obvious benefit of using icons rather than word menus, command languages, or natural languages is that icons require very little space, either on the disk or on the display screen. Hemenway (1982) mentions an iconic menu with the dimensions 2.5 by 4 cm in her article on this topic. The author gives an overview of icon use, suggesting that icons are more visually distinct than words and are superior to words for representing variation among a set of commands. She also discusses how icons represent command features and how symbolic representations might be evaluated. Smith, Irby, Kimball, & Verplank (1982) show how the design of the Xerox Star has maximized use of a familiar user conceptual model and easily recognized icons. The familiar model of a business office is the core of this system, and menus contain iconic representations of office procedures.

Eisenberg and Hill (1984) predict that natural language will be another aspect of the microcomputer interface in the near future. Whether this is the case, many of the barriers to natural language system use, including unavailability of computer storage space and high costs, may be decreasing as the capabilities of these systems are increasing. Hayes and Reddy (1983) describe in some depth the characteristics of a natural language system necessary for "graceful" interaction. These include robust communication, flexible parsing, domain knowledge, explanation facility, focus mechanisms, identification from descriptions, and generation of descriptions. More narrow in their focus, Furnas, Landauer, Gomez, and Dumais (1983) explore more thoroughly two of Hayes and Reddy's (1983) criteria: identification from descriptions and generation of descriptions. The results, which were intended to add to the research on natural language systems, showed that two people will provide the same name for an object only 15 percent of the time. The authors suggest an adaptive system which can develop a dictionary of synonyms.

A mathematical description of a system which is probably similar to the one Furnas, Landauer, Gomez, and Dumais (1983) suggest, that is, a system which can acquire language from examples, is contained in Swanenberg (1979). Additionally, an unusual perspective is introduced by Thomas (1978). He suggests a game-theoretic model of natural English which he presents as more applicable than an encode-decode model to the design of natural language systems.

It appears that features of natural languages are being incorporated into command languages. For example, DACCL, a command language discussed by Hill (197?), would appear to incorporate a number of concepts from natural language systems. This includes the ability to accept paraphrases and randomly created abbreviations, intelligent defaults, and easy modification of the command language grammar.

A unique system intended as an integrated user interface is the COUSIN interface being developed by Carnegie-Mellon (Hayes & Szekely 1983). Already operating under UNIX and SPICE systems, this language is based on a single interface for all applications, communications using forms, and intelligent help for form fillings.

Small Screen Specific

In the literature reviewed, no discussion of vocabulary in respect to small screens was found.

Guidelines or Standards

The Guidelines for Control Room Design Reviews (Nuclear Regulatory Commission, 1981) dictates that a command language should be based on the operator's point of view, should be logical, consistent, and should reflect the vocabulary and syntax of the expected user population. In addition, keywords should approximate real words, and the language should require an explicit command in order to terminate an interaction. Individual input words which must be typed should not exceed seven characters.

In the subsection addressing language issues, MIL-STD-1472C (DoD, 1981b) makes brief suggestions covering nomenclature, abbreviations and acronyms, standardized field, and feedback. In specific reference to command languages it says that the commands should be written in the language of the user and that command help should be available to the user at any time.

Engel and Granda (1975) devote a section of their document to language. They call for a logical and consistent language and focus on nomenclature; prompting; flexibility (in synonym ability, abbreviation, and command procedures); and intelligent selection of defaults as important language considerations. More specific language features are reviewed in the subsection entitled "Prompting and Structuring." A number of other applicable topics, including implicit prompting, are covered under another subsection entitled "User Energy Techniques."

The Human Engineering Guidelines for Management Information Systems (Hendricks, Kilduff, Brooks, Marshak, & Doyle, 1983) addresses the subject of language in Chapter 4 and lists suggested command words and potential and recommended commands in Appendix C. Information included in this document generally aims at consistency, ease of learning, simplicity, and clarity. These guidelines also recommend that users should be able to create their own input commands and that systems should be able to accept synonyms.

Ramsey and Atwood (1979) state that definitive guidelines in the area of computer languages are currently not supported by research, but reasonable guidelines could be developed. According to the authors, such guidelines would address the following topics: command language structure and complexity; statement syntax; separators and terminators; abbreviations; defaults; command choice; mechanisms for switching to computer initiated form; error messages and recovery; command stacking; and dialogue style.

Computer Science & Technology: Recommendations for Database Management System Standards (FIPS Task Group on Database Management System Standards, 1979) discusses query languages but does not call for standardization of syntax and semantics.

Williges and Williges (1984) include sections on command languages, formal query languages, restricted languages, and menus in their compendium of dialogue design considerations and pertinent references. The command language concerns include organization, nomenclature, defaults, editor orientation, user control, command operation, system lockout, special commands, and other specifics. Menu issues include selection, order of options, selection codes (including icons), layout, and content.

CONCLUSIONS

This literature review resulted in the documentation of existing articles concerning the human performance aspects of small screen displays. Physical display size, protocols, abbreviations, and vocabulary are the subtopics addressed. In conclusion, the literature points to several areas where there are voids in the published research, as well as where there are voids in current research trends.

Research Trends

Overall, the largest research trend appears to be the increasing use of small screen displays in various settings. The maturing of various display technologies has allowed them to be practical in size and able to be used in environmentally constrained settings where they once were impractical. Another of the major contributing factors to this conclusion is the advent of the portable computer. Many portable computers integrate smaller than standard screen displays, and their availability and usage are on the rise.

Countering this trend are technological advancements which may produce lightweight CRTs and larger flat panel displays (Bindra, 1984; Myers, 1984). For the future, it seems reasonable to predict that some interactive applications requiring small screens will continue to exist and may perhaps expand.

A large portion of the general human performance and display research focuses on legibility or psychophysical issues. The topic is beyond the scope of this document, but nonetheless important for consideration when using small screen displays. Frey, Sides, Hunt, and Rouse (1983), Grether and Baker (1972), and Snyder (1980) present a thorough discussion of the issues and Frey, Sides, Hunt, and Rouse (1983) use psychophysical information to make recommendations for screen size, e.g., number of characters per line.

Another definite trend in human performance research on displays evidenced by the literature is the holistic study of the human-computer interface. The topics of protocols and vocabulary, when and where they existed, were often discussed within the human-computer interface context and appear to be difficult to otherwise single out for study. In fact, consensus seems to support protocols and vocabulary being studied within the total system/user/task environment, making them environment dependent.

The research on protocols identified here also indicates a preliminary move toward standardization. This is also indicated, though to a lesser degree, for vocabulary. Literature to date focuses on more general guidelines rather than standards, but the trend for the future appears to be towards standardization.

The literature reviewed on protocols, vocabulary, and abbreviations also highlights a convergence of numerous fields of study. The fields included psycholinguistics and linguistics, behavioral and cognitive psychology, graphic design and information display, reading, and computer sciences, as well as branches of these disciplines. Some studies find their way into traditional human factors journals. References for the primary sources identified here suggest that this sampling is only a subset of the currently available information.

The overall trend in human performance and abbreviation strategies points to truncation as the optimum method. However, there appear to be other interacting factors indicating a need for further research.

Research Voids

Interest in the human performance aspects of small screen displays is a recent phenomenon and fast becoming an active area of research. Because the topic is current, several voids can be identified.

The most striking void is in the lack of specific research on small screens. The literature review did not discover any human performance research specifically related to small-size screen displays and protocols, abbreviations or vocabulary. Further study is generally needed in these areas.

Although a few articles reported on human performance and small screen size, their results were not conclusive. Further work is necessary to determine the appropriate trade-offs; in particular, the performance implications of using small screen displays on portable computers are unknown. Users expect to have the same kinds of applications programs on their portables as they do in their regular office environment. However, some of the research raises questions about that expectation and calls for further experimentation.

Given the above discussion, it is not surprising that specific standards are rare. In general, the standards documents reviewed are more similar to guidelines or high-level "rules-of-thumb" for the designer to follow. Abbreviation, protocol, and vocabulary standards exist, but more research is needed to make them more useful. No specific standards for small screens exist for any of the subtopics, indicating a direction that future experimentation could take.

ANNOTATED REFERENCES

Adam, E.C. (1983, August). Tactical aircraft displays. Information Display Systems Engineering, 1-11.
This paper discusses the application of small-screen, integrated displays and integrated controls to the cockpit environment of the Navy's F/A-1B hornet. The focus of the discussion is technical and hardware oriented with mention of two 5-inch viewable area CRTs. Of merit is the use of these small screens. However, there is no discussion of the number of characters per line, lines per screen, or graphics. The only reported human performance data are included on a chart depicting the "useful screen diameter" as a function of viewing distance and eye resolution limit. It is assumed that other human performance data exist but did not fit within the format of this article's presentation.

American National Standards Institute, Inc. (1969). American National Standard for the abbreviation of titles of periodicals (ANSI Z-39.5-1969), New York, N.Y.: Author.
This document is a listing of conventional abbreviations for titles of periodicals. It is included here as a representative publication on abbreviations. Truncation is recommended for generating new abbreviations.

Arblaster, A. (1982). Human factors in the design and use of computing languages. International Journal of Man-Machine Studies, 17(2), 211-224.
The authors provide a psycholinguistics-based overview of factors which make the structures of programs more obvious. Implications for programming languages, as well as other interface languages, are outlined.

Barfield, W., West, D., Robertson, M., Taylor, F.L., & Tamplin, N. (1983). Stress as a function of the rate at which information is processed on a video display terminal. In A. T. Pope & L. D. Haugh (Eds.), Proceedings of the Human Factors Society 27th Annual Meeting (pp. 516-520). Santa Monica, CA: Human Factors Society.
This experiment examined the relationship between stress and performance by varying rates of information and density levels (4 or 7 characters) on a VDT. Scrolling was used to exemplify the diverse conditions of information-overload and information-underload, while subjects searched for a visual target or targets. In general, data suggest that performance will decrease as the presentation rate on a VDT surpasses the user's information-processing abilities. Also, high density of characters on a VDT screen increases errors in identifying information on the screen. Among other things, the authors conclude that it is unproductive to fill a VDT screen with information that is not related to the decision making task. They also propose that to satisfy conditions of optimum density and speed, characters on the screen be increased while scrolling rates should be decreased.

Bernard, P.J., Hammond, N.V., MacLean, A., & Morton, J. (1982). Learning and remembering interactive commands in a text-editing task. Behaviour and Information Technology, 1(4), 347-358.
The authors theorize that generic names may have advantages over specific command names, because they may be easier to generate from memory, are used in varying contexts, and may be transported to a computer context more readily, and because they do not convey a precise meaning. While focusing on generic versus specific naming, their study includes an examination of learning and memory where initial training occurred under different task demands, and monitoring of differences between individuals within the user sample. The results indicate, among other findings, that neither recall nor recognition accuracy showed a reliable difference between generic and specific names. However, recall of command operations was better with specific commands. Also, data on individual differences suggest that differing demands imposed by experimental conditions may modulate a user's inclination to adopt particular cognitive strategies. These results indicate that command naming should be studied within the context of task demands.

Bernard, P.J., Hammond, N.V., Morton, J., Long, J.B., & Clark, I.A. (1981). Consistency and compatibility in human-computer dialogue. International Journal of Man-Machine Studies, 15(1), 87-134.
Three factors affecting the order of arguments in a command string were investigated: the consistent positioning of a recurrent argument; the relationship between argument entry order and order in natural language; and the relationship between argument entry order and the position of argument values on a VDU. Software specialists and naive users were tested. Software specialists preferred positionally consistent arguments where recurrent arguments were the first argument in the command string. Naive users showed preferences for structures similar to natural language (verb-objects), where the object is the first argument. Users also showed a small tendency to prefer expressions with recurrent argument as the second argument. Positionally consistent systems were more quickly learned. Since specialists' and naive users' preferences varied, the authors emphasize the importance of considering interrelationships in a total system environment.

Bindra, A.K. (1984, March). Flat panels are getting bigger, brighter, and better. Electronics, pp. 113-123.
Competition for the cathode-ray tube--electroluminescent, plasma-gas discharge and vacuum-fluorescent displays, and other new technologies--is the focus of this article. The author predicts that flat-panel displays will dominate CRTs where portability, power, and compactness are important factors, and he discusses some of those currently on the market. He also predicts that CRTs will dominate the field for color. No human factors issues are discussed, however, and the article is most valuable for its mention of the companies developing or marketing small displays.

Black, J.B., & Moran, T.P. (1982). Learning and remembering command names. In T. McDonald (Ed.), Proceedings of the Human Factors in Computer Systems Conference (pp. 8-11). Washington, DC: National Bureau of Standards.

Black and Moran attempt to begin determining how natural language affects the user's behavior with artificial command languages. A set of studies of command naming, learning, and memory showed that subject-supplied names for editing operations differed significantly from designers' names. However, memory results showed no difference between subject and designer-supplied names. Subject-generated words were words one frequently used in written English and were also general terms. The authors proposed that the underlying factor here is the discriminability of the name. A further study of command names, nonsense syllables and icons focused on frequency, discriminability and word/nonword, and whether or not these characteristics affect learning and memory. Results show: performance was best with infrequent, discriminating words, best on initial learning and on later choice; discriminability had no effect on free recall, also had no effect on nonwords to begin with; words were always better than nonwords in free recall, but better in learning and cued choice only when discriminating; and a word frequency also had an effect, with infrequent works performing better on all tasks. Methodological lessons learned are also presented.

Botterill, J.H. (1982). The design rational of the System/38 user interface. IBM Systems Journal, 21(4), 384-423.

This report discusses how current human factors research influenced one system design. Of particular interest are System/38's object orientation and its relationship to verb-object, hierarchical commands (like Create Document, Copy File), and menus; trade-offs required here when choosing an abbreviation scheme; the Create Command, which allows the user to create commands to invoke his/her own programs; a built in validity-checking feature that can report errors in keyword names, values, value types and length and interparameter value conflicts; displayed defaults; and an optional parameter prompter available when keying rather than executing commands. These innovations are based on research described in other articles mentioned here.

Bury, K.F., Boyle, J.M., Evey, R.J., & Neal, A.S. (1982). Windowing vs. scrolling on a visual display terminal. Human Factors, 24(4), 385-394.

The research presented tries to answer these questions: 1) Do novice users have a natural bent to window or scroll? 2) Is performance in one of these modes more efficient? 3) Does training in the conceptualization of either mode improve performance? and 4) Does the use of keytop graphic "scroll" figures affect performance? The authors summarize the results in this manner: 1) when allowed to self-define the system, a significantly greater number of novice users chose the window mode; 2) overall, subjects in the window groups performed significantly faster and with significantly fewer moves than did subjects in scroll groups; 3) an explanation and demonstration of the appropriate concept did not have significant effect on performance; and 4) the users of keytop graphic scroll figures did not perform significantly better.

Card, S. (1979). A method for calculating performance times for users of interactive computing systems. Proceedings of the International Conference on Cybernetics and Society (IEEE Transactions, CHI1424), 653-658.
An explanation of and mathematical equations for predicting interface performance with respect to time is presented in this article. Card shows how to break a task into its component parts, predict time necessary to perform those components, and subsequently predict time required for the entire task. By varying the method used to perform a task and making comparisons, the designer can find the "best" way to perform an operation, and decide on appropriate trade-offs. The author also shows how speed of the typist can be taken into consideration and how to compute benchmarks.

Card, S. (1982). User perceptual mechanisms in the search of computer command names. In T. McDonald (Ed.), Proceedings of the Human Factors in Computer Systems Conference (pp. 190-196). Washington, DC: National Bureau of Standards.
The author asks, "In what order should menu items be arranged," in an attempt to discover the information-processing mechanisms people use in selecting commands from a menu. He suggests that knowledge of these mechanisms, rather than numerous experiments pitting one design against another, will lead to better menus and, therefore, to better user performance. The study finds that of alphabetically, functionally, and randomly arranged menus, initial search is faster for alphabetically arranged, and that difficulty in visually locating the target item accounts for differences in search times. With practice, however, all arrangements are equivalent. Card relates this research to studies of memory units (chunks), and purports that the user learns locations of menu items by "building perceptual chunks." Furthermore, he suggests that the strongest perceptual organizers of chunks appear to be 1) between an item and the next one following and 2) between an item and other item in the same box (for menus divided into small boxes).

Card, S., Moran, T., & Newell, A. (1983). The psychology of human-computer interaction. Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.
This publication is a synthesis and refinement of previous work on modeling of human-machine interaction, and particularly text-editing performance. A set of goals, operators, and methods as well as a set of selection rules for choosing among competing methods make up the GOMS model. The book contains a thorough discussion of this model, mention of other models, mathematical equations for use in evaluation, and previously-established typing statistics.

Carroll, J.M. (1982). Learning, using and designing filenames and command paradigms. Behaviour and Information Technology, 1(4), 327-346.
Carroll looks at naming behavior in general and command naming specifically. He contends that "the very form of names implies specific properties their referents must have." With this paradigmatic nature of naming in mind, he analyzed 2,500 filenames created by fellow employees at IBM and used the results to structure several experiments. The conclusions include: names are not meaningless in a nontechnical sense; however, the question of which aspects of paradigmatic structure are psychologically significant remains open; congruence and hierarchical consistency as principles of command naming look promising. The author feels these principles can be easily applied to naming, they are constructive, and they are not task, user, or system specific. Broader questions are: 1) Could people use a file retrieval aid? (yes); 2) How much hierarchy in names is enough? (six levels unlimited with reservations); 3) Could people use a filenaming aid? (yes); 4) Should command customization be unstructured? (no)

Carroll, J.M. (1983). Presentation and form in user-interface architecture. Byte, 8(12), 113-122.
"Architectural presentation" is the specific nature of interface elements--the objects and actions that make up the interface and its concepts. "Architectural form" is how the system functions are interrelated in typical user scenarios. The author states that this distinction "has important implications for the organization of research efforts on user-interface issues and in particular for the development of user-interface guidelines." Presentation, he claims, cannot be studied except case by case and general rules cannot be developed. Guidelines may be generated, however, in respect to form. Arguments and examples are included.

Chafin, R.L., & Martin, T.H. (1979). A man-computer interface study for command and control computer systems. Proceedings of the International Conference on Cybernetics and Society (IEEE Transactions, CHI1474), 1, 21-25.
Preliminary results of a project developed by the NASA Jet Propulsion Lab to help the designer better understand operator attributes and identify system characteristics which influence man-computer performance are presented in this article. A conceptual system model was designed, based on human factors literature. User groups suggested the following guidelines for commands: 1) standardize commands; 2) make commands no more than three characters; 3) make arguments no more than four words; 4) make delimiters easy to type; and 5) frequently used commands should be made simple, and less frequently used ones more descriptive. Experiments planned will look at performance time, errors, directionality of control, complexity, and documentation style, along with other aspects of the user interface.

Cooper, R.G., Jr., Marston, P.T., Durrett, J., & Stimmel, T. (1982). A human-factors case study based on the IBM personal computer. Byte, 7 (4), 56-72.
This article gives some general human factors guidelines: 1) trade-offs are often necessary, 2) multiple sets of criteria are often necessary, and 3) know your user. A few comments on keyboards and how to use them to achieve certain goals are included.

Darlington, J., Dzida, W., & Herda, S. (1983). The role of excursions in interactive systems. International Journal of Man-Machine Studies, 18(2), 101-112.
This paper proposes that interactive systems be designed to permit two forms of dialogue, direct way and excursion. Using cognitive psychological theory, it is shown that excursions--information-gathering sequences of operations that enable the user to learn the commands which change his data sets--can prevent interactive deadlock or unsolvable incongruence, situations where inexperienced users do not know the command necessary to proceed and do not have at their disposal, within the system itself, the means to find that command. Consequently, computer dialogue should be designed to satisfy the needs of both novice and experienced users.

Darnell, M.J., & Neal, A.S. (1983). Text editing performance with partial and full page displays. In A.T. Pope & L.D. Haugh (Eds.), Proceedings of the Human Factors Society 27th Annual Meeting (pp. 821-825). Santa Monica, CA: Human Factors Society.
The results of a study where skilled typists edited 60 lines of 80 characters or 20 lines of 80 characters show that use of a full page display is advantageous in respect to locating time. However, there was no difference in displays in revising time, editing time, and editing errors. The authors concluded that "use of a partial page display resulted in the same productivity as a full page display for typical manuscript editing."

Davis, R. (1983a). Task analysis and user errors: A methodology for assessing interactions. International Journal of Man-Machine Studies, 19(6), 561-574.
This study uses Moran's Command Language Grammar (CLG) to describe users interacting with a statistics package, SPSS. Goals were to: 1) produce a theoretical analysis of the SPSS command structure; 2) superimpose observational data on the different levels of CLG to uncover common errors and their cause; and, 3) integrate theoretical and empirical approaches, and assess their utility. The author collected data from nine subjects, and the results uncovered problems such as mode errors, inconsistencies between subsystems and inefficient use of facilities, problems with use of the macro-facility by the casual user (suggesting the use of function keys instead), a large percentage (20 percent) of command misspellings, and inappropriate separators between commands and arguments. The author concludes that detailed error analysis can becoupled with formalized task analysis and "one can be used to predict the other generating many testable recommendations."

Davis, R. (1983b). User error or computer error? Observations on a statistics package. International Journal of Man-Machine Studies, 19(4), 359-376.
Davis provides further discussion of the system mentioned above to gain information concerning the complexity of statistical analyses attempted and their ease of implementation and to gain more insight into prevention or correction of errors. A limited number of guidelines on command names, (are they meaningful, flexible, functional) argument names, automatic error correction, documentation, and error messages specific to this system are presented.

Dean, M. (1982). How a computer should talk to people. IBM Systems Journal, 21(4), 424-454.
This article contains thorough guidelines for system message design and writing, including how to: 1) set human goals for messages; 2) apply psychology in writing messages; 3) write messages that accommodate intended users and their situations; 4) playact to evaluate the messages for usability before coding; 5) edit the messages for appropriate language; 6) design the computer program or system to produce the message; and 7) test the messages along with the running code. The article explains when to use messages and what they should say.

Dean, M. (1983). Simplify, simplify, simplify. Byte, 8(12), 161-172.
The basis of this article is informed personal opinion about the development of a chart-based database system. In general, the author suggests throwing out anything that does not enhance the user's understanding of a task. He claims "open-ended, command-driven software is dead."

Department of Defense. (1981a). Military Standard on Abbreviations for use on drawings, specifications standards, and in technical documents (MIL-STD-12D). Washington, DC: U.S. Government Printing Office.
This document is a compendium of conventional abbreviations. No rules for generation of abbreviations are included, and no rules can be extrapolated from the abbreviations therein without a thorough analysis.

Department of Defense. (1981b). Military Standard on Human engineering design criteria for military systems, equipment and facilities (MIL-STD-1472C). Washington, DC: U.S. Government Printing Office.
MIL-STD-1472C is a compilation of human performance guidelines to 1981. Sections on visual displays (5.2 - contains primarily psychophysical guides), and the personnel-computer interface (5.15) are included. Language, input/control, function keys, and abbreviations, are topics that are briefly discussed.

Duchnicky, R.L., & Kolers, P.A. (1983). Readability of text scrolled on visual display terminals as a function of window size. Human Factors, 25(6), 683-692.
This study contains a look at scrolling and size of a window through which text scrolls, as determined by 1) the length of the lines of text within the window, 2) the number of characters making up each line (density), and 3) the number of lines displayed (window height). Subjects read text of varying line lengths and density and 1, 2, 3, 4, or 20 lines high, setting the scroll rate as they wished. They then answered questions about the text. The results showed that line length, character density, and window height all significantly affected the rate at which the scrolled text was read. However, in specific reference to small displays, the authors note that "people can successfully read and comprehend continuous text presented in windows as small as one line of 15 characters, although they do not do as well as when reading from larger windows." Other specific recommendations are included in discussion of the experiments.

Edwards, S. (1983). Why is software so hard to use. Byte, 8(12), 127-138.
Edwards believes that easy-to-use software 1) gets a job done with no fuss, 2) does what you expect it to do, and 3) offers the user less information to become confused by. He suggests that the software which meets these standards uses good models of things with which the user is already familiar and is consistent. The author looks to standard program conventions, which he feels will not be implemented, and integrated software to help provide such features.

Ehrenreich, S.L. (1981). Query languages: Design recommendations derived from the human factors literature. Human Factors, 23(6), 709-725.
This article contains a review of pertinent literature and a compilation of guidelines. General recommendations include: 1) the organization of the database presented should match the organization perceived to be natural by the users; 2) minimize the use of quantificational terms, since people have great difficulty in using quantifiers unambiguously; 3) experimentally test major query language features prior to adopting them; 4) have the computer rephrase a user's query and display it for acceptance prior to execution; 5) use the method of simple truncation, e.g., deleting all but the word's first three to five letters, in forming abbreviations; and 6) the system's messages to the user should provide prompts or reminders of the current state of transaction development. Guidelines for formal query languages include: 1) the features of query langauge should be partitioned into groups or layers; 2) avoid use of such operations as "or more" and "or less;" and 3) for inexperienced users, the use of global terms is not recommended unless the specific terms of information subsumed under the global terms are retrieved together frequently. The article also makes the following recommendations concerning natural query languages, 1) systems should be capable of carrying out a "clarification dialogue", and 2) quasi-natural languages should be considered when it is impossible to teach a formal query language or when it is not feasible to develop a natural query language.

Eisenberg, J., & Hill, J. (1984). Using natural-language systems on personal computers. Byte, 9(1), 226-238.
The focus of this article is what Artificial Intelligence (AI) is bringing to the PC, including general capabilities and directions. The authors see natural language as just another part of an integrated interface, and suggest that it will soon be available on microcomputers.

Elkerton, J., & Williges, R. (1984). Information retrieval strategies in a file-search environment. Human Factors, 26(2), 171-184.
Elkerton and Williges study search strategies with respect to experience, file type, file length, and target type. The editor variable is window size--1, 7, 13, or 19 lines. Results showed large performance differences between one-line windows and other sizes. The authors comment, "A one-line window generally inhibited file-search performance." These researchers see evidence (taking into consideration Duchnicky & Kolers, 1983) that one-line windows should be avoided. They also encourage research which would "incorporate a response surface design" to determine optimal window size.

Ellis, N.C. (1978). A comparative study of seven segment numerics. Human Factors, 20(6), 655-660.
This article reports a comparison of segmented numerics and conventional numerals. Two studies provided the following conclusions: 1) reading difficulty is idiosyncratic to certain segmented numerics, and the type of numbers erroneously reported is predictable; 2) reducing viewing time reduced the readability of segmented numerics when compared to conventional numerals (with training this effect may be overcome); and, 3) without the opportunity for continued exposure to segmented numerics acquired reading skills are reduced significantly after 1 month.

Embley, D.W., Lan, M.T., Leinbaugh, D.W., & Nagy, G. (1978). A procedure for predicting program editor performance from the user's point of view. International Journal of Man-Machine Studies, 10(6), 639-650.
An explanation of a procedure to compare program editors with respect to terminal session time, that is, time necessary to perform program editing tasks, is discussed in this article. The authors describe a mathematical model which utilizes user think time, user typing rate, and computer response time. They also suggest procedures for: 1) selecting the editing tasks to be accomplished; 2) selecting the command sets of interest; 3) recording performance of the specified editing tasks and determining the number of command and keystrokes associated with each task; and, 4) calculating terminal session times for the various command sets using appropriate values for computer and user response times. A case study of the NUROS and CMS systems is also included, and suggestions for an enhanced model are made.

Embley, D.W., & Nagy, G. (1982). Can we expect to improve text editing performance In T. McDonald (Ed.), Proceedings of Human Factors in Computer Systems Conference (pp. 152-156). Washington, DC: National Bureau of Standards.
This article contributes an initial report on a large scale computer collection of editing data to: 1) determine the principal components of task time distribution; 2) study suboptimal performance; 3) predict editing task duration; 4) validate models of editing performance; 5) develop realistic models of editing activities that incorporate aspects neglected in previous studies, such as file manipulation and error correction; 6) compare editor quality; 7) improve guidelines for editor design; 8) provide data to facilitate editor standardization; 9) provide quantitative information to assist in training, learning, documentation, and help; and, 10) lay quantitative foundations for the study of cognitive processes involved in editing activities. While this study has produced very little measurable data to date, it is the type of research which should ultimately produce specific recommendations on the "best" operations techniques and "optimal" performance.

Engel, S., & Granda, R. (1975). Guidelines for man/display interfaces (IBM Technical Report 00.2720). Poughkeepsie, NY: IBM Poughkeepsie Laboratory.
This document is a compilation of established guidelines to 1975. Engel and Granda cover display formats, frame content, command language, recovery procedures, user entry techniques, principles, and response time. The authors do not cover any small-screen specific human performance criteria.

Finseth, C.A. (1982). Managing words: What capabilities should you have with a text editor? Byte, 7(4), 302-310.
The author's suggestions resulted in the EMACS system. He recommends that a text editor should: 1) be a screen editor; 2) provide a backup copy; 3) handle very large files well; 4) have multiple buffers and windows; 5) have good response time; 6) have at least 24 x 80 character screen--if not larger; 7) have a detachable keyboard; 8) have commands which don't require that your hands leave the basic keyboard; 9) have mnemonically arranged commands (easier to learn, more evenly spaced around the keyboard); 10) have commands which match the material being edited (words, sentences, paragraphs); 11) use control characters for commands; 12) have a way to recover from deletions; 13) have the ability to save and pick up quickly (called "state save"); and, 14) have easily modified defaults.

FIPS Task Group on Database Management System Standards. (1979). *Computer science & technology: Recommendations for database management system standards* (NBS Special Publication 500-51). Washington, DC: U.S. Department of Commerce, National Bureau of Standards.
This pamphlet contains a discussion of database management system standards, including why they might be considered, if they should be considered now, and specific recommendations. Topics covered include terminology, data description languages, data manipulation languages, data dictionary/directory facility, and query language/end user facilities. The task group's suggestion regarding query language/end user facilities is that "standardization of syntax and semantics of end-user facilities is not required. Such facilities are easily learned, problems and subject-matter dependent. . ." However, they called for broad, long range guidelines or conformance with ANSI guidelines in a number of other areas.

Foley, J. (1982). Human-computer dialogue: Interaction tasks and techniques--a survey and categorization. In P. Van Balen, & K. Moe (Eds.), *Proceedings of the Human Factors Considerations in System Design Symposium* (NASA Conference Publication 2246). Greenbelt, MD: National Aeronautics and Space Administration.
This report is a synopsis of a lengthier report on interaction techniques, particularly as applicable to graphics. The assumption here is that interaction sequences can be decomposed into a series of basic interaction tasks made up of these types: selection, position, orientation, path, quantify, and text. Foley discusses the further breakdown and analyses of these types.

Folley, L.J., & Williges, R.C. (1982). User models of text editing command languages. In T. McDonald (Ed.), *Proceedings of the Human Factors in Computer Systems Conference* (pp. 326-331). Washington, DC: National Bureau of Standards.
These study results suggest that expert and naive text editors have different models of the command language they use. Naive users have a smaller and simpler set of commands than experts. Experts have the same basic set but add more powerful commands; their experience has been the basis for development of more sophisticated strategies. The results argue for basing design of interactive editors on the user model.

Fong, E., Collica, J., & Marron, B. (1975). *Six data base management systems: Feature analysis and user experience* (NBS Technical Note 887). Washington, DC: U.S. Department of Commerce, National Bureau of Standards.
The authors review six database management systems, the trade-offs involved in the use of each, and present recommendations for federal government users. The following features were emphasized: 1) numerical data vs. textual data orientation; 2) retrieval vs. update orientation; 3) self-contained vs. host-language based; 4) ad-hoc query vs. predetermined transaction processing; 5) predefined process via user vs. procedural programming language; 6) network vs. nonrelational); and 7) large public vs. small "private."

Foster, J.J., & Bruce, M. (1982). Reading upper and lower case on Viewdata. Applied Ergonomics, 13(2), 145-149.
Viewdata is a system provided by the British Telcom that allows users to obtain information on their television screens. This study investigates the formatting of information of Viewdata frames to discover how alternative formats influence reader performance. Applicable here is the statement, "It has been recommended that a text frame should include 70-100 words."

Frey, D.R., Sides, W.H., Jr., Hunt, R.M., & Rouse, W.B. (1983). Computer-generated display system guide (Draft). Palo Alto, CA: Electric Power Research Institute.
This report contains psychophysical information for displays and recommends using this information to develop baselines for determining acceptable number of characters per line and spacing.

Furnas, G.W., Landauer, T.K., Gomez, L.M., Dumais, S.T. (1983). Statistical semantics: Analysis of the potential performance of key-word information systems. The Bell System Technical Journal, 62, 1753-1807.
The authors of this study asked people to give descriptions of various information objects, and analyzed their responses to determine how well the objects to which they refer can be inferred from what they say. The goals were to: 1) advance understanding of the psychological processes by which human semantic reference is generated; and, 2) model and estimate the strengths and weaknesses of information systems that take human-generated descriptions of sought items as their input. Results show that two people--the designer and the user--will provide the same name for an object only about 15 percent of the time, and that several methods of attempting to find "good" names improve chances, but not dramatically. Use of statistical data on reference behavior can improve the choice of a single name only by approximately a factor of two over the common procedure. The authors ultimately suggest an adaptive system that keeps track of the frequency with which objects are sought and the names used to seek them (successfully or unsuccessfully) for statistical analyses. This would, in turn, be incorporated into the system to increase successful searches. This approach focuses on what the user brings to the interaction.

Good, M. (1982). An ease of use evaluation of an integrated document processing system. In T. McDonald (Ed.), Proceedings of the Human Factors in Computer Systems Conference (pp. 142-147). Washington, DC: National Bureau of Standards.
Design guidelines for the Etude text processing system included: 1) commands in English-like phrases (verb-modifier-object form); 2) commands that are reversible; 3) a help key to provide online assistance at any time; and 4) feedback to the user for each keystroke. This article summarizes the results of an experimental evaluation of the system, where Etude was found to be easy to learn and use based on a study of learning time, time to create and edit letters, user anxiety and attitude. The exact relationship of the study to the specific guidelines above is vague.

Heckel, P. (1983). Walt Disney and user oriented software. Byte, 8(12), 143-150.

Disney's successful cartoon ideas are extrapolated and applied to software in this article. Heckel says: 1) make it interesting; 2) exaggerate reality; 3) think in visual terms; 4) prepare the audience; 5) don't crowd the screen; 6) involve the audience; 7) make a "best guess" of what the audience's experience will be; and, 8) try again.

Hemenway, K. (1982). Psychological issues in the use of icons in command menus. In T. McDonald (Ed.), Proceedings of the Human Factors in Computer Systems Conference (pp. 20-23). Washington, DC: National Bureau of Standards.

Hemenway discusses the use of graphic symbols (icons) in computer command menus. She suggests that icons are used because they are more visually distinct than words, require little space (an example was given of a menu which was displayed in a 2.5 by 4 cm area), and are superior to words for representing variation among a set of commands. However, their effect on user performance is currently unknown. The author speculates on the nature of icons and their use, including how command features are represented with icons, and how the structure command sets are represented. She then makes suggestions on evaluation of representations. The focus of the evaluation discussion is on what may influence 1) a user's discovery of what the icon depicts and 2) a user's "linking" of what is depicted with the command. It is suggested that representing command set structure with icon set structure may affect these two processes, allow the user to organize the command set in his or her mind, and consequently learn and remember more about each command.

Hendricks, D.E., Kilduff, P.W., Brooks, P., Marshak, R., & Doyle, B. (1983). Human engineering guidelines for management information systems. Alexandria, VA: U.S. Army Material and Readiness Command.

This document covers broad, and occasionally specific, considerations in system design. Of particular interest are the discussions of modeling, language (including suggested command names and alternatives), working in a file, and keyboard and input devices (with device advantages and disadvantages).

Hill, K.L. (1979). A user-engineered command language for research in thermonuclear fusion. Proceedings of the International Conference on Cybernetics and Society (IEEE Transactions, CH11424), 1, 27-32.

Hill describes a command language (DACCL) which incorporates "naturalness" and usability of the language in design. DACCL allows more or less English-like paraphrases and randomly-created abbreviations, provides intelligent default decisions, and permits easy modification of any part of the grammar. The language can also run on a nondedicated minicomputer. No test results are included.

Hirsh-Pasek, K., Nudelman, S., & Schneider, M.L. (1982). An experimental evaluation of abbreviation schemes in limited lexicons. Behaviour and Information Technology, 1(4), 359-369.
The results of three experiments which investigated learnability, encoding, and decoding of abbreviations in limited lexicons like computer command languages are provided in this article. The experiments were designed to provide data more closely related to a computer environment than previous abbreviation studies. Results indicated that simple truncation is the most learnable and encodable scheme, while minimum letters to distinguish the word is least useful. Phonics or vowel-drop techniques, however, allow easier decoding. It was also found that user-defined abbreviation schemes do not provide improved performance in learning, encoding, or decoding.

Jayaraman, S., Lee, M.J., & Konopasek, M. (1982). Human-computer interface considerations in the design of personal computer software. In T. McDonald (Ed.), Proceedings of the Human Factors in Computer Systems Conference (pp. 58-62). Washington, DC: National Bureau of Standards.
This article is a high level review of the Visicalc program in respect to information presentation, language and dialogue design, human-computer task allocation, and intelligence of interfaces. The authors describe features that, in their view, make this spreadsheet program an intelligent and friendly interface and suggest an additional Question Answering System (QAS) interface for use in mathematical modeling.

Johnson, W.T. (1983). Developing a truly portable Visicalc. Byte, 8(9), 66-76.
This article includes information on the implementation of Visicalc on a portable computer with one line of 32 characters. Included are specific techniques for viewing a spreadsheet on a small screen, as well as the keyword language which supports this program.

Jong, S. (1982). Designing a text editor? The user comes first. Byte, 7(4) 284-300.
Some general text editing features that contribute to ease of use are reviewed, based on the author's experience and on software design and human factors literature. Jong covers many features including: status information, modes, scrolling, blocking, multifile editing, audio signaling, color, paragraph formatting, cursor movement and cursor movement commands, function/control keys (should not use function keys for general purpose editing), prompts, key/command relationships, "idiot-proofing", text editing commands, string searches, user messages, and multilingual editors. The EMACS text editor is used as an example, and trade-offs are discussed.

Kraut, R.E., Hanson, S.J., & Farber, J.M. (1983). Command use and interface design. In A. Janda (Ed.), Proceedings of the CHI '83 Human Factors in Computer Systems Conference (pp. 120-124). New York, NY: ACM.
These researchers study the UNIX operating system in the thought that examination of people's natural use of an already existing system will provide a richer source of information for guiding development. After collecting data on 24,000 processes and 11,000 commands, data were analyzed to find the frequency with which commands were used and sequential dependencies among commands. Their findings led to suggestions for design of similar systems: 1) systems should provide functional command organization, a) core commands should be directly available to the user in all command contexts, b) command menus should serve as help facilities but not block other commands, c) users need a help thesaurus to accept a user's query and return semantically related system commands; 2) systems should provide relevant orienting information and feedback; and, 3) provide intelligent error control.

Lemmons, P., & Robertson, B. (1983). Improving the user interface at Digital Research. Byte, 8(7), 286-294.
Comments by Gordon Eubanks, vice-president of Digital Research's Commercial Systems Division, are the subject of this article. Of interest is the fact that this company is working on an iconic user interface project that was once publicized but is now top secret. Eubanks also comments on ease of learning as opposed to ease of use and suggests that the two issues are often confused. Function keys are easier to learn, but harder to use; control keys are harder to learn, easier to use. He feels there is a real need for solid analysis of keyboards and their functions.

Marcus, A. (1982). Typographic design for interfaces of information systems. In T. McDonald (Ed.), Proceedings of the Human Factors in Computer Systems Conference (pp. 26-30). Washington, DC: National Bureau of Standards.
A graphic designer makes suggestions for the presentation of screen information based on the SEEDIS system.

Miller, L.A., & Thomas, J.C., Jr., (1977). Behavioral issues in the use of interactive systems. International Journal of Man-Machine Studies, 9(4), 509-536.
This contained an extensive, though not exhaustive, discussion of important behavioral problems relating to system characteristics (performance, facilities, and on-line information) and interface characteristics (dialogue style, displays and graphics, and other input-output media) with suggestions for solutions. Among many other topics, Miller and Thomas discuss system response time, command languages and command organization, editing (the most important facility provided by computer systems, accounting for 75 percent of all commands used in one study), problems with line-oriented searches, keyboard design and layout, special application keyboards, and alphanumeric displays in the context of research to date. The article also includes a lengthy bibliography.

Moran, T.P. (1981). The Command Language Grammar: A representation for the user interface of interactive computer systems. International Journal of Man-Machine Studies, 15(1), 3-50.
Command Language Grammar (CLG) is a symbolic notation which allows the designer of a system and system interface to express on paper the interactions of man, task, and system. It can also be seen as a formula for constructing a "user model" of the system. This model includes a conceptual component (abstract concepts), a communications component (command language and the conversational dialogue), and a physical component (keyboard, display, etc.). Each of these components is comprised of description levels. Each of the description levels is, in turn, a refinement of the level above. There is a task level, semantic level, syntactic level, and an interaction of Moran's CLG lacks the ability to locate error sites and to define recovery methods.

Morgan, C. (1982). Deus ex machina of the technological age. Byte, 7(11), 6-10.
Morgan discusses the rise of portable computers, using the HP-75C as an example. Technical information dominates, and only a few questions with human performance implications are raised.

Myers, E. (1984, July 15). Flat panel furor. Datamation, pp. 46-48.
Myers discusses flat panel technologies and trends for the future. Interesting facts or comments are: 1) only LCDs and ELs can be battery-run, and ELs only for a few hours; 2) "25 lines are needed" for desktop-comparable operations (Costalano, Standard Resources); and 3) development of large panels is underway. Some manufacturers believe that CRTs will continue to dominate the display market over the next ten years, while others feel that flat panels will continue to find new markets.

National Technical Information Service. (1982). Electroluminescent display devices. (Citations from the NTIS Data Base). Springfield, VA: Author. (NTIS No. PB82-805227)
These abstracts, which refer to military and commercial EL applications, indicate that studies of ELs were begun in the early to mid-sixties and have continued to date. However, the abstracts are mainly technology-oriented. Human performance data is one aspect of legibility.

Nuclear Regulatory Commission. (1981). Guidelines for control room design reviews (NUREG-0700). Washington, DC: U.S. Government Printing Office.
This document includes sections on visual displays, and process computers (with CRT displays). A number of broad guidelines on abbreviations, language, data input, function controls, as well as psychophysical display information are covered.

Parthasarethy, S. (1982). Ergonomic considerations in the design of interactive languages for process control. Applied Ergonomics, 13(2), 129-131.
This article takes a very broad look at high-level language design in reference to TOOL (Topological Operations Language). The author suggests that interactive languages for process control must meet two ergonomic requirements: facility and safety. Among other things, he suggests that a language must be comprehensible to the operator at all times and should enable the operator to describe his or her actions correctly; the syntax should be simple and concise; and there must be a constant two-way flow of information. Additionally, the user should be able to split long instructions into smaller units and to link short ones together, as well as to split a major activity into several minor activities. Key words should have a direct physical meaning and avoid abstractions. Good software allows the user to make corrections in the middle of a process According to Parthasarethy, the user must be allowed to "retain his savoir-faire while he is provided with sophisticated tools."

Payne, S.J. (1983). Readability of liquid crystal displays: A response surface. Human Factors, 25(2), 185-190.
According to this article, backlighting has an adverse effect on reading LCDs, particularly when viewed at an angle, and performance improves as ambient light level increases. Those findings are based on a study of 120 subjects who performed 200 number identifications while reading from a four-digit, seven-segment, reflective liquid crystal display. Where legibility under low ambient light levels is required, the author suggests LCDs should include a control for user manipulation of back light, or that an alternative technology should be used.

Poller, M.F., & Garter, S.K. (1983). A comparative study of moded and modeless text editing by experienced editor users. In A. Janda (Ed.), Proceedings of the CHI '83 Human Factors in Computer Systems Conference (pp. 166-170). New York, NY: ACM.
This article presents the results of a comparison of the VI and EMACS editors, with specific attention to the advantages and disadvantages of "mode." Data collected from experienced users were analyzed on the basis of the how long it took to get final products and how good the final products were. User opinions were also solicited. The results suggested that moded editing may be preferable for fixed editing tasks and modeless for free composition. The authors conclude, however, that the ideal editor would combine features of mode and modeless editors. Specifically, they propose addition of basic cursor moving and delete commands using control character names to the inserting text modes.

Ramsey, H.R., & Atwood, M.E. (1979). Human factors in computer systems: A review of the literature (SAI-79-111-DENO). Englewood, CO: Science Applications, Inc.
This report contains general information on abbreviations and languages, vocabulary, display types, and windowing vs. scrolling.

Recicar, S.A. (1979). Computer science and technology: Selection of data entry equipment (NBS Special Publication 500-55). Washington, DC: U.S. Department of Commerce, National Bureau of Standards.
Information for Federal agencies to use in selecting data entry equipment is provided in this document. This booklet includes physical characteristics, strengths and weaknesses, input and output capabilities, and options. Of portable keyed devices, the author writes, "Typical display of input data is on paper, plasma display, LED display, or CRTs." However, of the "Edit/Validate Capabilities" of these systems, he states, "the edit/validate features of portable keyed devices are either non-existent or were not covered by the sources reviewed." Although Recicar mentions LEDs, LCDs and plasma displays, he covers only standard size CRTs in the section on alphanumeric display terminals.

Reed, A.V. (1982). Error-correcting strategies and human interaction with computer systems. In T. McDonald (Ed.), Proceedings of the Human Factors in Computer Systems Conference (pp. 236-238). Washington, DC: National Bureau of Standards.
Reed suggests that since humans use an error-correcting rather than error-preventing strategy, programming languages, and by extension, other computer languages, should be written accordingly. In terminology borrowed from cognitive sciences (frames, scripts, and templates), the author suggests that systems provide immediate and specific error messages, prompts and correction capabilities.

Riley, T.M. (1977). Multiple images as a function of LEDs viewed during vibration, Human Factors, 19(1), 79-82.
A study of angle and multiple images on LED (light-emitting diode) displays. If either the observer, the display, or both are moved so that the display cannot be fixated during the movement, the rate of refresh of the display diodes must be increased relative to the velocity of the movement.

Roberts, B. (1983). Sunrise systems. Byte, 8(6), 54-70.
This article is a review of Sunrise Systems' portable which has a 40-character by 6 line or 80-character by 3-line LCD. Much of the information presented here is technology oriented. The system is menu based, and assigned function keys remain constant across applications.

Roberts, T.L., & Moran, T.P. (1982). Evaluation of text editors. In T. McDonald (Ed.), Proceedings of the Human Factors in Computer Systems Conference (pp. 136-141). Washington, DC: National Bureau of Standards.
This article presents an objective, thorough, easy-to-use methodology for evaluating computer text editors from the viewpoint of their users. Criteria include time needed to perform a set of operations, errors made, ease of learning, and range of system functionality. Using this methodology to evaluate eight text editors, the authors show that diverse editors can be compared. However, while this methodology may show which editor is preferable, it will not show why, except in very general terms.

Robertson, B. (1983). California hardware. Byte, 8(11), 52-64.
Robertson reviews Convergent Technologies' Workslate, and other new products. Workslate is a spreadsheet system with a 16-line by 46-character LCD. Some text entry capabilities are provided. No performance data are included, though Robertson says that 50 to 100 potential users contributed their ideas to the design.

Roemer, J.M., & Chapanis, A. (1982). Learning performance and attitudes as a function of the reading grade level of a computer-presented tutorial. In T. McDonald (Ed.), Proceedings of the Human Factors in Computer Systems Conference (pp. 239-243). Washington, DC: National Bureau of Standards.
"Readability" of the text of a computer-presented tutorial is the main focus of this article, and the study described here raises questions about the relationships among language level and performance, user reading ability, attitudes toward language level and computers, and variations among persons with technical and non-technical occupations. A study of 54 males resulted in the following suggestions: 1) the simplest language should be used in designing computer dialogues, especially when instructions or procedures are presented; 2) if persons of limited reading ability will be among the users of a system, the design should reflect this fact; and, 3) well-designed, human-engineered dialogues can greatly improve users' attitudes toward computers.

Rogers, S.P. (1983). The integrated mission-planning station: Functional requirements, aviator-computer dialogue, and human engineering design criteria. Santa Barbara, CA: Anacapa Sciences, Inc.
This report details the Integrated Mission-Planning Station (IMPS), a computer-generated topographic display system being developed by the U.S. Army Avionics R&D Activity. IMPS contains a small size CRT display showing 12 lines of text, each with 16 characters. The report discusses, in depth, the human performance requirements used in determining the size of the display. In general, several predetermined physical assumptions were met, e.g., given the predetermined screen size, the active area will be 2.7 inches by 3.6 inches to minimize edge distortions. Specifically, the number of characters per line was in accordance with the predetermined physical requirements and MIL-STD-1472C. However, the report does not contain the results of any performance testing; it is anticipated that extensive simulation and testing will be conducted in the future. Also of interest are the discussions of abbreviations for use on the small-size display, the use of reverse video as a coding mechanism, and the functional protocols used in dialogue with the system. Examples of the latter include a single menu page, called the MENU button, used to initiate all procedures and a LAST PAGE button to return to the previous page in case of an error.

Rogers, S.P. (1984). Human engineering analyses used in the development of a computer-generated map display for Army aviators. Proceedings of the NCGA Computer Graphics '84 Conference (pp. 486-494).

The majority of this paper focuses on human factors/computer graphics issues. Of merit however is a brief discussion of a small size monochrome CRT display used as the control-display unit for a larger system. The CRT displays 12 lines of 16 characters each. The top line is reserved for advisory data and the bottom line is reserved to echo data entry from the keypad. The remaining 10 lines are used for data display, prompts, or as flexible function labels for the adjacent line select keys. The labels reflect the changes in function, dependent upon the transaction in progress. Some protocols for interacting with the display unit are also described. Unfortunately, no performance data are included.

Rogers, W.H., & Moeller, G. (1984). Comparison of abbreviation methods: Measures of preference and decoding performance. Human Factors, 26(1), 49-59.

The authors test the theory that the greater the word-abbreviation similarity, the easier the abbreviations will be to generate, recall, and interpret. Three basic approaches to abbreviation were explored: 1) rule-based (systematic application of one or more rules); 2) conventional (abbreviation by subject matter experts); and 3) stereotypic (abbreviation chosen most often by a large group of people).

Rosenberg, J.K. (1982). Evaluating the suggestiveness of command names. Behaviour and Information Technology, 1(4), 371-400.

Rosenberg points out that many designers do not consider suggestiveness in naming commands. He attempts to give designers a tool to measure how good the names are, tell which names are best and worst, and possibly be able to suggest better names. To do this the author defines "goodness," introduces principles of Tverskian featural analysis, and tests the approach with three experiments. Rosenberg proposes that this method of analysis can be applied to language design when a way is found to do suggestiveness calculations without surveying large groups of people. The article below provides further discussion of this approach.

Rosenberg, J. (1983). A featural approach to command names. In A. Janda (Ed.), Proceedings of the CHI'83 Human Factors In Computing Systems Conference (pp. 116-119). New York, NY: ACM.

Rosenberg suggests that featural analysis, a process borrowed from linguistics and psycholinguistics, can be used to further understand command design. This article explores application of the featural approach to suggestiveness, learning and remembering, congruence and hierarchy, universal commands, names and syntax, and nonwords as names, in the context of earlier research. The conclusion reached is that this type of analysis has the potential to provide a comprehensive theoretical framework for the design of command names. References for background information on featural analysis is provided.

Rosinski, R., Chiesi, H., & Debons, A. (1980). Effects of amount of visual feedback on typing performance. In G. Corrick, E. Hazeltine & R. Durst, Jr. (Eds.), Proceedings of the Human Factors Society 24th Annual Meeting (pp. 195-199). Santa Monica, CA: Human Factors Society.
The authors tested text entry performance (with respect to speed and errors) using 0, 1, 2, 3, 5, 9, 15, 26, 46, or 79 character feedback. Analysis of the results suggested that visual feedback had an effect only on the number of corrections made by the typists. This study may have implications for the comparison of small and large screen editing.

Rutkowski, C. (1982). An introduction to the Human Applications Standard Computer Interface, Part 1: Theory and practice. Byte, 7(10), 291-310.
Abstract ideas and concrete guidelines leading to the development of HASCI, the Human Applications Standard Computer Interface, are discussed in this article. From theory, Rutkowski distilled these rules for computer interfaces: 1) the interface is a means of controlling the tool; 2) the interface must accommodate the needs of both the application and the user; 3) the interface itself must present the information necessary for its user; 4) mastery of the interface may require practice; and, 5) with mastery, the interface must become transparent to the user. Specifically, he suggests: 1) clearly label the controls; 2) provide for transportable operator knowledge; 3) design out technical choices; 4) design in predictability and simplicity; 5) defang the computer--allow people to make errors and correct them; 6) show formatted text on the screen rather than just formatting symbols; and, 7) make a product of consumable quality.

Rutkowski, C. (1982). An introduction to the Human Applications Standard Computer Interface, Part 2: Implementing the HASCI concept. Byte, 7(11), 379-390.
Specific implementations and informal testing of suggestions in the document above are reported in this article. The HASCI keyboard (which includes system controls, file controls, applications controls and typestyle controls, as well as two sets of graphic symbols), display unit layout (a document window, interaction window, and prompt window) and menu strategy are introduced. The author suggests that menus should: 1) appear in the same place on the screen; 2) be designed so that the user may indicate choice by typing the first letter of the first word or positioning the cursor over that letter and pressing the return key, or typing the letter without pressing return; and, 3) be organized so that the most common choices occur first in position and potentially destructive choices last. Many new ideas are presented here, including the undo key which will repair nearly any accidental damage.

Savage, R.E., Habinek, J.K., & Barnhart, T.W. (1982). The design, simulation, and evaluation of a menu driven user interface. In T. McDonald (Ed.), Proceedings of the Human Factors in Computer Systems Conference (pp. 36-40). Washington, DC: National Bureau of Standards.
This article reviews in general terms the process and techniques required to develop and test an interface for a broad spectrum of experienced and inexperienced users. The authors designed a system which was largely self-explanatory, hierarchically structured, usable without frustration by novices and experienced users alike, and consistent in screen design. The interface was tested for a number of categories of user errors, and the resulting data pointed to menus with low probability for correct option selection. On those menus one or two incorrect alternatives had a high probability of selection relative to the correct alternative, or a large variety of incorrect alternatives was selected by the participants rather than the correct alternative. This indicates that the user had no useful information on the correct selection. A redesign of the system, which included breaking up and rewording complex or cluttered menus and renaming menu selections according to classification; and, job classification, resulted in fewer errors and the following suggestions for design considerations: 1) use terminology that the user will understand; 2) name selections according to function rather than the user's job classification; and, 3) use short menus with more hierarchical levels.

Scapin, D.L. (1981). Computer commands in restricted natural language: Some aspects of memory and experience. Human Factors, 23(3), 365-375.
Scapin's research shows that naive and experienced users have different requirements in respect to computer commands. Meanings were recalled better by experienced users. Experienced users can put command definitions into their own words more easily, but they also experience more conflicts among command synonyms. Scapin suggests that command standardization may avoid some problems, as may alternative prompting procedures. The author also suggests teaching commands with operational definitions (What does the user want to do) first, followed by functional definitions (What kinds of commands are available from the system).

Scapin, D.L. (1982). Generation effect, structuring and computer commands. Behaviour and Information Technology, 1(4), 401-410.
This study explores the realm of command names, their efficiency, structure, the recall of commands, the results of providing a structuring rule for naming commands, and the kind of methods involved in naming experiments. In a "realistic" environment, subjects pressed unmarked keys, discovered what each key did, and developed command labels for the functions. Where the commands were already labeled by the experimenter, the subjects evaluated the names according to their preference. Later subjects were asked to recall the labels. A subsequent experiment then imposed a structuring rule and tested recall. Results indicated that subjects performed better with their own computer command language than with an imposed language. However, "the existence of a structuring rule helps recall to a great extent." The author recommends that the naming method and a structuring rule be supplied.

Simpson, H. (1982). A human-factors style guide for program design. Byte, 7(4), 108-132.
This article discusses six general human factors guidelines in relation to data entry, display-screen design, and sequence control. It gives general suggestions based on classic sources, but contains nothing specific in terms of performance aspects.

Small, D., & Weldon, L. (1983). An experimental comparison of natural and structured query langauges. Human Factors, 25(3), 253-263.
The study compared an optimal natural language (humans rather than computers interpreted and answered questions) and a structured query langauge, SEQUEL. No difference in accuracy of queries was found, but response time was shorter for SEQUEL. A discussion of human problems with logic and an associated difficulty in use of query languages is included.

Smith, D.C., Irby, C., Kimball, R., & Verplank, B. (1982). Designing the Star user interface. Byte, 7(4), 242-282.
This article discusses principles for design of the user interface on the Xerox Star. The principles are: 1) familiar user's conceptual model; 2) seeing and pointing versus remembering and typing; 3) what you see is what you get; 4) universal conditions; 5) consistency; 6) simplicity; 7) modeless interactions; and, 8) user tailorability. Also recommended are the use of task analysis and top down, beforehand design of user interface.

Snyder, H.L. (1980). Human visual performance and flat panel display image quality. Arlington, VA: Office of Naval Research, Department of the Navy.
Snyder discusses CRT, flat-panel CRT, LED, EL, plasma, LCD, electrochromic, and electrophoretic technolgies. Current human performance data on each are included, as well as recommended applications. This is the most complete discussion of flat panels seen in this review.

Streeter, L.A., Ackroff, J.M., & Taylor, G.A. (1983). On abbreviating command names. The Bell System Technical Journal, 62(6), 1807-1825.
Given a word or phrase, can one predict how it most likely will be abbreviated? The researchers: 1) examined abbreviations to see how much regularity existed naturally; 2) incorporated the observed regularity into a set of abbreviation rules; and, 3) tested learning and recall. The study shows that people are most likely to abbreviate one-syllable words by deleting word-internal vowels, multiword terms by forming acronyms, and polysyllabic words by truncation. However, much of what individuals do is idiosyncratic, and factors other than human predisposition are considered. e.g., problems with difficult and slow formation of abbreviations when using the vowel deletion method on long words, or the fact that some words need more than four characters to distinguish. After weighing a number of criteria, the authors suggest that truncation should be used, except with monosyllabic words (vowel deletion) and multiple-word terms (acronym formation). However, if the task requires generating full names from abbreviations, vowel deletion abbreviations are suggested

Swanenburg, T.J.B. (1979). Machine acquisition of language from examples. Proceedings of the International Conference on Cybernetics and Society (IEEE Transactions, CHI1424) 1, 17-20.
A mathematical description of a machine which acquires increasingly sophisticated natural language skills is presented here. The author asks, "What is an efficient description of the environment" and defines a system which requires the smallest number of bits to reconstruct a message. This system must use a certain number of memory locations, must create new memory concepts, and must have the ability to interact with its environment. Conclusions include: 1) a system can find optimum or near-optimum descriptions from a very small subset of all possible messages; and, 2) the time required to reach a certain level of organizaton increases on the dimension of the input message.

Thimbleby, H. (1983). Guidelines for 'manipulative' text editing. Behaviour and Information Technology, 2(2), 127-161.
The author breaks text editing into three types: symbolic (enter keyboard commands), manipulative (special keys have immediate effect), and gesture (using pointing devices). In this article Thimbleby explores the primitive functions of manipulative text editing and critically reviews the current options for operations techniques. These five rules are distilled for the designer: 1) what you see is what you get; 2) it can be used with your eyes shut; 3) what you type is what you get; 4) separate implementation from interface; and, 5) provide the same 'low-level' interface throughout.

Thomas, J.C. (1978). A design-interpretation analysis of natural English with applications to man-computer interaction. International Journal of Man-Machine Studies, 10(6), 651-668.
Thomas proposes a game-theoretic rather than encoding-decoding conceptual model for natural language communication. His premise includes the idea that discussing and resolving conflicts are primary to game theory and applicable to human-compucer interaction. The author compares these two approaches and provides suggested implications for natural language design.

Thomas, J., & Schneider, N. (1982). A rose by any other alphanumeric designator would smell as sweet. Behaviour and Information Technology, 1(4), 323-325.
Thomas and Schneider provide brief rationales for studying command naming, in both practical (to produce an ergonomically "good" computer system) and scientific (behavioral scientists have a unique opportunity to observe firsthand a revolution in human thinking and communication) terms, as an overview and introduction to other articles on naming in the same issue of "Behaviour and Information Technology". They offer suggestions and warnings to persons interested in using the results naming research. They also suggest the following five rules deduced from studies performed to date: 1) use the terminology of the user; 2) avoid words with multiple parts of speech; 3) do not introduce mindless inconsistency; 4) use the terms you require from the user in messages to the user; and, 5) use synonyms in the index to manuals.

Tyler, S.W., Roth, S., & Post, T. (1982). The acquisition of text editing skills. In T. McDonald (Ed.), Proceedings of the Human Factors in Computer Systems Conference (pp. 324-325). Washington, DC: National Bureau of Standards.
This study collected protocols of text-editing performance for four groups: extreme novice, novice, intermediate, and experts. The analysis showed some differences between novice and experts. Novices were fixed in their editing methods whereas experts utilized the flexibility of the system. In composing corrections each group performs the task in a lower overall time, moving from extreme novice to expert and making tradeoffs between typing speed and time. Experts demonstrated superior knowledge of separate commands but all groups displayed poor understanding of the underlying abstract concepts of the model. Training should emphasize fixed procedures for common problems and require extensive practice in use of a full range of commands.

Waern, Y., & Rollenhagen, C. (1983). Reading text from visual display units (VDUs). International Journal of Man-Machine Studies, 18(5), 441-465.
Waern and Rollenhagen present a high level review of VDU reading issues with indications as to the direction of completed psychological research, and suggestions for further research. This article is of note because of its omission of screen size as a display parameter (character size, character shape, intercharacter spacing, stability, resolution, luminance, contrast and chromaticity) that affects human vision and consequently may affect many human performance aspects of display units.

Williges, B., & Williges, R. (1984). Dialogue design considerations for interactive computer systems. In R. Muckler (Ed.), Human Factors Review '84 (pp. 167-208). Santa Monica, CA: Human Factors Society.
The authors provide a referenced current compilation of human performance guidelines for dialogue design. General information on choice of dialogue (command language, form-filling, computer inquiry, menus, formal query languages, restricted natural language) abbreviations, and input devices is included.

Wszola, S.T. (1983). How to choose a portable. Byte, 8(9), 34-47.
This article is an overview of the portable and pocket computers currently on the market. The author's discussion is technologically oriented, but comments on available software are included, as is a list of products with small displays.

Yavelberg, I.S. (1982). Human performance engineering considerations for very large computer-based systems: The end user. The Bell System Technical Journal, 61(5), 765-797.
The author makes human performance engineering suggestions based on experience with a system developed for use by Bell System Service Representatives. The article includes general information on identifying the target population, integration of a computer into the work environment, operation characteristics of systems that have an effect on performance, and more specific information on input/output design features. Regarding commands he recommends short, simple, consistent structure emphasizing clarity. General "protocols" information is included, but nothing at the keystroke level or on abbreviations.

Zepecki, F.J. (1983). The Gavilan--a full-function portable computer. Byte, 8(9), 80-90.
The Gavilan displays 8 lines of text, graphic symbols, icons on an LCD, and provides a mouse. Zepecki presents a largely technical review, but discusses briefly a Zoom function to help process large files and the system's capacity to feed 24 x 80 video signals to an external monitor. An explanation of the mouse feature is also included.

SUPPLEMENTARY BIBLIOGRAPHY

Boies, S.J. (1974). User behavior on an interactive computer system. IBM Systems Journal, 1, 1974.

Boulay, B. (1981). The black box inside the glass box: Presenting computing concepts to novices. International Journal of Man-Machine Studies, 14(3), 237-249.

Cuff, R.N. (1980). On casual users. International Journal of Man-Machine Studies, 12(2), 163-187.

Davis, E., & Sweezy, R.W. (1983). Human factors guidelines in computer graphics: A case study. International Journal of Man-Machine Studies, 18(2), 113-133.

Eastman, C.M. (1983). A lexical analysis of keywords in high level programming languages. International Journal of Man-Machines Studies, 19(6), 595-607.

Fitter, M. (1979). Towards more "natural" interactive systems. International Journal of Man-Machine Studies, 11(3), 339-350.

Gaines, B.R. (1981). The technology of interaction--dialogue programming rules. International Journal of Man-Machine Studies, 14(1), 133-150.

Geiselman, R.E., & Samet, M.G. (1982). Notetaking and comprehension for computer-displayed messages: Personalized versus fixed formats. In T. McDonald (Ed.), Proceedings of the Human Factors in Computer Systems Conference (pp. 45-50). Washington, DC: National Bureau of Standards.

Gilfoil, D.M. (1982). Warming up to computers: A study of cognitive and affective interaction over time. In T. McDonald (Ed.), Proceedings of the Human Factors in Computer Systems Conference (pp. 245-250). Washington, DC: National Bureau of Standards.

Granda, R.E., Teitelbaum, R.C., & Dunlap, G.L. (1982). The effect of VDT command line location on data entry behavior. In R. Edwards, & P. Tolin (Eds.), Proceedings of the Human Factors Society 26th Annual Meeting (pp. 621-624). Santa Monica, CA: Human Factors Society.

Hartson, H.R., & Johnson, D.H. (1983). Dialogue management: New concepts in human-computer interface development (CSIE-83-13). Arlington, VA: Office of Naval Research.

Hollnagel, E. (1983). What we do not know about man-machine systems. International Journal of Man-Machine Studies, 18(2),135-143.

Huchingson, R.D., Williams, R.D., Reid, T.G., & Dudek, C.L. (1981). Formatting, message load, sequencing method, and presentation rate for computer-generated displays. Human Factors, 23(5), 551-559.

Jagodzinski, A.P. (1983). A theoretical basis for the representation of on-line computer systems to naive users. International Journal of Man-Machine Studies, 18(3), 215-252.

Johnson, D.H., & Hartson, H.R. (1983). Issues in interaction language specification and representation (CSIE-83-15). Arlington, VA: Office of Naval Research.

Jones, A.U. (1977). A user-oriented data-base retrieval system. IBM Systems Journal, 1, 4-17.

Junge, M.K. (1979). Effect of cursor characteristics on pilots' tracking performance on a cathode-ray tube in dynamic simulation. Ergonomics, 22(12), 1363-1370.

Koppa, R.J. (1985). A study of data entry keyboards: The 4 x 4 keypad (Technical Note 4-85). Aberdeen Proving Ground, MD: U.S. Army Human Engineering Lab.

Landauer, T.K., Dumais, S.T., Gomez, L.M., & Furnas, G.W. (1982). Database systems: Human factors in data access. The Bell System Technical Journal, 61(7), 2487-2509.

Lang, R., Lang, K., & Auld, R. (1981). A longitudinal study of computer-user behavior in a batch environment. International Journal of Man-Machine Studies, 14(3), 251-268.

Lovesey, E.J. (1977). The instrument explosion--a study of aircraft cockpit instruments. Applied Ergonomics, 8(1), 23-30.

Malhotra, A., Thomas, J.C., Carroll, J.M., & Miller, L.A. Cognitive processes in design. International Journal of Man-Machine Studies, 12(2), 119-140.

Morgan, C. (1982). A revolution in your pocket. Byte, 7(4) 6-18.

Newell, A.F., & Brumfit, P.J. (1979). Experiments concerned with reading "newscaster" style displays. International Journal of Man-Machine Studies, 11(3), 287-300.

Nickerson, R.S. (1981). Why interactive computer systems are sometimes not used by people who might benefit from them. International Journal of Man-Machine studies, 15(4), 469-483.

Oberquelle, H., Kupka, I., & Maass, S. (1983). A view of the human-machine communication and co-operation. International Journal of Man-Machine Studies, 19(4), 309-333.

Peterson, L.A. (1985). Engineering practices (EP) study on keypad/ keyboards. Aberdeen Proving Ground, MD: U.S. Army Human Engineering Laboratory.

Rich, E. (1983). Users are individuals: Individualizing user models. International Journal of Man-Machine Studies, 18(3), 199-214.

Rouse, W.B. (1977). Human-computer interaction in multitask situations. IEEE Transactions on Systems, Man, and Cybernetics, SMC-7, 384-391.

Sebrechts, M.M. (1983). Cognitive guidelines for computer-aided instruction: A navigation case study. In A.T. Pope & L.D. Haugh (Eds.), Proceedings of the Human Factors Society 27th Annual Meeting (pp. 642-646). Santa Monica, CA: Human Factors Society.

Spiegler, I. (1983). Modeling man-machine interface in a data base environment. International Journal of Man-Machine Studies, 18(1), 55-70.

Stewart, T.F.M. (1976). Displays and the software interface. Applied Ergonomics, 7(2), 137-146.

Strobhar, D.A., & Sweeney, P.J. (1983). Power plant man-machine modeling with systems dynamics. In A.T. Pope & L.D. Haugh (Eds.), Proceedings of the Human Factors Society 27th Annual Meeting (pp.15-19). Santa Monica, CA: Human Factors Society.

Swanston, M.T., & Walley, C.E. (1984). Factors affecting the speed of acquisition of tabulated information from visual displays. Ergonomics, 27(3), 321-330.

Tainsh, M.A. (1982). On man-computer dialogues with alphanumeric status displays for naval command systems. *Ergonomics*, *25*(8), 683-703.

Tagg, S.K. (1981). The user interface of the data analysis package: Some lines of development. *International Journal of Man-Machine Studies*, *14*(3), 297-316.

Thimbleby, H. (1980). Dialogue determination. *International Journal of Man-Machine Studies*, *13*(3), 295-304.

Ting, T.C., & Badre, A.N. (1976). A dynamic model of man-machine interactions: Design and application with an audiographic learning facility. *International Journal of Man-Machine Studies*, *8*(1), 75-88.

Ward, F.E., Greene, F., & Martin, W. (1983). Color matches and discrimination functions for selected hues. In A.T. Pope & L.D. Haugh (Eds.), *Proceedings of the Human Factors Society 27th Annual Meeting* (pp. 249-253). Santa Monica, CA: Human Factors Society.

Waterman, D.A. (1978). A rule-based approach to knowledge acquisition for man-machine interface programs. *International Journal of Man-Machine Studies*, *10*(6), 693-711.

Woodson, W.E., & Coburn, C. (1980). *Human engineering design criteria and modern control display components and standard parts* (Technical Report RS-CR-80-2). Redstone Arsenal, AL: U.S. Army Missile Command.

Wright, P., & Lickorish, A. (1983). Proofreading texts on screen and paper. *Behaviour and Information Technology*, *2*(3), 227-235.

END
DTIC
9-86