2

HUMAN FACTORS DESIGN GUIDELINES
for the
ARMY TACTICAL COMMAND AND CONTROL SYSTEM (ATCCS)
SOLDIER-MACHINE INTERFACE

Version 2.0

Edited by

L. W. Avery

S. E. Bowser, Ph.D.

AES Action Officer: Capt. S. J. Brown

PNL Task Leader: S. E. Bowser, Ph.D.

PNL Program Manager: W. A. Hesser

May 1992

Prepared by

Pacific Northwest Laboratory
Operated for the U. S. Department of Energy
by Battelle Memorial Institute

This report was prepared for the Commander, U.S. Army Tactical Command and Control System Experimentation Site, Fort Lewis, Washington 98433, in accordance with the provisions of ADEA/AES SOW #300A13442 and under a Related Services Agreement with the U.S. Department of Energy Contract DE-AC06-76RLO 1830. The views, opinions, and findings contained in this document are those of the authors and should not be construed as official Department of the Army position, policy, or decision unless so designated by other official documentation.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*	**2. REPORT DATE** May 1992	**3. REPORT TYPE AND DATES COVERED** Final Report Sep 90-May 92

4. TITLE AND SUBTITLE
Human Factors Design Guidelines for the Army Tactical Command and Control System (ATCCS) Soldier-Machine Interface : Version 2.0.

5. FUNDING NUMBERS
P208015

6. AUTHOR(S)
Avery, Larry W.; Bowser, Samuel E.

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Battelle Pacific Northwest Labs
Richland, WA

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Army Tactical Command and Control System
Experimentation Site
Ft. Lewis, WA 98433-5000

10. SPONSORING/MONITORING AGENCY REPORT NUMBER
AES-R-002

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release: Distribution unlimited

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

These guidelines provide a set of rules to assist the design of the interface between soldiers and computers. Interface is defined as the means by which the user inputs, manipulates, and retrieves data. The purpose of these guidelines is to introduce consistency in the look and feel of the soldier-computer interface across battlefield functional areas and between software revisions. Application of the guidelines should result in an interface that is easier to understand, learn, and use which should in turn lead to reduced training requirements and greater user acceptance. Document includes a glossary and index.

14. SUBJECT TERMS
Descriptors: Acceptability, Army Personnel, Battlefields, Computer Programs, Computers (over)

15. NUMBER OF PAGES
695

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	**18. SECURITY CLASSIFICATION OF THIS PAGE**	**19. SECURITY CLASSIFICATION OF ABSTRACT**	**20. LIMITATION OF ABSTRACT**
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	SAR

Descriptors: (continued) Human Factors Engineering, Instructions, Interfaces, Man Machine Systems, Touch, Training, User Needs, Color Displays, Decision Support Systems, Data Displays, Data Storage Systems, Online Systems, Screens(Displays), Liquid Crystal Display Systems, Stereoscopic Display Systems, Maps

Identifiers: Computer Applications, ATTCS(Army Tactical Command and Control System), Soldier Computer Interface, SCI(Soldier Computer Interface), Query Language, Online Help, Embedded Training, Large Screen Display Systems, Touch Interaction Devices, Windows(Computer Programs), Human Computer Interface, HCI(Human Computer Interface)

Contributing Authors:

L. W. Avery
S. E. Bowser
S. M. Adams
R. V. Badalamente, Ph.D.
D. T. Donohoo
D. A. Gellert
K. D. Hargrove
W. A. Hesser
J. G. Heubach, Ph.D.
P. A. O'Mara, Ph.D.
D. J. Pond, Ph.D.
R. B. Randall
S. E. Reynolds
M. S. Rowley

Acknowledgments

The authors gratefully acknowledge and thank the following people for their significant contributions in preparing this document.

R. L. Broderson
S. L. Elwell
T. E. Fleming
L. D. Hatfield
N. I. Hymer
S. P. Lane
D. R. Minihan
S. F. Savage
A. Schur
M. T. Upton
J. C. Worl

Special thanks to Jane Mosier of The MITRE Corporation for providing Guidelines for Designing User Interface Software (Smith and Mosier 1986) on electronic medium.

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

LIST OF FIGURES

LIST OF FIGURES (Continued)

LIST OF FIGURES (Continued)

LIST OF FIGURES (Continued)

LIST OF FIGURES (Continued)

LIST OF TABLES

1.0 INTRODUCTION

1.1 PURPOSE

The purpose of this document is to provide the military command and control system developer with a set of user-computer interface design guidelines. These guidelines are intended to aid in the design of an understandable, efficient, and standardized human-to-computer interface.

1.2 BACKGROUND

The military is continuously developing computer-based systems to enhance command and control. With the emergence of increasingly complex military technology and the proliferation of computerized tactical systems, the process of information fusion and management for command and control has become much more difficult. To reduce this difficulty and provide better decision-making tools, the developing computer-based command and control systems need to be characterized by high performance and reliability.

Command and control system performance and reliability are products of the performance and reliability of each individual component. These components include not only hardware and software, but any user involved in operation, maintenance, or utilization of the output from the system. Of all these components, the user tends to vary most in performance and is the most difficult to predict. Yet the user can be one of the most important elements for successful completion of the system mission. The key, then, to a successful system is to minimize this variable performance through the design of the interface between the user and the hardware and software.

The human-computer interface (HCI) encompasses any interaction between the user and the system. This includes controls, displays, environmental concerns (e.g., lighting and noise), workspace layout, procedures, and documentation. The design of these elements has a major impact on manpower, personnel selection, training, logistics, safety, and human performance, all elements of concern within military systems. For any computerized system, one of the most critical design elements is the HCI.

The HCI is the human's window into a computerized system. Essentially, it is that portion of the software and hardware that connects the user with the resident applications programs and allows the user to input, manipulate, and retrieve data. As such, it becomes the most important aspect of the system design. To achieve maximum system performance, the HCI must be designed to facilitate the user's ability to perform the required tasks. In essence, the designer must attempt to prevent user errors from occurring (Shneiderman 1987).

To facilitate user performance and reduce errors, three factors must be accommodated during the design process.

- First, the functions and tasks that are to be performed by the system and the operational environment must be known. This allows development of an understanding of the overall system dynamics.
- Second, an analysis of the capabilities and limitations of the system users must be completed. This allows development of an understanding of which tasks are best performed by the human and which are best performed by the hardware and software. In addition, this understanding provides the groundwork for task and interface design to ensure the user can successfully perform the required tasks.

- Finally, a consistent set of rules for designing the interface must be applied. This is where Human Factors Engineering (HFE) HCI design *Guidelines* become important.

Design guidelines for the HCI provide three major benefits to the system developer.

- First, designers have a resource upon which to draw to aid in developing usable display screens and interactive procedures. This is especially important with the rapid pace of knowledge acquisition on human performance and computer systems and with the emergence of the graphical user interface as the dominant architecture for the human-computer interface.

- Second, the guidelines induce a tendency towards consistency of design, which is one of the fundamental principles of human factors engineering design.

- Third, there will be reduced training and personnel selection requirements and a possible reduction in manpower requirements for the system.

The end result of employing HFE HCI design *Guidelines* is a system that is easier to use, learn, understand, and thus more reliable.

The U.S. Army Combined Arms Command (CAC) recognized the advantages that HFE HCI design *Guidelines* would bring to the Army Tactical Command and Control System (ATCCS) program. These advantages included bringing cohesion to the various command and control system interfaces being developed, enhancing the usability of each HCI, and improving consistency of design across military systems. These advantages would reduce training requirements for each system and increase user acceptance. CAC requested the ATCCS Experimentation Site (AES) develop a set of HFE HCI *Guidelines* for the ATCCS. This report presents the second edition of *Guidelines*.

1.3 OBJECTIVE

The objective of this document is to provide a set of overarching guidelines focused on designing the user-computer interface to enhance user performance. The design *Guidelines* are not meant to be specific to any particular application within the command and control family of systems but to address design attributes that may be common to all systems. Moreover, the *Guidelines* are to be a "living document," in that they will be revised over time to reflect a better understanding of human performance and computer systems gained through continuing research.

1.4 APPLICABILITY

Two factors influence the applicability of these *Guidelines* to command and control systems: the software architecture being used and the functional requirements of the command and control system.

1.4.1 Software Architecture

The target systems for these *Guidelines* are based on a software architecture composed of UNIX as the operating system, X-Windows, and Motif. The *Guidelines* have been developed to address design considerations germane to that environment. This does not preclude their use for other types of environments, such as Open Look. The *Guidelines* are generic enough to be applicable to almost any graphical user interface and, to a lesser extent, text-based interfaces. The command and control system developer needs to be aware that using a software architecture other than X-Windows and Motif will limit portability to and reusability by other systems.

1.4.2 Functional Requirements

Each command and control system developer must determine applicability of the *Guidelines* depending on their system's functional requirements. These functional requirements are driven, in many ways, by command and control tasks being supported by the system. Two basic types of tasks are performed by most command and control systems: information management and tactical display and control. Most systems are a combination of both, with one task being dominant. In information management, the system is primarily a database and communications device. Information is drawn out of the database for display during the development of courses of action, with subsequent dissemination of orders through electronic file transfer. Typically, the user does not need to respond immediately to the information displayed by the system.

In tactical display and control systems, the system performs real-time or near real-time command and control tasks, with constant update of the current situation display by the computer system. Users often must make immediate assessments and decisions from the data displayed. Some database functions are performed, but primary actions are to monitor the current tactical situation and control friendly assets on a timely basis.

1.5 TARGET AUDIENCE

The target audience for this document includes military personnel representing program management and system managers and command and control user-computer interface designers. Ideally, these individuals should be knowledgeable of the characteristics of the intended user population and the tasks they must perform. In addition, the users of these *Guidelines* should have some knowledge of human-performance considerations.

1.6 APPLICATION

As Smith (1986) discusses, HCI design guidelines are generally stated recommendations, typically selected, and possibly modified for a specific system or set of systems. They are based on generally accepted practices or human performance data and provide guidance that will maximize the usability of an HCI. Guidelines differ from standards in that they are more system-specific and do not have the weight of law or contract behind them. Guidelines are used to derive design rules, which provide specifications for a specific system application and require no further interpretation by software designers. Algorithms are then used to implement the design rules in the software.

The guidelines contained herein do not represent anything more than recommendations. They may not necessarily be applied directly to any specific control systems but must be tailored, as necessary. This is especially true for tactical information display and control systems versus tactical information management systems. The requirements for each type of system are very different, the former being designed as a real-time control system, while the latter is more of a database management system. Figure 1.1 illustrates the process for using the *Guidelines*.

The Open Software Foundation is a consortium composed of MIT, DEC, Hewlett-Packard, and others.

A question may arise regarding how these *Guidelines* relate to the Open Software Foundation (OSF)/Motif Style Guide (1990). The OSF/Motif style guide, while a very thoughtful and well prepared set of design guidelines, does not necessarily reflect the special needs of a tactical computer system. The breadth of topic areas is not as great as is needed, nor should tactical systems have the flexibility encouraged by the Open Software Foundation. In addition, the guide obviously does not accommodate the requirements of a tactical information display system not using Motif. In essence, this document extends, enhances, and focuses the information contained in the OSF/Motif Style Guide.

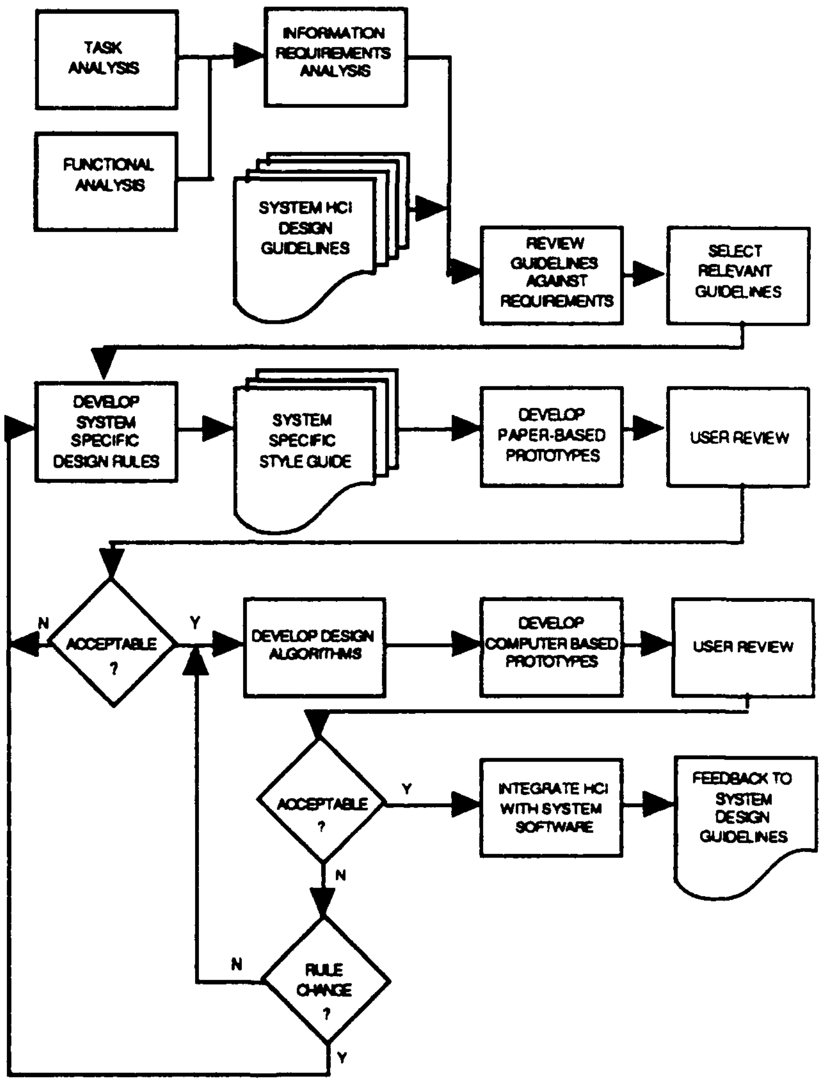


FIGURE 1.1: Process Flow for Utilization of *Guidelines* in the Development of Human-Computer Interfaces

1.7 GUIDELINES ORGANIZATION

The materials presented in the following sections are guidelines. They do not encompass all HCI topic areas relevant to command and control, nor are they necessarily comprehensive within those select areas covered. The authors intend to provide the developer immediate guidance and to open up a dialog to further refine the *Guidelines*. They also encourage comment and feedback on the *Guidelines*.

Topic areas contained in this document include:

- Interactive Control
- Form Filling
- Function Keys
- Menu Design
- Direct Manipulation
- Windowing
- Map and Situation Displays
- Presentation Graphics
- Screen Design
- Color Usage
- Decision Aids
- Query Language
- On-Line Help
- Embedded Training
- Special Displays.

1.7 Introduction - Guidelines Organization

New sections will be added as they are researched and tailored for the development community, when further knowledge is gained of the system designs, and when new research results are available. The reader should be aware that all of the information contained within this document is based on established HCI principles. These principles have, however, been interpreted and tailored to make them as specific as possible to command and control systems.

1.8 RESEARCH NEEDS

From the results of the literature review, a number of areas were identified that require additional research before specific guidelines can be provided. Many of these areas are associated with introducing the graphical user interface and electronic map displays to tactical systems.

A large portion of the research on human performance with computerized systems is more relevant to character-based systems and text-editing tasks. The graphical user interface introduces new methods for visualizing and performing tasks. Some key areas that require research for military applications include:

- metaphor and icon design for tactical systems
- maximum number of windows that should be open for effective human and system performance
- navigation through complex data systems
- color usage for tactical graphical user interfaces, especially foreground and background combinations
- special display considerations.

Electronic map displays also introduce a completely new aspect to tactical computer display design. Some issues that need to be researched regarding tactical map graphics include:

- color coding on a complex, colored background (e.g., an electronic map)
- the degree the electronic map needs to mirror the standard tactical situation map detail and color
- user-prompted versus continuously visible insets.

2.0 INTERACTIVE CONTROL

Interaction between the computer and the user is performed through a two-way communication process: 1) the user inputs commands, and 2) the computer responds to the input. Generally, two interchangeable names are given to this process - sequence control (Smith and Mosier 1986) and interactive control (DoD 1989). For the purposes of these *Guidelines*, the term "interactive control" will be used.

Interactive control of a system occurs through a give-and-take of command and response between the user and the computer, called a "dialog." The following have been identified as the eight major types of user-computer dialogs (Smith and Mosier 1986):

- Question and Answer - The user responds to questions posed by the computer.

- Form Filling - The user enters a series of commands or data items in predefined fields. These fields may be mandatory or optional.

- Menu Selection - The user selects from predefined option lists by pointing with a device, such as a mouse, or keying in associated codes.

- Function Keys - The user controls the dialog by using fixed or variable function keys on the keyboard.

- Command Language - The user makes control entries by composing specified messages for the computer.

- Query Language - The user employs a specialized type of command language to elicit information from a computer system. This is used extensively with databases.

- Natural Language - The user can compose messages to control the computer based on natural, not specialized, languages.

- Graphical Interaction - The user makes selections and controls the computer *interaction* by direct manipulation.

Each of these eight types of dialogs can be used individually or combined as a suite or set of techniques. Open Software Foundation (OSF)/Motif, for example, supports a combination of menu and graphical interaction techniques.

Command and control systems appear to concentrate on using menu, function key, graphical interaction, and form-filling dialogs. Therefore, Section 2 of this *Guidelines* document will deal with these techniques only. Information on query and natural language is presented in Section 13. Additional information on interactive dialog can be found in documents such as Smith and Mosier (1986), Helander (1988), or DoD (1989b).

Eight principles form the basis for designing a good human-computer dialog (Shneiderman 1987; Bailey 1982). These principles are as follows:

- Strive for consistency of design across terminology, menus, command structure, etc. for all applications.
- Enable frequent users to use shortcuts, improving user acceptance and overall system performance.
- Offer informative feedback for all user actions.
- Design dialogs to yield closure. The user will then feel a sense of accomplishment and will know when to go on to the next task.
- Offer simple error handling, both by system error-checking and ease in correcting an identified error.
- Allow easy reversal of actions, such as an UNDO capability.
- Enable the user to feel in control of the interaction with the system.

- Reduce the short-term memory load on the user by means of intuitive displays and interactive sequences, sufficient training, and on-line helps and tutorials.

The design of the user interface should consider tasks performed by the user as the primary criteria. Ease of programming software is a secondary consideration to factors that improve the application user interface.

The primary dialog types used by command and control systems share certain design considerations and guidelines. These are addressed in this section and organized into six topics: general, context definition, transaction selection, interrupts, error management, and alarms.

2.1 GENERAL

The following general guidelines for interactive control apply to command and control systems.

2.1.1 Displayed Context

If the results of a control entry vary depending on a prior action of the user or computer, display a continuous indication of the current context (mode).

2.1.2 Irrelevant Data

The user should have the capability to remove irrelevant items from the display and to reverse this action (i.e., retrieve information that was removed).

2.1.3 Page-Back Capability

When the requested data exceed the capacity of a single display frame, give the user some easy methods of moving back and forth over displayed material by paging or panning/scrolling.

2.1.4 Upper and Lower Case Equivalent

For interpreting user-composed control entries, treat upper and lower case letters as equivalent.

The interface should display to the user, as needed and in immediately usable form, the terminology and commands necessary to perform the task associated with the displayed information.

2.1.5 User-Callable Unfamiliar Term Descriptions

Interface dialog should be written to provide the capability for the user to call up descriptions of unfamiliar terms and commands through context-sensitive HELP screens. See Section 14 for additional information.

2.1.6 User-Paced Sequence Control

Allow the user to pace control entries, rather than having to keep pace with computer processing or external events.

2.1.7 Logical Transaction Sequences

The sequence of transactions (e.g., number and sequence of steps in a task) should be designed from the perspective of what is logical to the user, not what is logical from the perspective of computer processing or ease of programming.

2.1.8 Automated Information Entry

Informational elements required for every communication or transaction should be routinely and automatically included after first input (e.g., call signs and authentication procedures).

2.1.9 Customized Display/Control Options

Allow the user to customize the information displayed on a screen to the particular tactical mission or scenario. For example, the user should have the flexibility to define which files can be displayed concurrently and what tactical data will be utilized in a single display. See Figure 2.1A and 2.1B for examples.

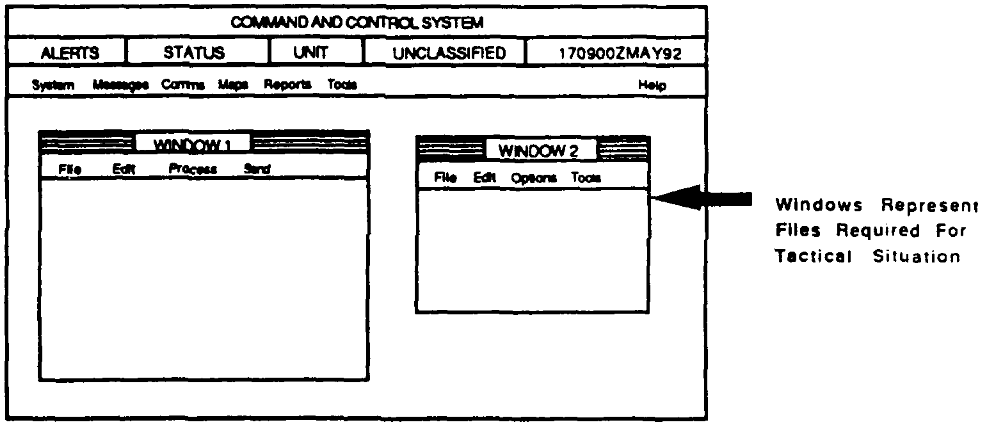


FIGURE 2.1A: Example of How a Screen Display Can Be Customized to Display Required Files

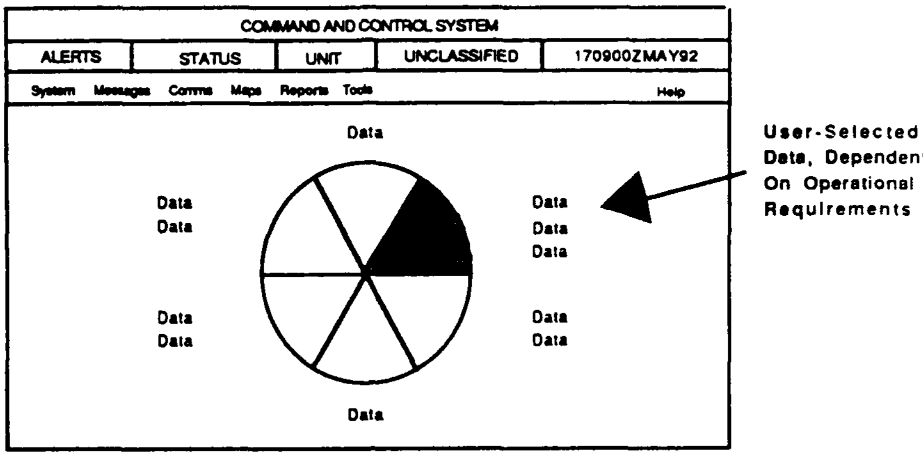


FIGURE 2.1B: Example of How a Screen Display Can Be Customized to Display Specified Information

2.1.10 Distinctive Display of Control Information

Design all displays so features, such as prompts, advisories, etc. relevant to interactive control are distinctive in position and/or format.

2.1.11 System Matched to User Abilities

Command and control systems should be adaptable to individual differences and should accommodate the variety of user abilities, whether novice or expert. For example, accelerator keys for menu selection or command stacking should be available to the expert.

2.1.12 Response Demand

A response should be demanded from the user while instructions on how to respond are still visible on the display.

2.1.13 Consistency

2.1.13.1 Consistent User Control Actions

Ensure interactive control actions are consistent in form and consequence. Employ similar means to achieve similar ends, from one transaction to the next and from one task to another throughout the command and control system.

2.1.13.2 Compatibility with User Expectations

Ensure the results of any control entry are compatible with user expectations, so a change in the state or value of a controlled element is displayed in an expected or natural form. For example, NEXT PAGE should call up the next page of the active file, not an unrelated file.

2.1.13.3 Congruent Names for Control Functions

When selecting names for interactive control functions, choose names that are semantically congruent with natural usage, especially for paired opposites. For example, to move a cursor up, use UP. For the opposite command, use DOWN, not LOWER.

2.1.14 Control

2.1.14.1 Explicit User Control

Allow the user to complete a control entry or action through an explicit action, such as ENTER, before the system interrupts to indicate a computer-recognized word.

2.1.14.2 Minimal User Actions

Ensure control actions are simple, particularly for real-time tasks such as fire control that require rapid user response. Control logic should permit completion of a task with the minimum number of actions, consistent with user abilities.

2.1.14.3 Control Availability

Allow the user to make control entries as needed, in essence, stacking commands.

System response time, in this context, refers to the time between keystroke and screen response. It does not refer to response time for a query of a database.

2.1.14.4 Delay

A sequence of control entries should not be slowed by delays in computer response. In general, system response time should be in the range of 5-50 milliseconds and no longer than 0.2 seconds.

2.1.15 Feedback

2.1.15.1 Periodic Feedback

Ensure the system provides periodic feedback that indicates normal operation is occurring if the user waits more than 15 seconds for the computer to respond.

2.1.15.2 Feedback for Control Entries

Ensure that the computer acknowledges every control entry immediately; for every action by the user, some reaction from the system should be apparent.

2.1.15.3 Indicate Completion of Processing

When computer processing in response to a control entry is lengthy, give the user overt and positive indication of when the processing has been completed.

2.1.15.4 Feedback During Data Entry

Provide displayed feedback for all user actions; display keyed entries stroke by stroke.

2.1.16 Lockout

2.1.16.1 Response-Time-Induced Keyboard Lockout

If application processing time requires a delay of concurrent user inputs and no keyboard buffer is available, the keyboard should be locked out until the computer is ready to accept the next input.

2.1.16.2 Keyboard Restoration Indication

When keyboard lockout has been terminated, provide a clear indication to the user.

2.1.16.3 Interrupt to End Control Lockout

In situations where control lockout occurs, provide the user with a means of aborting the transaction that caused the lockout. A method such as a special function key can accomplish this transaction. The system should not reset and lose previous processing when aborted.

2.1.17 Response Time

2.1.17.1 Appropriate Computer Response Time - Dialog

Ensure that the speed of computer response to user entries is appropriate to the type of dialog. Responses to menu selections, function keys, and most entries during graphic interaction should be immediate.

2.1.17.2 Appropriate Computer Response Time - Perceptual

Ensure that the speed of computer response to user control entries is appropriate to the transaction involved. Generally, those transactions perceived by a user to be simple should have faster responses.

2.1.18 Cursor Design

2.1.18.1 Distinctive Cursor

A contrast ratio of 3:1 is the minimum recommended for an office environment.

Indicate the current cursor position by displaying some distinctive cursor symbol at that point. In all cases, try to obtain the highest contrast possible between the cursor and the background. Cursor size should be such that the cursor is not lost in the "clutter" of the background.

2.1.18.2 Easy Cursor Positioning

Provide the user with an easy, accurate means of pointing a displayed cursor at different display elements and/or display locations. The cursor positioning should work consistently throughout the application.

2.1.18.3 Confirming Cursor Position

For most graphics data entry, pointing should be a dual action, first positioning a cursor at a desired position, then confirming that position to the computer.

2.2 CONTEXT DEFINITION

2.2.1 Application-Provided Context Definition to the User

Design the interactive control of the application such that the user maintains an understanding of the context for the task being performed. The system should prompt expected user actions. For example, results of previous steps in the task affecting the present step and current options should be displayed.

2.2.2 Context Established by Prior Entries

Design the interactive control software to interpret current control actions in the context of previous entries; do not require the user to re-enter data. Prompt the next logical action by the user.

2.2.3 Record of Prior Entries

Allow the user to request a summary of the results of prior entries (i.e., a history file) to help determine present status.

2.2.4 Display Operational Mode

When context for a user task is defined by the operational mode, display to the user the current mode and any other pertinent information.

2.2.5 Consistent Display of Context Information

Ensure information displayed to provide context for interactive control is distinctive in location and format and consistently displayed frcm one transaction to the next throughout all related applications.

2.2.6 Highlighting Selected Data

When a user is performing an operation on some selected display item, highlight that item.

2.2.7 Display Control Parameters

Allow the user to review any active control parameter(s).

2.3 TRANSACTION SELECTION

2.3.1 Consistent CONTINUE Option

At any step in a defined transaction sequence, if there is only a single appropriate next step, provide a consistent control option, such as ENTER, to continue to the next transaction.

2.3.2 Indicating Control Defaults

When control is accomplished by keyed command or option code entries, and a default is defined as a null control entry, indicate that default to the user. See Figure 2.2.

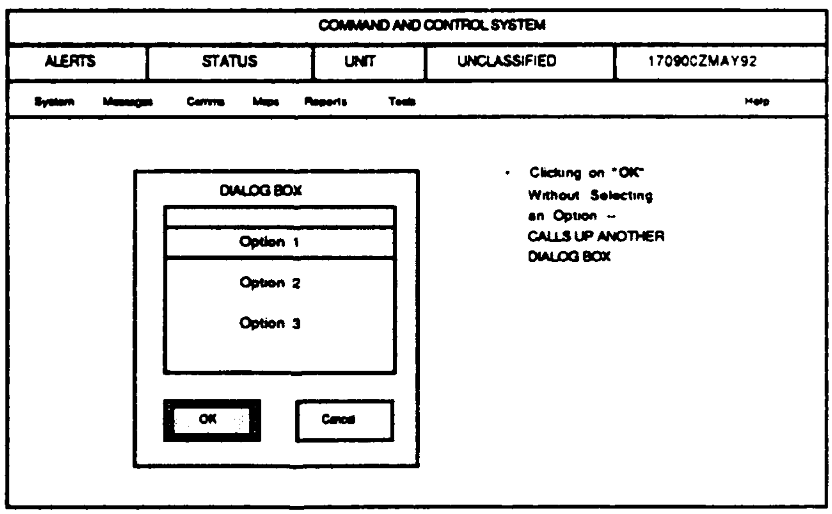


FIGURE 2.2: Example of How a System Can Display Default Status

2.3.3 User-Specified Transaction Timing

When appropriate to task requirements, allow the user to specify transaction timing. For example, the user should be able to specify when a requested transaction should start, when the transaction should be completed, or the periodic scheduling of repeated transactions.

2.3.4 Display Option Codes

When the user must select options by code entry, display the code associated with each option in a consistent, distinctive manner.

2.3.5 Available Options

Ensure the user can clearly distinguish between available and unavailable options. Displaying unavailable options in a visually distinctive manner may aid navigation.

2.3.6 Stacked Control Entries

Stacked control entries, also called stacked commands, allow the user to input a series of command entries at one time. This can be done by continuous entry while the computer processes previous commands, or by typing in a series of commands and then entering them simultaneously.

Allow the user to key a sequence of commands or option codes as a single stacked control entry.

2.3.6.1 Abbreviation in Entry Stacking

For control entry stacking, accept command names or their abbreviations or option codes, just as if those control entries had been made separately.

2.3.6.2 User Definition of Macro Computer Commands

Provide flexibility in transaction selection by allowing the user to assign a single name to a defined series of control entries. Use that named macro for subsequent command entry. All applications should include a predefined informational query process (see Section 13).

2.3.6.3 Consistent Order in Entry Stacking

For control entry stacking, require entries be made in the order they would normally be made when performing a succession of separate control entry actions.

2.3.7 Cursor Placement

2.3.7.1 Cursor Placement for Keyed Entry of Options

When the user must select options by keyed entry of a corresponding code, place the *cursor in the control entry area* at display generation.

2.3.7.2 Cursor Placement for Pointing at Options

When the user will need to select among displayed options by pointing, place the cursor on the first (most likely) option at display generation.

2.3.8 Prompting Control Entries

Provide the user with whatever information may be needed to guide control entries. Incorporate prompts in a display at any point in a transaction sequence, and/or provide prompts in response to requests for HELP.

2.3.9 General List of Control Options

Provide a general list of basic control options that will always be available to serve as a home base or consistent starting point for control entries. For an example, see Figure 2.3 below (see also Paragraph 5.3.3).

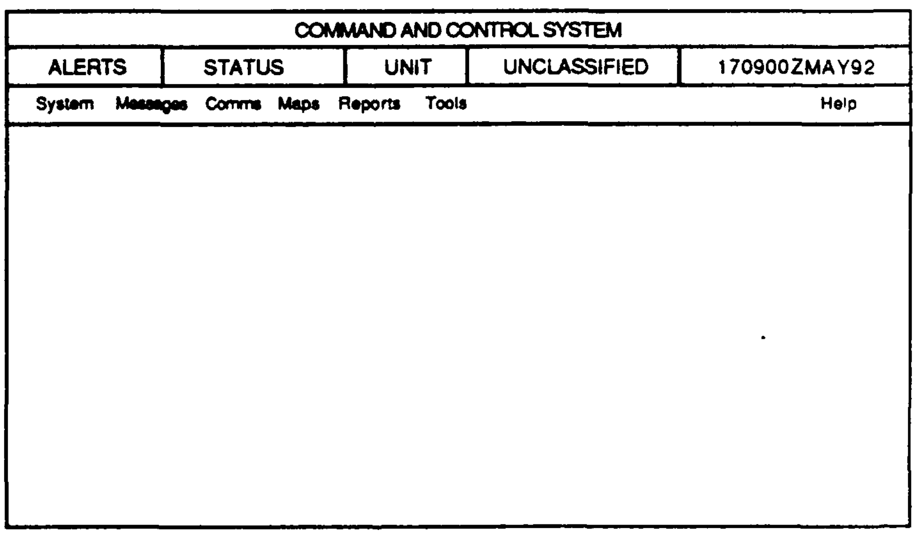


FIGURE 2.3: Example of a General List of Control Options

2.4 INTERRUPTS

2.4.1 REVIEW Option

If appropriate, provide a nondestructive REVIEW option that will return to the first display in a defined transaction sequence, permitting the user to review a sequence of entries and make necessary changes.

2.4.2 PAUSE and CONTINUE Options

If appropriate, provide PAUSE and CONTINUE options that will interrupt and later resume a transaction sequence without any change to data entries or control logic for the interrupted transaction.

2.4.3 Indicating PAUSE Status

If a PAUSE option is provided, display some indication of the PAUSE status whenever that option is selected by a user, and prompt the CONTINUE action that will permit resumption of the interrupted transaction.

2.4.4 END Option

If appropriate, provide an END option that will conclude a repetitive transaction sequence.

2.4.5 Aborting or Escaping from a Function

The system should make it easy for the user to abort, escape, or exit from a current operation or function (see Paragraph 2.5.9).

2.4.6 Indicating System Status

The user should be informed that system action is continuing. The "working" indication should have dynamic aspects to keep the user informed as to continuing system function.

2.4.7 SUSPEND Option

2.4.7.1 Usage

If appropriate, provide a SUSPEND option that will preserve current transaction status when a user leaves the system and permit resumption of work when the user later logs back onto the system.

2.4.7.2 Indicating SUSPEND Status

If a SUSPEND option is provided, display some indication of the SUSPEND status whenever a user selects that option. Prompt the user with those procedures that permit resumption of the suspended transaction at the subsequent log-on. For example, specifically prompt the user with "Type EXIT to return to application."

2.4.8 System Interruptions

The system should interrupt the user only when necessary to prompt response, to provide essential feedback, and to signal errors.

2.4.9 CANCEL Option

If appropriate, provide a CANCEL option that will erase changes just made by the user and restore the current display to its previous version.

2.4.10 Distinctive Interrupt Options

If different types of user interrupts are provided, design each interrupt function as a separate control option with a distinct name.

2.4.11 GOBACK Option

If appropriate, provide a nondestructive GOBACK option that will display the previous transaction.

2.4.12 RESTART Option

If appropriate, provide a RESTART option that will cancel entries made in a defined transaction sequence and will return to the beginning of the sequence. When data entries or changes will be nullified by restart, require the user to CONFIRM.

2.5 ERROR MANAGEMENT

2.5.1 User Confirmation of Destructive Entries

When a control entry (including log-off) will cause an extensive change in stored data, procedures, and/or system operation, and particularly if it cannot be easily reversed, notify the user and require confirmation of the action before implementation. See Paragraph 2.5.9.

2.5.2 User Warned of Potential Data Loss

Word the prompt for a CONFIRM action to warn the user explicitly of any possible data loss.

2.5.3 Errors in Stacked Commands

If an error is detected in a stacked series of command entries, the system should either consistently execute to the point of error or consistently require the user to correct errors before executing any command.

2.5.4 Partial Execution of Stacked Commands

If only a portion of a stacked command can be executed, notify the user and provide appropriate guidance to permit correction, completion, or cancellation of the stacked command.

2.5.5 Flexible GOBACK for Error Correction

Allow the user to GOBACK easily to previous steps in a transaction sequence in order to correct an error or make any other desired change.

2.5.6 Explicit Entry of Corrections

When the user has completed correcting an error, whether a command entry or data entry, require an explicit action to re-enter the corrected material. Use the same ENTER action for re-entry that was used for the original entry.

2.5.7 Prompting Command Correction

If an element of a command entry is not recognized or is logically inappropriate, the system should prompt the user to correct that element, rather than require re-entry of the entire command.

2.5.8 Immediate Data Correction

If a data entry transaction has been completed and errors are detected, allow the user to make corrections directly and immediately.

2.5.9 Distinctive Confirm Action

Provide an explicitly labeled confirm control, such as a function key or widget (e.g., control button or dialog box) different from the ENTER control, for user confirmation of questionable or destructive control and data entries. See Figure 2.4.

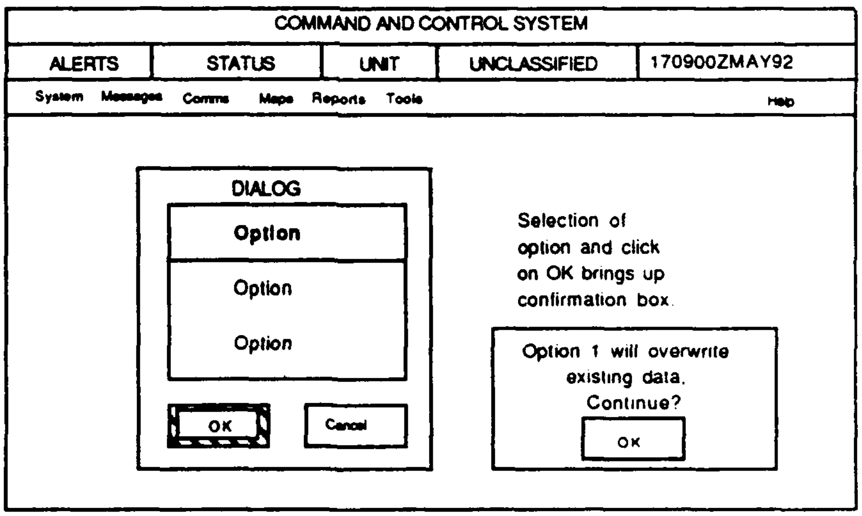


FIGURE 2.4: Example of a Distinctive Confirm Action, Using a Dialog Box

2.5.10 UNDO to Reverse Control Actions

Ensure that any user action can be immediately reversed by an UNDO command.

2.5.11 Appropriate Response to All Entries

Design software to provide an appropriate response for all possible control entries, correct and incorrect. For example, selecting an incorrect function key should cause a message indicating the appropriate selections.

2.5.12 Appropriate Terms For All Entries

Software must be consistent in the use of terms, and only the most explicit term should be used. The term "cancel" should be used for cancel functions, rather than a simple acknowledgement such as "OK." Use of complex terms (i.e., "Save & Apply" or "Exit to Prior Screen") should be avoided, if possible. Complex terms should have one consistent meaning within an application.

2.5.13 Display Duration

Notices, alerts, and informational displays should remain visible to the user until responded to by specific user action. Field use of computers creates a situation where the user may not be continuously monitoring the screen presentation. Therefore, automatic time-outs are not recommended.

2.5.14 Selection Errors

The pointing device interface uses both single and double clicks for control actions. The software needs to protect the system from inadvertent double clicks by the user. The protection supplied should be consistent with user and system requirements.

2.5.15 Inappropriate Item Selection

The user should be cued but not allowed to select items that are not available. Output fields should not allow data entry without the user acknowledging the selection of the option. The software should prevent data entry in any field that is not appropriate.

2.6 ALARMS

2.6.1 Special Acknowledgment of Critical Alarms

When the user must acknowledge special or critical alarms in a unique way, such as a special combination of key strokes, ensure this acknowledgment does not inhibit or slow the response to the condition initiating the alarm.

2.6.2 Alarm Reset

Provide the user with a simple means of turning off an auditory alarm without erasing any displayed message that accompanies the auditory signal. For noncritical alarms, provide a simple method for acknowledging and turning off the signal.

2.6.3 Distinctive and Consistent Alarms

Ensure alarm signals and messages are distinctive for each class of event, such as INCOMING MESSAGE ALERT, TERMINAL STATUS, TRACK ALERT, etc.

2.6.4 Alarm Definition by User

When monitoring tactical situations or tactical data status, allow the user to define the conditions (such as priorities, percentages, target flight path, etc.) that result in a software-generated alarm, alert, or status message.

REFERENCE LIST

Paragraph	References
2.1.1	Smith and Mosier (1986) para 3.0-9
2.1.2	Lickteig (1989) p. 9
2.1.3	Williams et al. (1987a) Appendix A p. A-2; Smith and Mosier (1986) para 2.7.2-2
2.1.4	Smith and Mosier (1986) para 3.0-12
2.1.5	Chao (1987) p. 360 and 361
2.1.6	Smith and Mosier (1986) para 3.0-17
2.1.7	Smith and Mosier (1986) para 3.0-7
2.1.8	Lickteig (1989) p. 35
2.1.9	Lickteig (1989) p. 34; Williams et al. (1987a) Appendix A p. A-3
2.1.10	Smith and Mosier (1986) para 3.0-8
2.1.11	Hamel and Clark (1986) p. 29; Smith and Mosier (1986) para 3.0-3
2.1.12	Williams et al. (1987b) Appendix A p.A-3
2.1.13.1	Smith and Mosier (1986) para 3.0-6, 19
2.1.13.2	Smith and Mosier (1986) para 3.0-16
2.1.13.3	Smith and Mosier (1986) para 3.0-11
2.1.14.1	Smith and Mosier (1986) para 3.0-5
2.1.14.2	Smith and Mosier (1986) para 3.0-2
2.1.14.3	Smith and Mosier (1986) para 3.0-19
2.1.14.4	Smith and Mosier (1986) para 3.0-19; Salvendy (1987)

REFERENCE LIST (Cont.)

Paragraph	References
2.1.15.1	Williams et al. (1987b) Appendix A p A-3; Hamel and Clark (1986) p. 30; DoD (1989b)
2.1.15.2	Smith and Mosier (1986) para 3.0
2.1.15.3	Smith and Mosier (1986) para 3.0-15; Baeker (1980); Hamel and Clark (1986) p. 30; Mallary (1985) p. 26; McCann (1983) p. 4; Slominski and Young (1988) p. 5
2.1.15.4	Smith (1986) para 1.0-3
2.1.16.1	DoD (1989a) para 5.15.4.1.1.1; Smith and Mosier (1986) para 3.0-20
2.1.16.2	DoD (1989a) para 5.15.4.1.1.2
2.1.16.3	Smith and Mosier (1986) para 3.0-21, DoD (1989a) para 5.15.4.1.1.3
2.1.17.1	Smith and Mosier (1986) para 3.1-2
2.1.17.2	Smith and Mosier (1986) para 3.0-18;
2.1.18.1	Bowser (1991) p. 6; Lewis and Fallesen (1989) p. 94; HFS (1988)
2.1.18.2	Bowser (1991) p. 6; Lewis and Fallesen (1989) p. 94
2.1.18.3	Lewis and Fallesen (1989) p. 94
2.2.1	Bowser (1991) p. 6; Smith and Mosier (1986) para 3.4-1; Bullinger et al. (1987) pp. 312-3; Hamel and Clark (1986) p. 26-27; Mallary (1985) p. 28, 9 p. 2, 14 p. 6, 3 p. 360

REFERENCE LIST (Cont.)

Paragraph	References
2.2.2	Smith and Mosier (1986) para 3.4-2
2.2.3	Smith and Mosier (1986) para 3.4-3
2.2.4	Smith and Mosier (1986) para 3.4-4
2.2.5	Smith and Mosier (1986) para 3.4-7
2.2.6	Smith and Mosier (1986) para 3.4-6
2.2.7	Smith and Mosier (1986) para 3.4-5
2.3.1	Smith and Mosier (1986) para 3.2-12
2.3.2	Smith and Mosier (1986) para 3.2-11
2.3.3	Smith and Mosier (1986) para 3.2-19
2.3.4	Smith and Mosier (1986) para 3.2-8
2.3.5	Smith and Mosier (1986) para 3.2-10
2.3.6	Smith and Mosier (1986) para 3.2-13
2.3.6.1	Smith and Mosier (1986) para 3.2-15
2.3.6.2	Bowser (1991) p. 6; Smith and Mosier (1986) para 3.2-18
2.3.6.3	Smith and Mosier (1986) para 3.2-14
2.3.7.1	Smith and Mosier (1986) para 3.2-7
2.3.7.2	Smith and Mosier (1986) para 3.2-6
2.3.8	Smith and Mosier (1986) para 3.2-5
2.3.9	Smith and Mosier (1986) para 3.2-2
2.4.1	Smith and Mosier (1986) para 3.3-5
2.4.2	Smith and Mosier (1986) para 3.3-8
2.4.3	Bowser (1991) p. 7; Smith and Mosier (1986) para 3.3-7
2.4.4	Baeker (1980); Williams et al. (1987a) Appendix A p. A-2; Smith and Mosier (1986) 3.3-1

REFERENCE LIST (Cont.)

Paragraph	References
2.4.6	Harell (1987) p.5
2.4.7	Smith and Mosier (1986) para 3.3-3
2.4.8	Smith and Mosier (1986) para 3.3-2
2.4.9	Smith and Mosier (1986) para 3.3-4
2.4.10	Smith and Mosier (1986) para 3.3-6
2.5.1	Smith and Mosier (1986) para 3.5-7 and 3.5-11
2.5.2	Smith and Mosier (1986) para 3.5-8
2.5.3	Smith and Mosier (1986) para 3.5-4
2.5.4	Smith and Mosier (1986) para 3.5-5
2.5.5	Smith and Mosier (1986) para 3.5-13
2.5.6	Smith and Mosier (1986) para 3.5-6
2.5.7	Smith and Mosier (1986) para 3.5-3
2.5.8	Smith and Mosier (1986) para 3.5-12
2.5.9	Smith and Mosier (1986) para 3.5-9
2.5.10	Smith and Mosier (1986) para 3.5-10
2.5.11	Smith and Mosier (1986) para 3.5-1
2.5.12	Bowser (1991) p. 8
2.5.13	Bowser (1991) p. 7
2.6.1	Smith and Mosier (1986) para 3.6-5
2.6.2	Smith and Mosier (1986) para 3.6-4 and 3.6-3
2.6.3	Smith and Mosier (1986) para 3.6-2
2.6.4	Smith and Mosier (1986) para 3.6-1

3.0 FORM FILLING

Form Filling, as an interactive dialog, requires little or no training and allows a relatively slow system response time. Command and control systems primarily use form filling for completing standard message and data entry forms. As with any aspect of the human-computer interface design, consistency of design is of paramount importance. The following guidelines deal more with interactive control than with data entry. For more information on data entry, see Smith and Mosier (1986), MIL-STD-1472D (DoD 1989), or DOD-HDBK-761A (DoD 1989b).

This Page Intentionally Left Blank.

3.1 GENERAL

3.1.1 Usage

Form-filling dialog should be used:

a. as an aid for composing complex control entries

b. as a means of displaying default values for the parameters in complex control entries

c. for tasks where some flexibility in data entry is needed (such as the inclusion of optional as well as required items), where users will *have moderate training, and/or where* computer response may be slow.

3.1.2 Interrupts for Multiple Entries

Where forms have multiple entries, provide the user BACKUP, CANCEL, and RESTART capabilities for editing the form prior to final input into the system. See Figure 3.1.

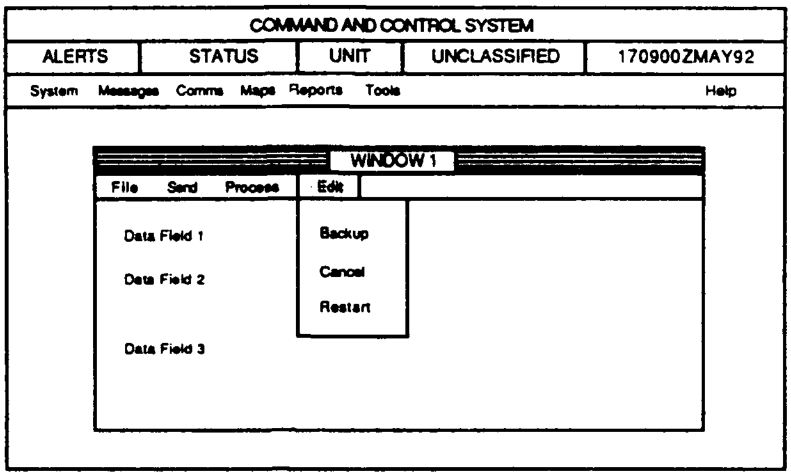


FIGURE 3.1: Example of Interrupt Capability for Multiple Entries

3.1.3 Explicit Data Entry

Data entry should be accomplished through an explicit action, such as pressing the ENTER key.

3.2 DEFAULTS

3.2.1 Automatic Display of Default Data

If default values are used in data entry fields, display them automatically in the appropriate data entry field.

3.2.2 Replacement of Default Values

When the user replaces a default value in a data entry field, ensure the default definition is not changed.

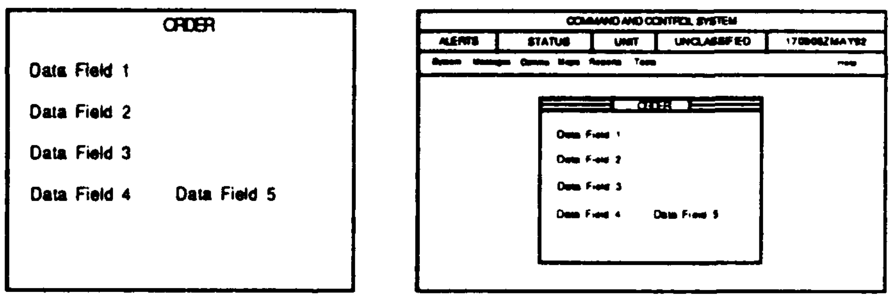
3.3 CONSISTENCY

3.3.1 Consistent Format for Control Forms

Ensure forms for control entry are consistent in format.

3.3.2 Format of Form and Hard Copy

When the user is entering data in a computer from a hard copy, the computer form and hard-copy format should be identical, as illustrated in Figure 3.2.



Paper Original

Data Entry Form

FIGURE 3.2: Example of How a Paper Entry Form and a Computer Data Entry Form Should Be Consistent

3.3.3 Entry Dialog Consistency

Dialog strategies for entering words and numbers should be consistent for a given set of logical functions throughout the system.

3.3.4 Standard Formats

Data and/or processes that have standard information requirements need to provide the standard format as part of the data screen. Message formats should include a template for the standard format. Using data entry screens that do not conform to user-accepted format will confuse users.

3.4 CURSOR MOVEMENT

3.4.1 Cursor Movement Into Non-Data Area

Command and control applications should not allow the user to move the cursor into a non-data entry area during form filling.

3.4.2 Convenient Cursor Movement

Ensure the user has a convenient method for cursor control, such as the use of the tab, enter key, or pointing device.

3.4.3 Cursor Movement by Explicit Action

When moving from one data entry field to another, the user should be required to take an explicit action, such as hitting the tab control. The software should not automatically advance to the next field.

3.4.4 Cursor/Pointing Device Interaction

Pointing device-to-cursor movement ratio should be close to 1:1. If appropriate, the user should be able to select the movement ratio.

3.4.5 Initial Cursor Location

When the user first calls up a form, the cursor should be positioned in the first character space of the first data entry field. See Figure 3.3.

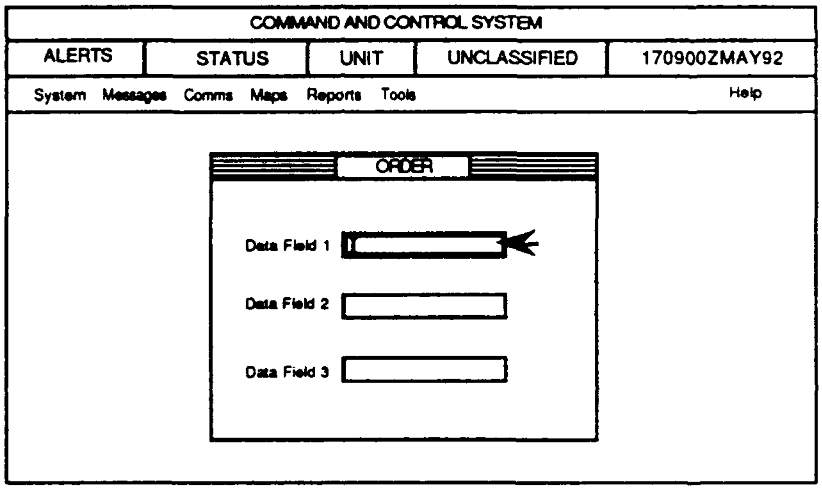


FIGURE 3.3: Cursor Should Appear in the First Character Space of the First Data Entry Field

3.5 DATA FIELD

3.5.1 Variable Data Field Format

For data entry fields with variable lengths, the software should automatically justify or truncate the data for the user. No leading characters should be required. See the example in Figure 3.4.

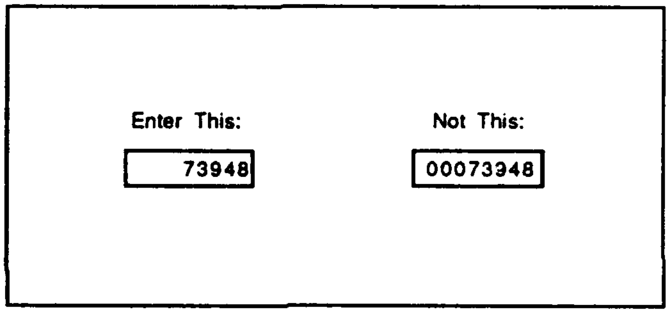


FIGURE 3.4: Data Entry Should Not Require Leading Zeros

3.5.2 Consistent Format of Data Fields

The format of data fields used frequently on different forms within and among applications should be consistent from one display to another and should use a format convention consistent with the user's expectations.

3.5.3 Computer Identification of Data Omission

When the user omits data, the software should present a message bringing attention to the missing data. The user should either be able to immediately insert the missing data or to identify with a special symbol that the data entry will be deferred.

3.5.4 Subgroups Within a Data Field

For data fields longer than 5 to 7 characters, break the field into subgroups of 3 to 4 characters that are separated by a space or delimiter. This should follow a convention consistent with the user's expectations.

3.5.5 Data Field Boundaries

Data fields should have distinctly marked boundaries.

3.5.6 Data Field Identification

Data entry fields should be clearly identified. Because the interface designs are often complex, users need a positive visual means to identify data entry fields.

3.5.7 Field Length

Data entry fields should be of fixed length, with cues given for their length, as illustrated in Figure 3.5.

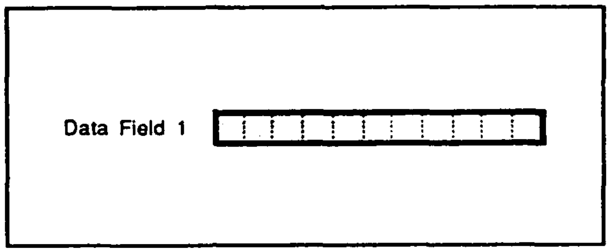


FIGURE 3.5: Visual Cues for Field Length

3.5.8 Overwriting

Data entry should not require overwriting of existing or default information. The field should either be empty, or the user should be required to perform an explicit control entry to erase the default data.

3.5.9 Numeric Data Fields

Numeric data in decimal format should use the decimal as part of the data display. Care should be taken to ensure the field size is adequate for the data range.

3.6 ERROR MANAGEMENT

3.6.1 Error Correction for Characters and Fields

Ensure the user can easily correct errors on a character-by-character and field-by-field basis.

3.6.2 Error Messages

Ensure the software provides understandable error messages to the user when an unacceptable value is entered in a data field.

3.7 FORM LAYOUT

3.7.1 Multiscreen Form Numbering

If multiscreens are used for a transaction, provide page numbers for each screen, as illustrated in Figure 3.6.

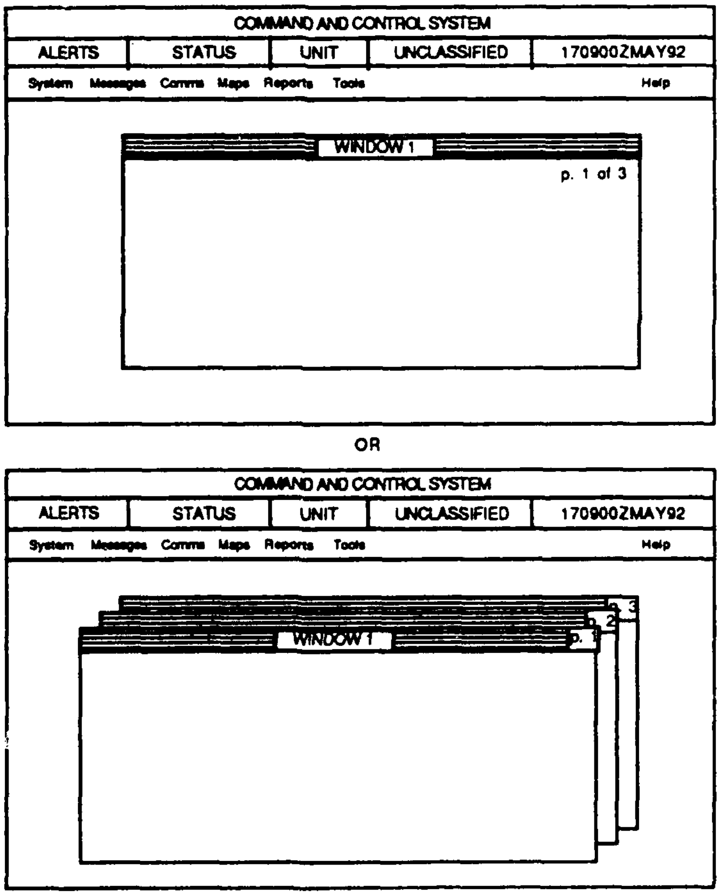


FIGURE 3.6: Example of How Multiple Screens Can Be Numbered

3.7.2 Logical Grouping of Data Fields

Related data fields should be grouped together on the same form.

3.7.3 Explanatory Messages for Data Fields

Provide explanatory messages for data fields that become visible when the cursor is placed in a field, when a user queries a field by clicking on the title, or by a context-sensitive help system. See the example in Figure 3.7.

3.7.4 Distinguishing Data Fields from Other Information

Messages and instructions on a form should be distinguished from data entry fields through consistent location or other means of high-lighting. Refer to Figure 3.7.

3.7.5 Spacing and Boundaries

Each data field should have visible space and boundaries between it and other fields.

3.7 Form Filling - Form Layout

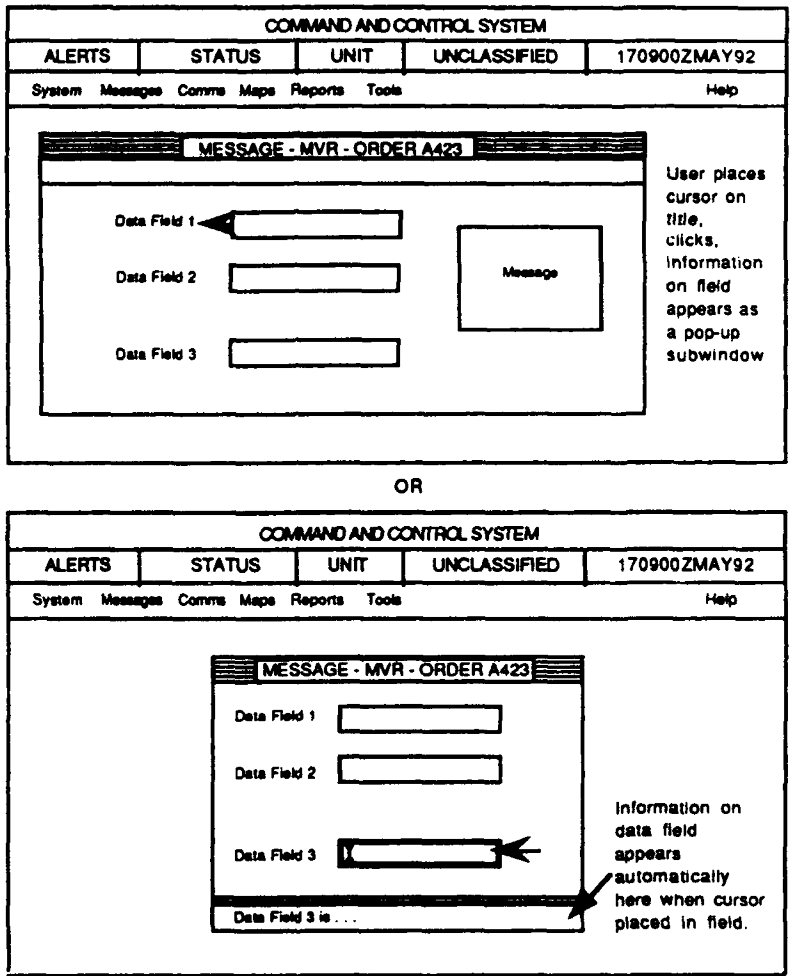


FIGURE 3.7: Example of How Explanatory Messages Can Be Provided

3.7.6 Grouping and Sequencing Fields

Data entry fields should be grouped and ordered on the form in a way that is logical for the task to be performed. This can be by sequence, frequency, or importance.

3.7.7 Form Title

Each form-filling dialog display page should have a meaningful title located at the top of the form, as illustrated in Figure 3.8.

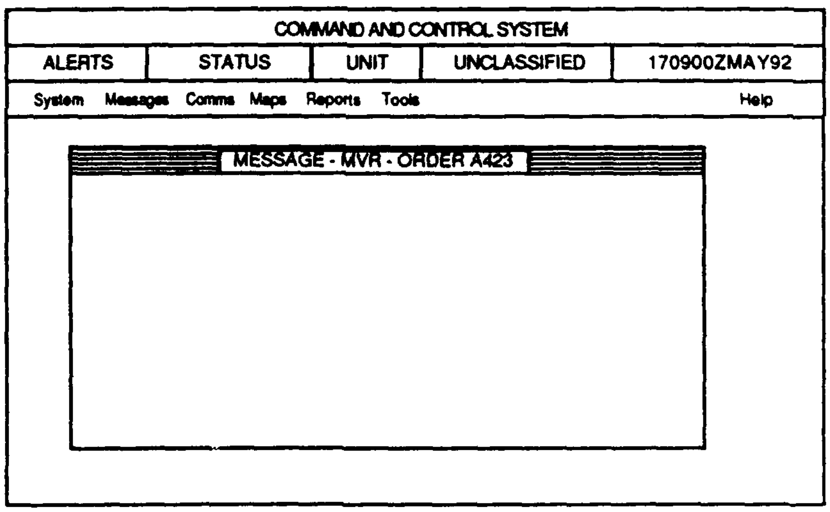


FIGURE 3.8: Example of a Form Title

3.7.8 Optional Field Labels

Optional fields should be labeled or coded in a readily apparent manner. See Figure 3.9.

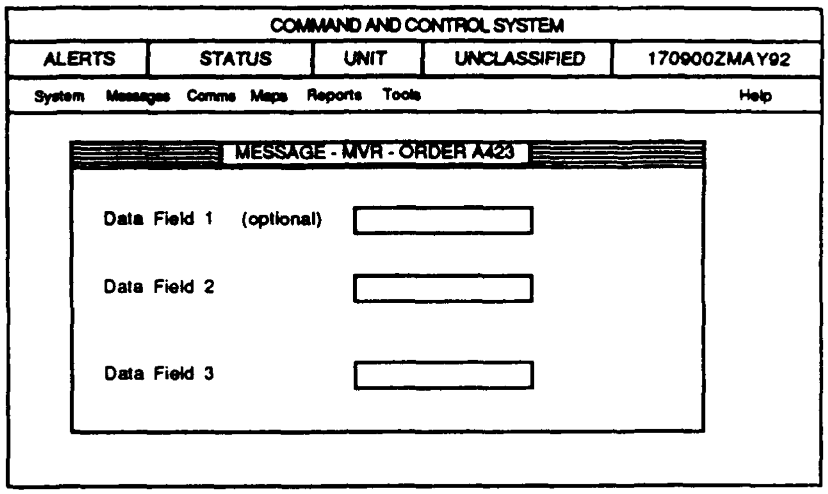


FIGURE 3.9: Example of an Indication of an Optional Field

3.7.9 Optional Field Defaults

When a data entry field in a form is optional, any value displayed in that field should be a default value. See Paragraph 3.5.7.

3.7.10 Mandatory Fields

The software application should not allow the user to bypass a mandatory field without data entry. See Paragraph 3.5.3.

3.8 LABELING

3.8.1 Distinctive Labeling

Data fields, unless similar or identical, should have distinctive, explicitly descriptive labels.

3.8.2 Data Field Label Location

Application data entry field labels should be located either directly to the left or above the actual entry field and separated by at least one character.

3.8.3 Similar Data Field Labeling

Similar data entry fields should be labeled and located consistently for all forms.

3.8.4 Consistent Labels

Labels and instructions should be consistent from one application to another within a command and control application and to the extent possible across all command and control systems.

3.8.5 Field Label Familiarity

Labels for data fields should be composed of terms familiar to the user and the task to be performed.

3.8.6 Understandable Labeling

Labeling for data fields and instructions should be easily understood by the typical user.

3.8.7 Units of Measure

Units of measure should be part of the data entry field label. If measurement units can change, this portion of the label should change automatically when new units are selected.

3.8.8 Blanks Versus Nulls

There should be a visible distinction between blanks and nulls in a data field.

REFERENCE LIST

Paragraph	References
3.1.1a	Smith and Mosier (1986) para 3.1.2-2
3.1.1b	Smith and Mosier (1986) para 3.1.2-3
3.1.1c	Smith and Mosier (1986) para 3.1.2-1
3.1.2	Chao (1986) p. 13
3.1.3	Chao (1986) p. 13
3.2	Chao (1986) p. 13
3.3.1	Smith and Mosier (1986) para 3.1.2-4
3.3.2	Sidorsky (1984) p. 1.1-11; Chao (1986) p. 13
3.3.3	Lewis and Fallesen (1989) p. 8; Chao (1986) p. 13
3.3.4	Bowser (1991) p. 8
3.4.1	Chao (1986) p. 13
3.4.2	Shneiderman (1988) p. 702
3.4.3	Chao (1986) p. 13
3.4.4	Chao (1986) p. 13
3.5	Chao (1986) p. 13
3.5.9	Bowser (1991) p. 8
3.6	Shneiderman (1988) p. 702
3.7.1	Chao (1986) p. 13
3.7.2	Chao (1986) p. 13
3.7.3	Chao (1986) p. 703
3.7.4	Chao (1986) p. 13
3.7.5	Shneiderman (1988) p. 702
3.7.6	Shneiderman (1988) p. 702
3.7.7	Shneiderman (1988) p. 702

REFERENCE LIST (Cont.)

Paragraph	References
3.7.8	Shneiderman (1988) p. 703; Chao (1986) p. 13
3.7.9	Chao (1986) p. 13
3.7.10	Avery et al. (1990) p. 3-19
3.8.1	Chao (1986) p. 13
3.8.2	Chao (1986) p. 13
3.8.3	Chao (1986) p. 13
3.8.4	Shneiderman (1988) p. 702
3.8.5	Shneiderman (1988) p. 702
3.8.6	Shneiderman (1988) p. 702
3.8.7	Chao (1986) p. 13
3.8.8	Chao (1986) p. 13

4.0 FUNCTION KEYS

The two types of function key are fixed and variable. The fixed key has only one predefined function associated with it. The variable key function will vary depending on the system mode or level within the interactive dialog. The function for the variable key is communicated to the user by changing the label located adjacent or internal to the key or through soft keys. Soft keys are objects on the display screen that represent the function keys on the keyboard. As the function of a key changes, the soft key labeling also changes. Fixed and variable function keys can be used together and with other dialog methods.

As with any interactive control method, the designer should be aware of the following overarching design guidelines:

- consistent design in terms of placement, labeling, and procedural logic

- easy association with the function being called up through labeling located adjacent to the function keys

- feedback

- spatial consistency between the labeling and the function key.

This Page Intentionally Left Blank.

4.1 GENERAL

Function keys are located on the keyboard and activate a computer software function when pressed.

4.1.1 Usage

Function key dialog should be considered for:

a. frequently required control entries

b. tasks requiring only a limited number of control entries or in conjunction with other dialog types as a ready means of accomplishing critical entries that must be made quickly, without syntax error

c. interim control entries (i.e., for control actions taken before the completion of a transaction).

4.1.2 Feedback for Function Key Activation

When function key activation does not result in any immediately observable response from the computer, provide users with some other form of computer acknowledgment and feedback. No system function should be activated without an indication to the user.

4.1.3 Disabling Unneeded Function Keys

When function keys are not needed for any current transaction, temporarily disable those keys under computer control; do not require the user to apply mechanical overlays for this purpose (see Paragraph 4.1.7).

4.1.4 Function Key Meaning

In general, each function key should control only one function. If a key must control more than one function, display the actual or current meaning of the function to the user by displaying soft keys on the screen.

4.1.5 Soft Key Design

Soft function keys displayed on the screen should be located close to the actual keyboard function keys and in the same spatial orientation. For example, on command and control system keyboards with function keys across the top, place soft keys at the bottom of the screen, directly above the keyboard as illustrated in Figure 4.1.

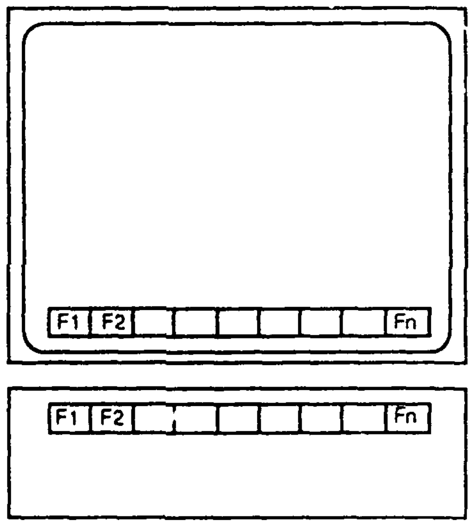


FIGURE 4.1: Example of Soft Key Location

4.1.6 Redundant Activation of Soft Key Function

The user should be able to activate the function represented on a soft key through either the function key or a pointing device, such as a mouse.

4.1.7 Indicating Active Function Keys

If some function keys are active and some are not, indicate the current subset of active keys in some noticeable way, such as brighter illumination or blanking of corresponding soft key labels on the display. See Figure 4.2.

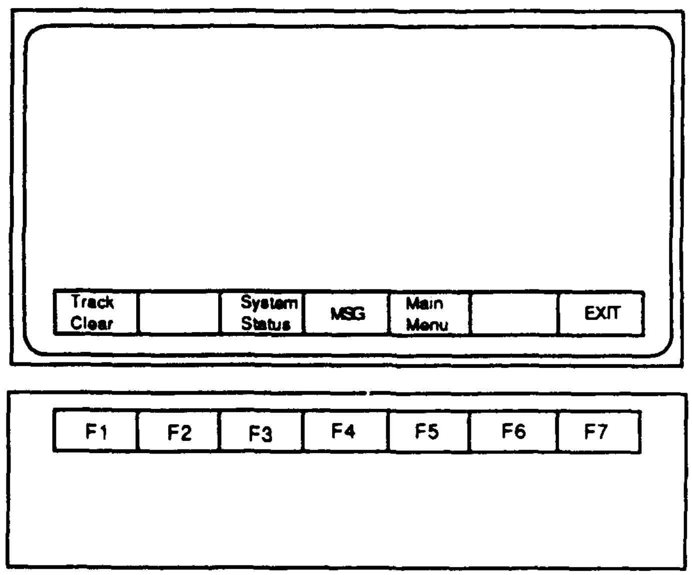


FIGURE 4.2: Suggested Method for Indicating Active and Inactive System Function Keys

4.1.8 Key Functionality Load

Avoid overloading the functionality of keys, however, provide the user with all necessary function controls required to perform the task.

4.1.9 Easy Return to Base-Level Functions

If functions assigned to a set of keys change as a result of user selection, give the user an easy means *to return to the initial, base-level functions* or menu. See Figure 4.3.

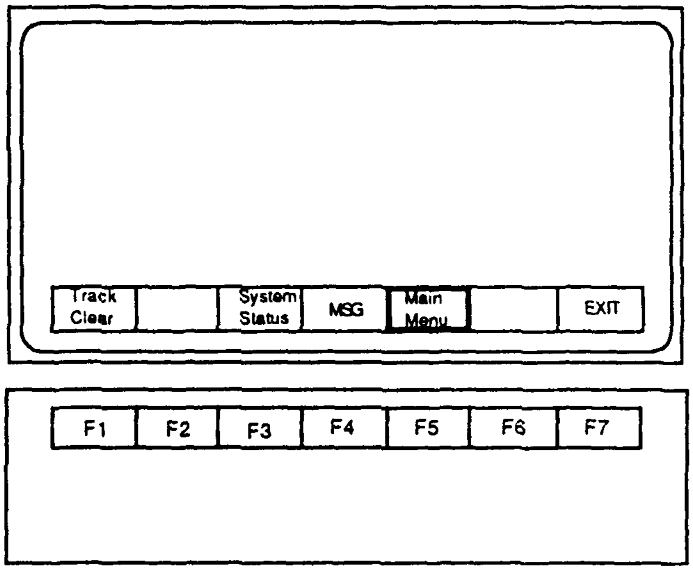


FIGURE 4.3: Recommended Method for a Return to Base-Level Functions

4.2 CONSISTENCY

4.2.1 Consistent Functions in Different Operational Modes

When a function key performs different functions in different operational modes, assign equivalent or similar functions to the same key.

4.2.2 Consistent Assignment of Function Keys

If a function is assigned to a particular key in one computer transaction, assign that function to the same key in other transactions.

4.3 DOUBLE KEYING

4.3.1 Logical Pairing of Double-Keyed Functions

If double (control/shift) keying is used, the functions paired on one key should be logically related to each other.

4.3.2 Consistent Logic for Double Keying

If double (control/shift) keying is used, the logical relation between shifted and unshifted functions should be consistent from one key to another.

4.4 LABELING

4.4.1 Distinctive Labeling of Function Keys

Label each function key informatively to designate the function it performs; make labels sufficiently different from one another to prevent user confusion.

4.4.2 Labeling Multifunction Keys

If a key is used for more than one function, always indicate to the user which function is currently available.

4.4.3 Labeling of Menu Options for Function Keys

When designing a command and control menu where options are selected through variable function keys, avoid using a function key number (e.g., F1, F2) as option designator. Instead, place the function key label just above the key on the display. See example in Figure 4.4.

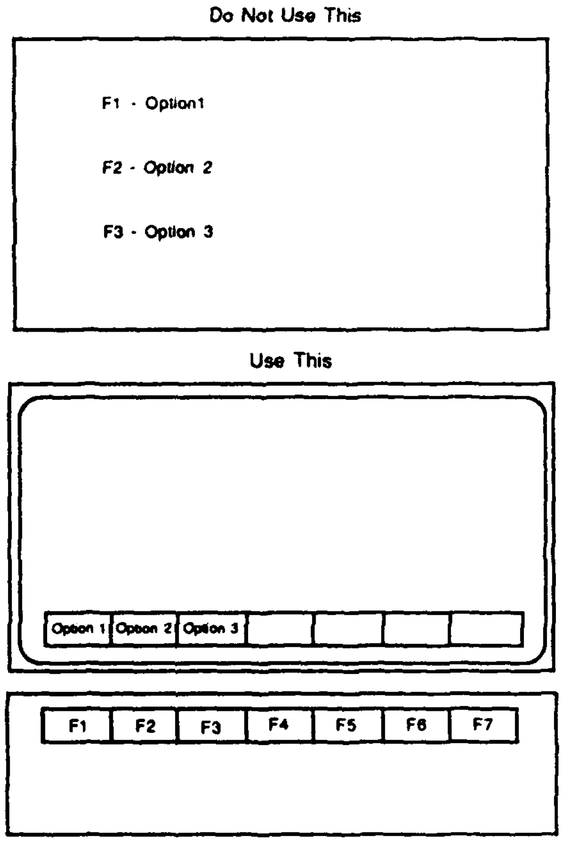


FIGURE 4.4: Recommended Location for Function Key Labels

4.5 LAYOUT

4.5.1 Layout Compatible with Use

Make layout of function keys compatible with their importance. Give keys for emergency functions a prominent position and distinctive coding (e.g., size and/or color).

4.5.2 Safeguards

Provide physical protection, software disabling, or interlocks for keys with potentially disruptive consequences.

4.5.3 Distinctive Location

Group function keys in distinctive locations on the keyboard to facilitate learning and use; place frequently used function keys in the most convenient locations. For command and control systems, this should be at the top of the keyboard, just below the corresponding labels.

4.6 SINGLE KEYING

4.6.1 Single Activation of Function Keys

Ensure any key will perform its labeled function with a single activation and will not change function with repeated activation without indicating the new function or change in mode.

4.6.2 Single Key for Continuous Functions

When a function is continuously available, assign that function to a single key.

4.6.3 Single Keying for Frequent Functions

Keys controlling frequently used functions should allow single key action and should not require double (control/shift) keying.

This Page Intentionally Left Blank.

REFERENCE LIST

Paragraph	References
4.1.1a	Smith and Mosier (1986) para 3.1.4-2
4.1.1b	Smith and Mosier (1986) para 3.1.4-1
4.1.1c	Smith and Mosier (1986) para 3.1.4-3
4.1.2	Bowser (1991) p. 9; Smith and Mosier (1986) para 3.1.4-10
4.1.3	Smith and Mosier (1986) para 3.1.4-12
4.1.4	Ziegler and Fähnrich (1988) p. 129
4.1.5	Ziegler and Fähnrich (1988) p. 129
4.1.6	Ziegler and Fähnrich (1988) p. 129
4.1.7	Smith and Mosier (1986) para 3.1.4-11
4.1.8	Nielson (1987) p. 248; Bullinger et al. (1987) p. 309; McCann (1983) pp.3-4
4.1.9	Smith and Mosier (1986) para 3.1.4-16
4.2.1	Smith and Mosier (1986) para 3.1.4-15
4.2.2	Smith and Mosier (1986) para 3.1.4-14
4.3.1	Smith and Mosier (1986) para 3.1.4-7
4.3.2	Smith and Mosier (1986) para 3.1.4-8
4.4.1	Smith and Mosier (1986) para 3.1.4-4
4.4.2	Smith and Mosier (1986) para 3.1.4-5
4.4.3	Sidorsky (1994) p. 1.1-12
4.5.1	Smith and Mosier (1986) para 3.1.4-18
4.5.2	Smith and Mosier (1986) para 3.1.4-18
4.5.3	Smith and Mosier (1986) para 3.1.4-17
4.6.1	Smith and Mosier (1986) para 3.1.4-9
4.6.2	Smith and Mosier (1986) para 3.1.4-13
4.6.3	Smith and Mosier (1986) para 3.1.4

This Page Intentionally Left Blank.

5.0 MENU DESIGN

Using menus as a dialog is widespread within computer systems. Command and control menus are frequently used in conjunction with other interactive methods, such as direct manipulation.

Using menus as a dialog has some advantages, a major one being that this requires little training or sophistication on the user's part. A user only needs to know the meaning of each option, then is guided step-by-step through the operation of the system. The number of keystrokes required to access a system function may also be reduced, thereby speeding the user-to-computer transaction.

On the other hand, using menus as a dialog has disadvantages. It does not enhance retention of commands and may actually increase response time for the more experienced user. Menus may take up a large part of the display surface. In addition, for complex sequences, using menus may require an extensive menu tree structure, and the user may easily become lost navigating through a complex menu tree.

A number of different types of menuing techniques are available to the designer, including pull-down, pop-up, and sequential display. Pull-down and pop-up menus tend to be used more in direct manipulation types of dialog. Sequential display, where a control action causes another menu to overwrite the previous menu, is used more in text-based systems.

All these types of menuing techniques can be hierarchical, or branching, in nature.

The following pages provide detailed design guidelines for menus used in command and control systems. To ensure a high level of user performance with menus, the designer should be aware of the following general guidelines:

- Consider selecting menus when:

  - tasks involve choosing among a constrained set of alternative actions

  - tasks require infrequent entry of data

  - the user may have little training

  - the computer response is relatively fast

  - tasks require infrequently used commands

  - command sets are so large that the user is not likely to commit all commands to memory

- Design the menu tree structure broad and shallow, rather than narrow and deep. Keep the number of top-level options large, with a small number of sublevels.

- Consider the experienced user and provide a mechanism by which the menu structure can be bypassed using a direct command.

The reader should also note that some of the guidelines discussed in the following paragraphs may be more appropriate for designing sequential display menus than menus used in direct manipulation. The designer should use judgment regarding which approach to take.

The designer should conform to a single style of interface, such as "Motif," throughout an application. Varying interface style confuses the user. Widgets or graphics as menu item selectors should be unique and clearly identifiable by the user.

This Page Intentionally Left Blank.

5.1 GENERAL

5.1.1 Consider Response Time and Display Rate

Display rate refers to the speed of "painting the screen."

If computer response time is long, create menus with a larger number of items. If display rate is slow, create menus with fewer items to reduce display time.

5.1.2 Instructions and Error Messages

Menu instructions and error messages should be indented and placed in the same position on the screen so the user knows where to look for this information.

5.1.3 Explicit Option Display

When entries for any particular computer transaction consist of a small set of options, show those options in a menu added to the working display, rather than require a user to remember them or to access a separate display.

5.1.4 Stacking Menu Selections

Stacking refers to stringing multiple commands together and executing them with one action.

For menu selection by code entry, when a series of selections can be anticipated before the menus are displayed, permit the user to combine those selections into a single stacked entry.

5.1.5 Menus Distinct from Other Displayed Information

If menu options are included in a display also intended for data review and/or data entry, ensure they are distinct from other displayed information. Locate menu options consistently, and use consistent visual cues for their special function.

5.1.6 Menu Bars

Menu bars provide system functions in a bar across the top of the display screen. The following guidelines apply to menu bars.

5.1.6.1 Using Menu Bars

Large screen displays in this context are defined as displays intended for multi-viewers. This includes projections and theater-type displays.

A menu bar is best used when screen size is small. When using large screen displays, the distance the cursor is required to travel may be too great to be effective.

5.1.6.2 Visibility of Menu Bar Options

Menu bar options, as illustrated in Figure 5.1, should remain constantly visible.

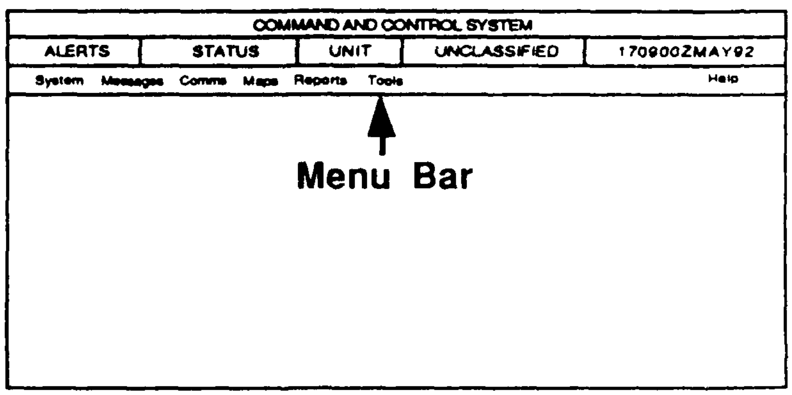


FIGURE 5.1: Example of a Menu Bar

5.1.7 Pull-Down Menus

Pull-down menus are lists of options attached to a selection on the menu bar that remain visible until user action is taken.

Pull-down menus, as illustrated in Figure 5.2, should be used instead of pop-up menus when cursor position on the screen is not important for information/option retrieval.

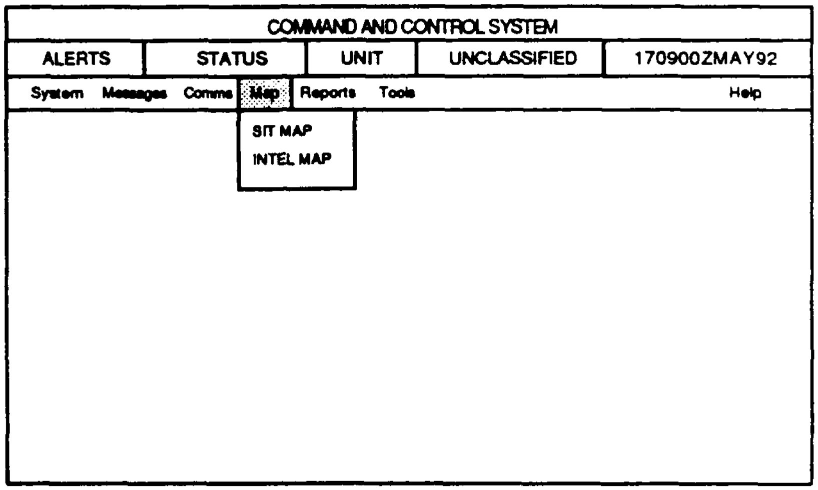


FIGURE 5.2: Example of a Pull-Down Menu

Pop-up menus are specific to their area on the display; each window or object may have its own individual pop-up menu.

5.1.8 Pop-Up Menus

Pop-up menus are lists of options that appear on the display screen in the form of a window (see Section 7.0). The following guidelines should be used when designing pop-up windows.

5.1.8.1 Pop-Up Menu Location

Pop-up menus should be connected to the cursor location and pop up near the object or higher level menu being manipulated, as illustrated in Figure 5.3.

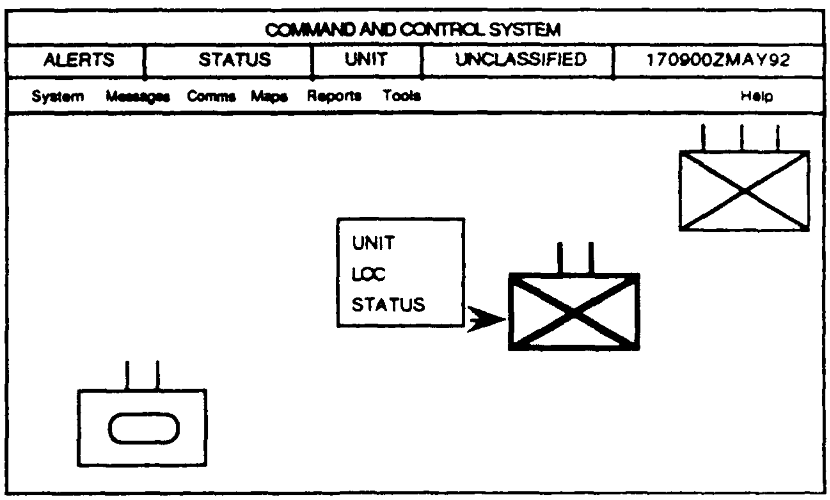


FIGURE 5.3: Example of a Pop-Up Menu

5.1.8.2 Selecting Options From Pop-Up Menus

Two methods to select from a pop-up menu are: 1) hold the button down while traversing options, then release to make the selection, or 2) move the cursor and press the button again for the selection. Use the second method because, although not the easier, it is less error-prone.

5.1.8.3 Selection Highlighting

When an option has been selected from a pop-up menu, it should remain highlighted.

5.2 FORMAT

5.2.1 General

5.2.1.1 Menu Format

Lists of menu and submenu items should be brief (no more than five to nine options), arranged in separate columns, aligned, and left-justified.

5.2.1.2 Consistent Display of Menu Options

When menus are provided across different displays, design them so option lists are consistent in wording and order.

5.2.1.3 Logical Grouping of Menu Options

Format a menu to indicate logically related groups of options, rather than an undifferentiated string of alternatives.

5.2.1.5 Sequence or Frequency Ordering

For a small number of menu items, use sequence or frequency to determine menu order.

5.2.1.6 Alphabetic Ordering

For a large number of menu options, use alphabetic ordering of menu items.

5.2.1.7 Numbering Menu Options

When task order is important, menu options should be listed by number, not by letter.

5.2.1.8 Display of Options

Design of a menu for a graphical user interface should display unavailable menu items in a visually distinct manner. Refer to Paragraph 2.3.5.

5.2.1.9 Single-Column List Format

When multiple menu options are displayed in a list, display each option on a new line (i.e., format the list as a single column).

5.2.1.10 Overlapping Items

Ensure menu options do not overlap controlled functions or appear to do so to the user.

5.3 HIERARCHICAL MENUS

5.3.1 Usage

For direct manipulation interactive control, hierarchical menus allow the user to browse the available options.

Use hierarchical menus:

a. when menu selection must be made from a long list and not all options can be displayed at once

b. if a selection list exceeds 10-15 items.

5.3.2 General Guidance

5.3.2.1 Organization and Labeling of Hierarchical Menus

When hierarchical menus are used, organize and label them to guide the user within the hierarchic structure. Currently active menu selections need to be identified to the user. The preferred method would use more than one mode (i.e., color and font, size and color of text, etc.).

5.3.2.2 Easy Selection of Important Options

Design hierarchical menus to permit immediate user access to critical or frequently selected options.

5.3.2.3 Indicating Current Position in Menu Structure

When hierarchical menus are used, display an indication of the user's current position in the menu structure. This could be done in the menu title, or as a page X of N notation on the menu page.

5.3.2.4 Consistent Design of Hierarchical Menus

When hierarchical menus are used, ensure the display format and option selection logic are consistent at every level of the hierarchical menu structure.

5.3.2.5 Graphic User Interface for Hierarchical Menus

Hierarchical menu design in a graphical user interface should be as simple as possible. The use of complex graphic structures is distracting to the user.

5.3.3 Navigating Hierarchical Menus

5.3.3.1 Inclusion of System-Level Menu

The system-level menu will act as a home base to which a user can always return as a consistent starting point for control entries.

Provide a system-level menu of basic options as the top level in a hierarchical menu structure, as illustrated in Figure 5.5.

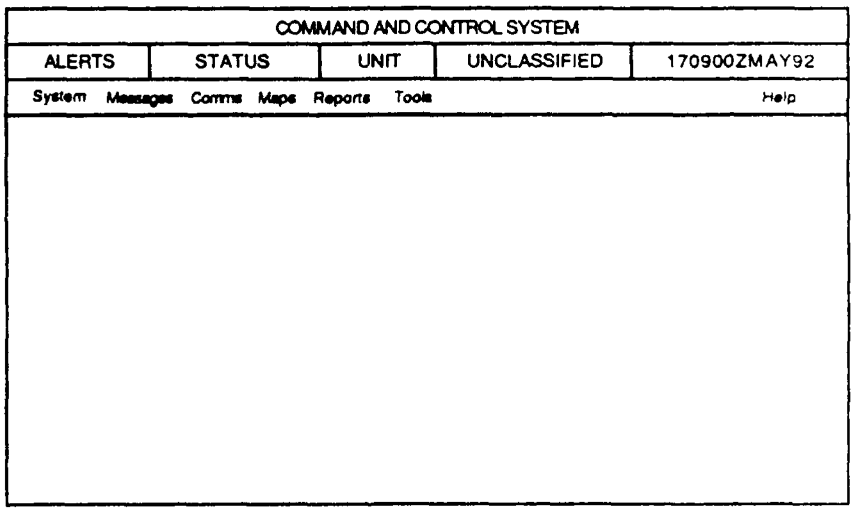


FIGURE 5.5: Example of a System-Level Menu

5.3.3.2 Organization and Labeling of System-Level Menu Listed Options

Control options for the system-level menu should be grouped, labeled, and ordered in terms of their logical function, frequency, and criticality of use.

5.3.3.3 Return to the System-Level Menu

When hierarchical menus are used, require the user to take only one simple control action to return to the system-level menu.

5.3.3.4 Return to Higher Level Menus

When hierarchical menus are used, require the user to take only one simple control action to return to the next higher level.

5.3.3.5 Control Options Distinct From Menu Branching

Format the display of hierarchical menus so options that actually accomplish control entries can be distinguished from those which merely branch to other menu frames. See Figure 5.6.

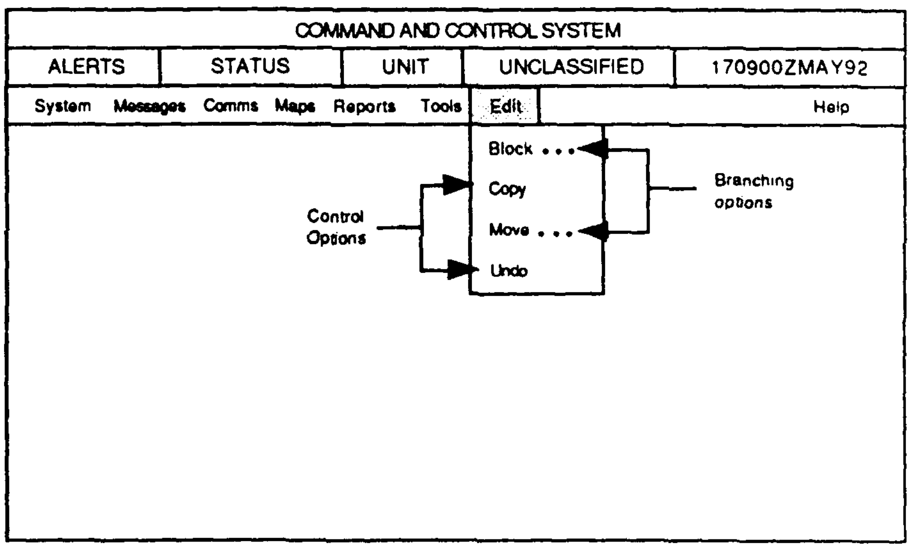


FIGURE 5.6: Distinction Between Control Options and Command Options

5.3.3.6 Hierarchical Menu-Browsing Methods in Direct Manipulation

Two basic methods for browsing options in hierarchical menus are used in direct manipulation interactive control:
1) select an option from one menu, which causes another menu to pop-up, or 2) move the cursor towards the right side of an option, causing a menu to pop-up. See Figure 5.7.

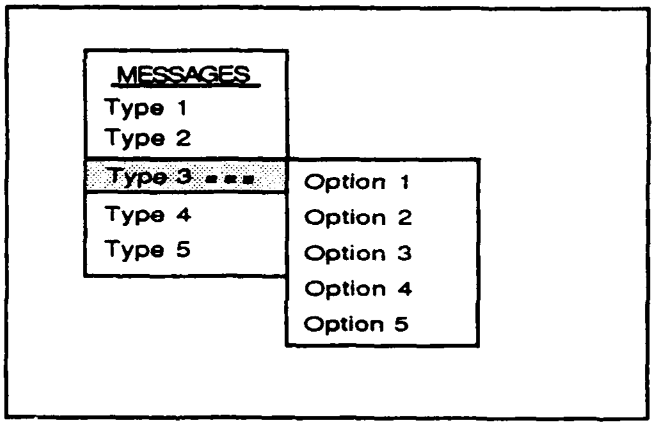


FIGURE 5.7: Example of a Hierarchical Menu

5.3.3.7 Use of Multiple Paths

Provide multiple paths to accommodate both the experienced and inexperienced user. Allow the experienced user to use type-ahead, jump-ahead, or other shortcuts to navigate through the menu selection system.

5.3.4 Hierarchical Menu Tree Depth and Breadth

5.3.4.1 Minimal Steps in Sequential Menu Selection

When the user must step through a sequence of menus to make a selection, design the hierarchical menu structure to minimize the number of steps required.

5.3.4.2 Use Broad Menu Trees

Use a broad and shallow menu tree, rather than a narrow and deep menu tree for command and control systems, as illustrated in Figure 5.8.

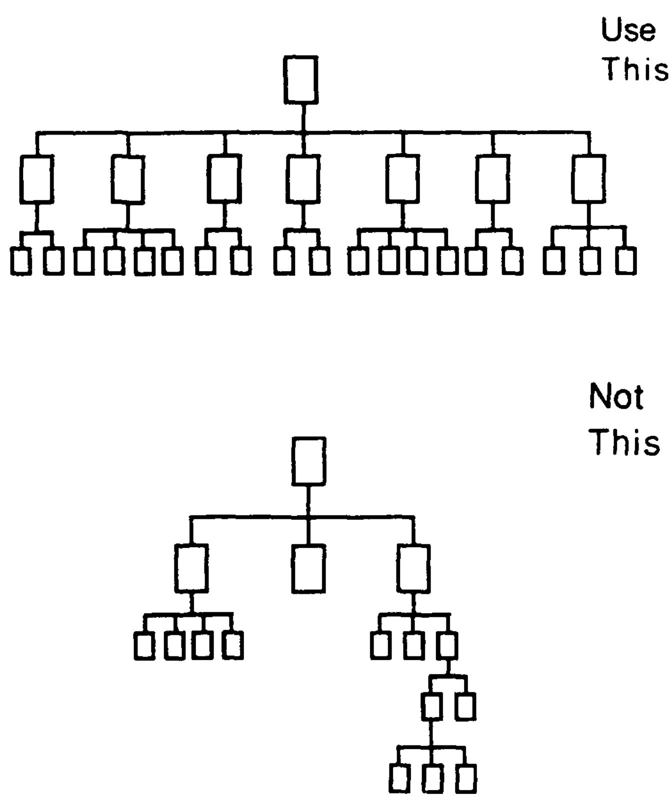


FIGURE 5.8: Broad and Shallow Menu Tree vs. Narrow and Deep Menu Tree

5.3.4.3 Minimize Menu Choices in the Middle

Minimize the number of menu choices midway through a hierarchical menu, as the user is more likely to get lost at this stage.

5.3.4.4 Software Navigation Aids

Software navigation aids are needed to assist the user in quickly selecting the proper area in the software. The user should be able to switch between software modules in a quick, easy manner, using an interface such as a tree or organization chart. The function should include the ability to select a menu or submenu directly, without going through intermediate steps. See Figure 5.9.

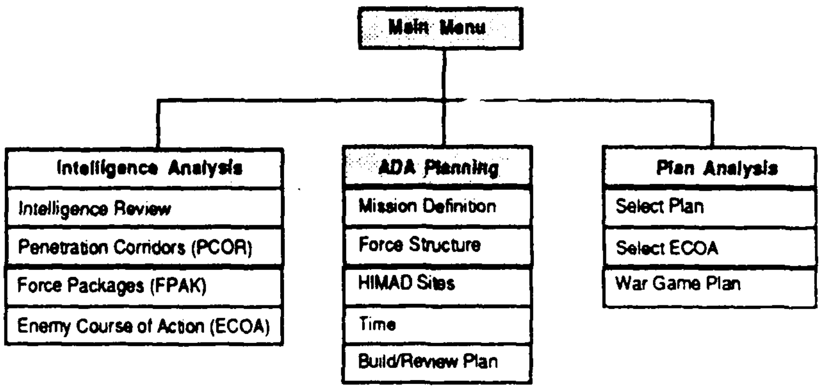


FIGURE 5.9: Example of a Tree Diagram Interface

5.4 ITEM SELECTION

5.4.1 General

5.4.1.1 Automatic Cursor Placement

When making a menu selection by pointing, on menu displays not included with data displays, the computer should place the cursor automatically at the first listed option. When menu selection is by code entry, place the cursor in the command entry area.

5.4.1.2 Minimize Menu Selections

Keep the number of menu selections to the absolute minimum to reduce system menu selection time.

5.4.1.3 Use a Combined Mode of User Interface

The users should be able to use two modes for menu selection: keying in a numeric or letter code, or placing the cursor at the option and selecting. See Figure 5.10.

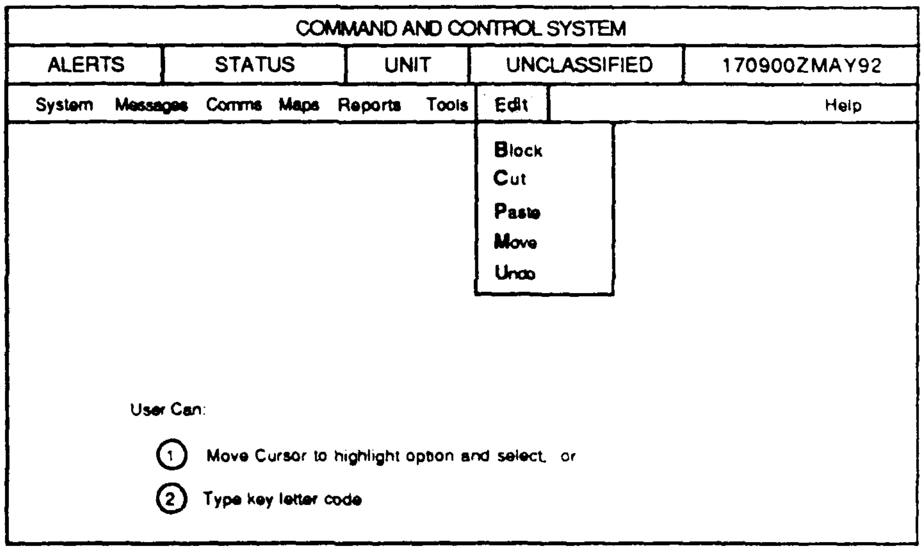


FIGURE 5.10: Example of a Combined Mode User Interface

5.4.1.4 Feedback for Menu Selection

When a user selects and enters a control option from a menu, if no natural response is immediately observable, the software should display some other acknowledgment of that entry. See examples in Figure 5.11.

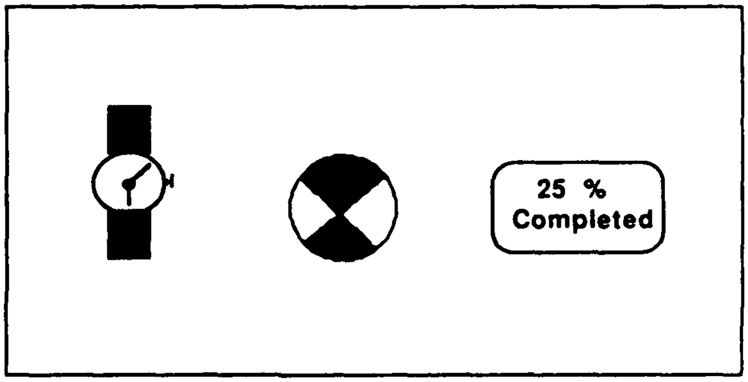


FIGURE 5.11: Icons that Acknowledge Entry Selection

5.4.1.5 Standard Area for Code Entry

When menu selection is accomplished by code entry, provide a standard command entry area where the user enters the selected code. Place that entry area in a fixed location on all displays.

5.4.1.6 Allow Abbreviated Menu Selections

Allow menu selections by the user to be accepted in either abbreviated or complete form. For example, the user should be able to use Q, QU, or QUIT.

5.4.2 Selection by Pointing

5.4.2.1 Menu Selection by Pointing

When menu selection is the primary means of sequence control, and especially if choices must be made from extensive lists of displayed control options, permit option selection by direct pointing (e.g., by mouse, trackball, etc.). See Section 6, Direct Manipulation.

5.4.2.2 Large Pointing Area for Selecting Options

The acceptable pointing area for menu options should be as large as is consistently possible. The area should include at least the displayed option label, plus a half-character distance around that label.

5.4.2.3 Dual Activation for Pointing

If menu selection is accomplished by pointing, provide dual activation, where the first action designates (positions a cursor at) the selected option, and a separate, second action makes an explicit control entry.

5.5 MENU OPTION LABELING

5.5.1 General

5.5.1.1 Use of Key Words

Menu items should begin with a key word.

5.5.1.2 Menu Options Worded as Commands

The wording of menu options should consistently represent commands to the computer, rather than questions to the user.

5.5.1.3 Menu Categories

Menu category labels should be comprehensible and unique. The words, phrases, and titles should state options in clear English.

5.5.1.4 Labeling Grouped Options

If menu options are grouped in logical sub-units, give each group a descriptive label that is distinctive in format from the option labels themselves. See Figure 5.12.

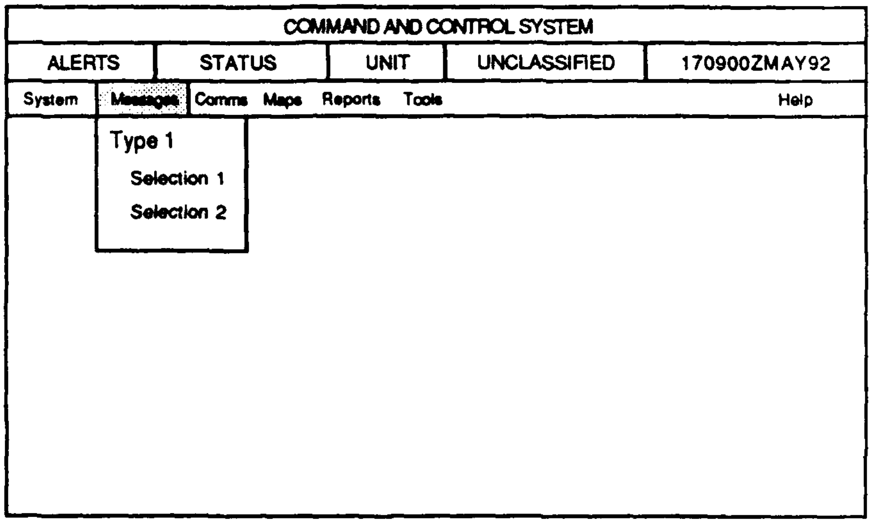


FIGURE 5.12: Example of Distinctive Subunit Labels

5.5.1.5 Use Familiar Terminology

Use familiar terminology when labeling menus, but ensure items are distinct from one another.

5.5.2 Selector

Lettered menu items have the following advantages: more single entry keys are available; there is less chance of a keying error; and mnemonic keying of entries is possible.

5.5.2.1 Best and Worst Selectors for Menu Items

Mnemonics is a technique to assist in improving the user's memory. Compatible or mnemonic letters are the best selectors for menu items while incompatible letters are the worst. Numbers are intermediate selectors.

5.5.2.2 Do Not Combine Codes

Numbered menu items have the following advantages: sequencing of items is clear; non-typists can easily locate numbers; and the user can quickly see how many options are available.

Letter and numeric codes should not be combined in the dialog.

5.5.2.3 Selection of Menu Titles

Use selectors that closely match the item represented, to facilitate user retention of commands.

5.5.2.4 Numbering

Number menu items starting with 1 -- not with 0.

5.5.2.5 Consistent Coding of Menu Options

If letter codes are used for menu selection, use those letters consistently in designating options from one transaction to another.

5.5.2.6 Displaying Option Code

When the user must select options by code entry, display the code associated with each option in a consistent distinctive manner. See Table 5.1.

TABLE 5.1: Examples of Option Codes

Code		Option
P	=	Previous Page
N	=	Next Page
U	=	Undo
Del	=	Delete

REFERENCE LIST

Paragraph	References
5.1.1	Shneiderman (1987) p. 107
5.1.2	Shneiderman (1987) p. 115
5.1.3	Smith and Mosier (1986) para 3.1.3-16; Sidorsky (1984) para 5.1-14
5.1.4	Smith and Mosier (1986) para 3.1.3-36
5.1.5	Smith and Mosier (1986) para 3.1.3-20
5.1.6.1	Ziegler and Fähnrich (1988) p. 130
5.1.6.2	Ziegler and Fähnrich (1988) p. 130
5.1.7	Ziegler and Fähnrich (1988) p. 130
5.1.8.1	Ziegler and Fähnrich (1988) p. 129
5.1.8.2	Ziegler and Fähnrich (1988) p. 129
5.1.8.3	Ziegler and Fähnrich (1988) p. 129
5.2.1.1	Lickteig (1989) p. 13, 30 Appendix A p. A-1; Bailey (1982) p. 346; Chao (1986) p. 15
5.2.1.2	Smith and Mosier (1986) para 3.1.3-19; Shneiderman (1988) p. 702.
5.2.1.3	Smith and Mosier (1986) para 3.1.3-22 and 4.4-3; Shneiderman (1988) p.702; Paap and Roske-Hofstrand (1986) p. 384
5.2.1.4	Smith and Mosier (1986) para 3.1.3-23; Lickteig (1989) p. 15; Shneiderman (1988) p. 702
5.2.1.5	Galitz (1984) p. 120
5.2.1.6	Galitz (1984) p. 120; Chao (1986) p. 16

REFERENCE LIST (Cont.)

Paragraph	References
5.2.1.7	Williams et al. (1987a) Appendix p. A-3
5.2.1.8	Smith and Mosier (1986) para 3.1.3-17 and 3.1.3-18; Sidorsky (1984) para 5.1-13
5.2.1.9	Smith and Mosier (1986) para 3.1.3-3
5.2.1.10	Shneiderman (1987) p. 100
5.3.1a	Smith and Mosier (1986) para 3.1.3-25; Nielsen (1987) p. 384; Lickteig (1989) p. 13; Ziegler and Fähnrich (1988) p. 129
5.3.1b	Sidorsky (1984) para 5.1-14; Chao (1986) p. 15
5.3.2.1	Bowser (1991) p. 9; Smith and Mosier (1986) para 4.4-4
5.3.2.2	Smith and Mosier (1986) para 3.1.3-28; Sidorsky (1984) para 5.1-14; Galitz (1984) p. 119
5.3.2.3	Smith and Mosier (1986) para 3.1.3-30; DoD (1989a) p. 267; Chao (1986) p. 15; Shneiderman (1987) p. 115
5.3.2.4	Smith and Mosier (1986) para 3.1.3-32
5.3.2.5	Bowser (1991) p. 9

REFERENCE LIST (Cont.)

Paragraph	References
5.3.3.1	Smith and Mosier (1986) para 3.1.3-26, 4.4-2, 3.2-2; Shneiderman (1988) p. 702; Galitz (1984) p. 120; Sidorsky (1984) para 1.1-12, 3.2-21
5.3.3.2	Smith and Mosier (1986) para 3.2-3
5.3.3.3	Smith and Mosier (1986) para 3.1.3-34; Galitz (1984) p. 120; DoD (1989a) p. 266
5.3.3.4	Bowser (1991) p. 10; Smith and Mosier (1986) para 3.1.3-33; Sidorsky (1984) para 5.1-14; DoD (1989a) p. 267; Chao (1986) p.16
5.3.3.5	Smith and Mosier (1986) para 3.1.3-31; Galitz (1984) p. 120
5.3.3.6	Ziegler and Fähnrich (1988) p. 129
5.3.3.7	Smith and Mosier (1986) para 3.1.3-35; Shneiderman (1988) p.702; Shneiderman (1987) p. 118; Chao (1986) p. 16; Laverson and Shneiderman (1987) p. 104; Sidorsky (1984) para 5.1-15
5.3.4.1	Smith and Mosier (1986) para 3.1.3-27; Sidorsky (1984) para 5.1-14
5.3.4.2	Shneiderman (1988) p. 702; Norman and Chin (1988) p. 63
5.3.4.3	Norman and Chin (1988) p. 63

REFERENCE LIST (Cont.)

Paragraph	References
5.3.4.4	Galitz (1984) p. 120
5.4.1.1	Galitz (1984) para 3.1.3-29; Chao (1986) p. 15
5.4.1.2	Parkinson et al. (1988) p. 691
5.4.1.3	Antin (1988) p. 181
5.4.1.4	Smith and Mosier (1986) para 3.1.3-9
5.4.1.5	Smith and Mosier (1986) para 3.1.3-8; Chao (1986) p. 15
5.4.1.6	Sidorsky (1984) para 5.1-15
5.4.2.1	Smith and Mosier (1986) para 3.1.3-4; DoD (1989a) p. 266
5.4.2.2	Smith and Mosier (1986) para 3.1.3-5
5.4.2.3	Smith and Mosier (1986) para 3.1.3-6
5.5.1.1	Shneiderman (1988) p. 702
5.5.1.2	Smith and Mosier (1986) para 3.1.3-11; Sidorsky (1984) para 5.1-13
5.5.1.3	Shneiderman (1987) p. 87; Bailey (1982) p. 343
5.5.1.4	Smith and Mosier (1986) para 3.1.3-24; Sidorsky (1984) para 5.1-14; Chao (1986) p.16
5.5.1.5	Shneiderman (1987) p. 100

REFERENCE LIST (Cont.)

Paragraph	References
5.5.2.1	Laverson et al. (1987) p. 106; Shneiderman (1987) p. 117 and 118; Chao (1986) p. 16; Smith and Mosier (1986) para 3.1.3-13; Galitz (1984) p. 120
5.5.2.2	Chao (1986) p. 16
5.5.2.3	Laverson et al. (1987) p. 105
5.5.2.4	Sidorsky (1984) para 2.1-13; Chao (1986) p. 16
5.5.2.5	Smith and Mosier (1986) para 3.1.3-14; Sidorsky para 5.1-14
5.5.2.6	Smith and Mosier (1986) para 3.2-8; Shneiderman (1987) p. 115

This Page Intentionally Left Blank.

6.0 DIRECT MANIPULATION

For graphical user interfaces, a major type of interactive dialog is through direct manipulation. In a direct manipulation dialog, the user controls the interface with the computer by acting directly on "objects" on the display screen. An object may be an icon (which is a pictographic symbol that represents a computer process or data [Gittins 1986]), menu option, symbol, button, or dialog box. The user highlights the object and implements the action by the use of a pointing device, such as a mouse. Actions include moving an object, querying a database, calling up a preformatted message, or sending a message over a communications system. The result of an action is immediately observable.

The direct manipulation of a computer system is analogous to controlling a tactical vehicle. The user uses a control, such as the steering wheel, to input a command to the vehicle and is rewarded by an immediate response. With a computer, the user moves a cursor over an object, such as a symbol, and presses a button to input a command, such as querying status. The system responds immediately by displaying the status in a pop-up window next to the object. This is unlike a command-language-based system, where the user types in a command, hits ENTER, then waits for a response from the system.

Direct manipulation user interfaces are characterized by continuous representation of the object of interest, physical actions or button presses to effect computer actions, incremental reversible actions, and immediate visual feedback. These characteristics provide the user with a greater feeling of control and often lead to better performance and greater acceptance.

A reason for using direct manipulation in the user interface is the reduced time required to learn new applications. Efficiencies in learning result from using both standard, consistent actions in the application environment and metaphors to guide the user. A metaphor uses the visual nature of a direct manipulation interface to map objects in the application onto a visual representation familiar to the user. Metaphors effectively control the complexity of the user interface by making actions, procedures, and concepts similar to those already known to the user. By exploiting the user's prior knowledge, the designer permits the user to think in terms familiar to the application domain, rather than in terms of low-level computer concepts. This results in applications that are easier to learn and easier to use. Some effective examples are an accountant's worksheet in a spreadsheet application or the physical world in an object-oriented programming environment.

The following considerations should be kept in mind when designing a direct-manipulation, user-computer interface.

- Selecting the metaphor should leverage prior knowledge in a way that is specific to the user's environment. The metaphor should represent the system function in a way that is meaningful to the user. The office metaphor described above may not be appropriate for a tactical command and control system.

- The metaphor should be simple, but not so simple that it leads to underutilization of the system because it does not model all the available capability. The user will not be aware of what is not modeled.

- The metaphor used for icons and system interaction should be tested with representatives of the intended user population.

- The following hardware considerations should be incorporated for an effective direct manipulation system: 1) a high-resolution, bit-mapped display system, 2) greater central processing and memory to support the bit-mapped windowing system, and 3) rapid operating speed to provide the immediate response needed for effective user-computer interaction.

The designer should recognize that the ways in which a human perceives figures impacts how icons are designed. The icon designer needs to incorporate Gestalt principles of human perception (Lewis and Fallesen 1989), briefly described as follows:

- Humans see the simplest or most efficient interpretation of an icon. Therefore, the designer should avoid complex shapes that the user may misinterpret.

- The user will associate a meaning with an icon. The stronger the associated meaning, the more easily the icon will be recognized and remembered. Therefore, the designer should create icons that have intrinsic meaning to the user.

- Users tend to mentally group objects. Grouping can be based on proximity, similarity, arrangement of objects that define a closed region, arrangement of objects in straight or smoothly curving lines, symmetry of arrangement, and objects undergoing simultaneous, correlated changes. Grouping can provide additional meaning to icons, but the designer should ensure that unrelated icons are not inadvertently grouped by users.

- Figure-ground relationships are important to how a user perceives an icon. Figure refers to an object, which has a shape and stands out from the background. Ground refers to the area that is perceived to continue behind the figure. Keep in mind the following points regarding this topic.

  - The size of a figure relative to its background is important. The smaller the size of a figure relative to the ground, the more likely it will be perceived as a figure.

  - When shape only is used as a discriminator for figures, convex shapes will likely be perceived as figures and concave shapes as holes.

  - A contour line will be perceived as belonging to only one of the areas it delineates.

  - Position affects whether an object will be perceived as a figure. Centrally positioned objects and the lower portion of a surface divided horizontally into two parts are seen as figures.

- The greater the contrast between an object and its background, the greater the perception of the object as a figure.

The following pages provide detailed guidelines for designing the user-computer interface when direct manipulation is used. The designer should recognize that the literature has little explicit guidance for direct manipulation as it applies to military command and control systems. Therefore, it is imperative that before formalizing the user-computer interface, detailed research and careful testing of alternatives be done with users who represent the intended user population.

6.1 GENERAL

Direct manipulation of displayed objects should be provided as a means of interactive control, because command and control applications may be used by casual system users and turnover in personnel will be high.

6.1.1 Hardware Considerations

The designer should be aware of the following hardware considerations.

6.1.1.1 Display System Requirements

High resolution screens and a bit-mapped hardware architecture are required for direct manipulation systems.

6.1.1.2 Use of Positive Image

Because direct manipulation is designed to represent the actual product, a positive image (dark foreground on light background) is best, as it represents printed output. See the example in Figure 6.1.

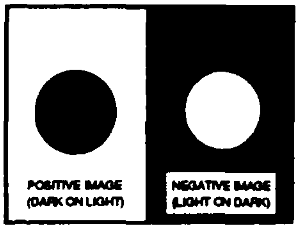


FIGURE 6.1: Example of Positive and Negative Images

Software must be flexible enough to accommodate key-board cursor keys should the pointing device fail.

6.1.1.3 Pointing Device

Direct manipulation is most efficient when a pointing device such as a mouse, trackball, or other device is used. Touch-interactive devices are discussed in Section 16.

6.1.2 Screen Arrangement by the User

The user should be able to arrange windows and icons on the screen to meet the individual task needs. However, the user should not be able to move the window or icon off the screen.

6.1.3 Function Control

Five methods should be considered for invoking a function, file, or operation with direct manipulation: Function Keys, Menu Bar, Pop-up Menus, Pull-down Menus, and Icons. Icons are discussed in Paragraph 6.3. Function keys are discussed in Section 4, and menuing is discussed in Section 5.

6.1.4 Interaction

Operator interactive tasks should use the most appropriate input mode. The keyboard is recommended for extensive alphanumeric data. Using pointing devices to select from menus is usually more effective. Where both modes are present, the software should allow both keyboard and pointing device selection of items.

6.2 METAPHORS

Metaphors associate interface objects in the application or problem domain with a visual representation familiar to the user (e.g., trash cans equal delete). They should be used carefully and tested with the user population. The following paragraphs provide guidance on metaphor design.

6.2.1 Metaphor Model

The metaphor should:

a. model the system being controlled

b. be appropriate for the user's tasks, functions and environment. The office metaphor may not necessarily be appropriate for military command and control.

6.2.2 Metaphor Consistency with Objects

Metaphors should be consistent with the objects chosen to represent the functions. For example, deleting a file with recovery capability would be represented by a trash can, whereas deleting a file permanently would be represented by a paper shredder.

6.2.3 Complex Metaphors

Avoid using complex metaphors. Complex metaphors, like complex icons, can lead to increased inferences of meaning and errors by the user.

Oversimplification occurs when the metaphor does not model all the capability available to the user, causing under-utilization of the system.

6.2.4 Metaphor Oversimplification

Although metaphors should be as simple as possible, avoid oversimplification.

6.2.5 Metaphors Versus Self-Contained Icons

If effective self-contained symbols (icons) can be designed for information presentation, use them over multiple icons of a complex metaphor.

6.2.6 Generalization of Metaphors

Metaphors should be general enough to allow the user to understand and use other metaphors or media, such as text-based systems.

6.2.7 Metaphor Tutoring

Icon metaphors should be designed to tutor the user towards a more complete understanding of the underlying functional system.

6.2.8 Connotations Induced by Metaphors

Metaphors, especially those used by more than one cultural or national group (e.g., NATO forces), should be developed carefully to ensure they do not have a negative connotation for the user. See also Paragraph 6.3.9.3.

6.3 ICONS

Icons are pictographic symbols that represent objects, processes, or data. Icons are the most visible manifestation of a metaphor, as described in the introduction to Section 6. The following paragraphs provide guidelines for designing icons. However, it must be stressed that more research needs to be performed to arrive at a complete set of icons for military command and control.

The designer should consider the display media when designing the icon (e.g., an icon design for a high-resolution display may lose meaning when presented on a low-resolution display).

6.3.1 Use of Icons

When using direct manipulation, use icons as visual representations of system functions that can be manipulated by the user.

6.3.2 Iconic Menus

Iconic menus are groups of icons that act the same as textual menus, allowing selection of system options. Figure 6.2 illustrates an iconic menu.

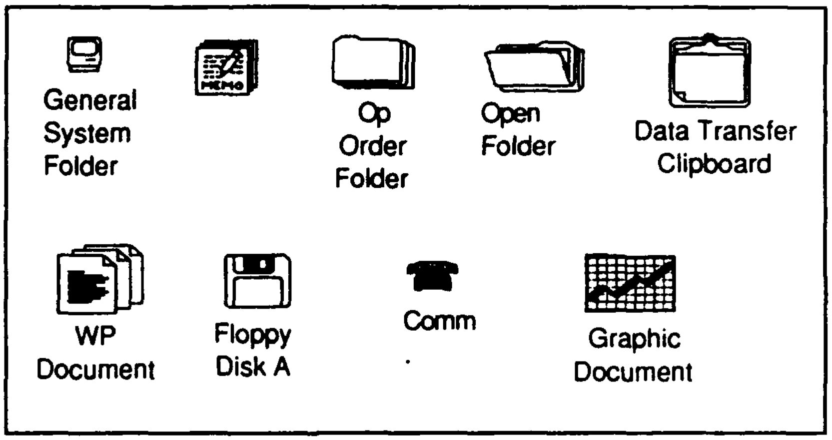


FIGURE 6.2: Example of an Iconic Menu

6.3.2.1 Use of Icons for the Multilingual User

When a user will be multilingual (e.g., NATO Forces), devise iconic menus for control functions.

6.3.2.2 Icon Menu Option Defeat

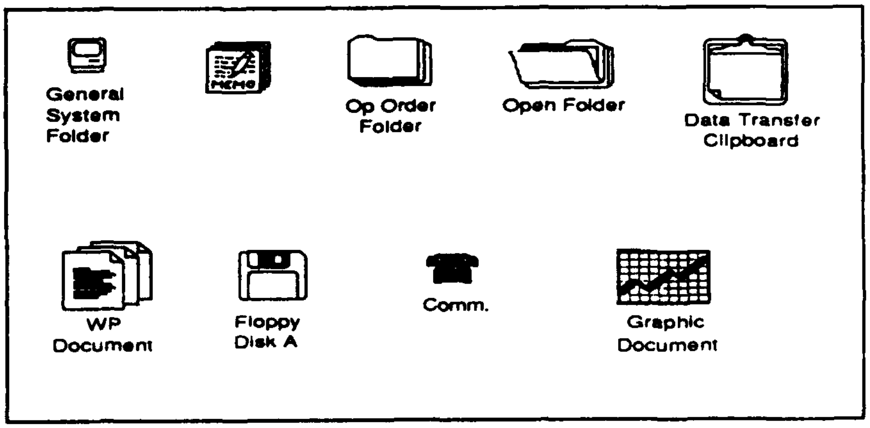
When using iconic menus, design the system such that once an action has been initiated through an icon (e.g., printing), nonselectable icons cannot be manipulated. Provide the user with a visual indication of which icons are unavailable.

6.3.3 Icon Mapping to Functions

When an icon is used to represent more than one function (i.e., mapped), do not repeat the icon for each function. Selection of the icon should cause a menu to appear that allows the user to select the specific function to be performed. For example, an icon for selecting communication devices, when selected, would cause a menu listing each type of communication device to appear. The user would then select the device to be used.

6.3.4 Switching to Textual Representation

Include a feature that allows the user, when working with icons, to switch to a textual representation of the functions or files. The text should be listed sequentially to produce a logical transition from icon to text. See Figure 6.3.



Communications	10k	Folder	August 1, 1990
Data Transfer Clipboard	15k	Clipboard	July 10, 1990
General system folder	50k	Startup Folder	Jun 1, 1990
Graphic Document	30k	Graphic Document	August 12, 1990
Op Order Folder	120k	Folder	March 23, 1990
Open Folder	112k	Folder	August 23, 1990
WP Document	35k	Word Processing Doc.	July 30, 1990

FIGURE 6.3: Example of Textual Representation

6.3.5 Testing Icon Design

It is imperative to test icon design with a group of users who represent the intended user. This should ensure the meaning of the icon is implicitly understood.

6.3.6 Using Color-Coding in Icons

Use caution when color-coding, and use color only if it is redundant to another coding method. See Section 11, Color Usage, for additional color guidelines.

6.3.7 Consistency of Icon Design

6.3.7.1 Consistent Icon Meaning

Icon meaning should be consistent across displays and standardized within an application. To the extent possible, it should also be standardized and consistent across all command and control systems.

6.3.7.2 Common Primitives and Boundaries

As feasible, use a common set of primitives (code that defines a specific shape, form, or color) and boundaries for icons. This will improve the user's ability to recognize and associate icons with their meanings.

6.3.7.3 Consistency of Interaction

Interaction with icons should be consistent with other system dialogs, especially those used for windows. For example, if the user can layer and expand windows, the user should be able to do the same with icons.

6.3.8 Icon Labeling

6.3.8.1 Additional Labeling of Icons

An icon shape generally represents a class of items or functions. To distinguish the precise function, provide an additional label that names each icon.

6.3.8.2 Icon Label Placement

Icon labels should be placed underneath the icon, as illustrated in Figure 6.4. If labels are not used, the user should be able to query the system to get a definition of the icon.

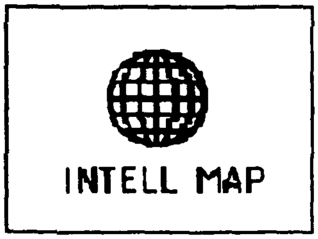


FIGURE 6.4: Example of a Text Title for an Icon

6.3.9 Icon Shape

6.3.9.1 Designing Icon Shape

Design icon shapes to provide a visual representation that matches the user's expectations and allows an association between the icon and the function being controlled. With respect to the user, icon shape should be a concrete, not abstract, concept.

6.3.9.2 Simplicity of Icon Design

Icon shapes should be as simple as possible to ensure user recognition. If the icon shape is too complex, the user may make errors in recognizing the icon. Icon shape should show or exaggerate an object's natural features. See Figure 6.5.

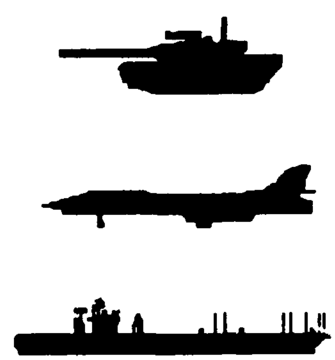


FIGURE 6.5: Examples of Icon Shapes

6.3.9.3 Design for Different Cultures and Nationalities

If icons are to be used with different cultural or national groups (e.g., NATO Forces), use technological shapes or forms rather than natural objects. The examples shown in Figure 6.6 include areas shaped as circles, triangles, or boundaries shaped as lines with pluses (+) as end marks.

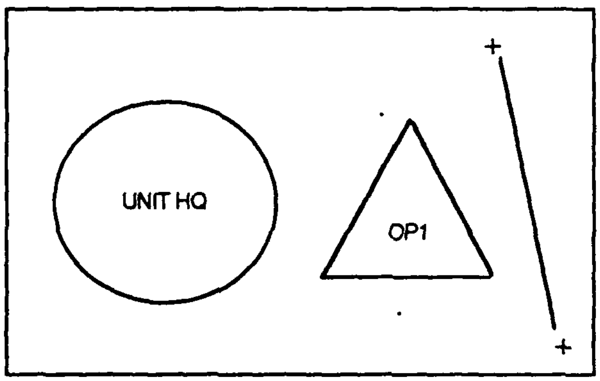


FIGURE 6.6: Example of the Use of Technological Shapes

6.3.9.4 Number of Unique Icon Shapes

The fewer unique icon shapes used, the more effective the user-computer interface. At a maximum, no more than 20 unique shapes should be used.

6.3.9.5 Boundary Lines

Icon boundary lines should be solid and closed and should have a high contrast value. Corners should be smooth. The best boundary is based on the contrast between the figure and the underlying display background.

6.3.9.6 Open Boundaries

If a boundary is left open, be aware that the user will tend mentally to close the open boundary. Design the opening to ensure this closure will not occur.

6.3.9.7 Icon Figure-Ground Form

Icon figure-ground (foreground lines, etc.) should be clear and stable.

6.3.9.8 Depiction of Opposite Functions

When icons represent opposite functions, design the icons so they mirror one another. See Figure 6.7.

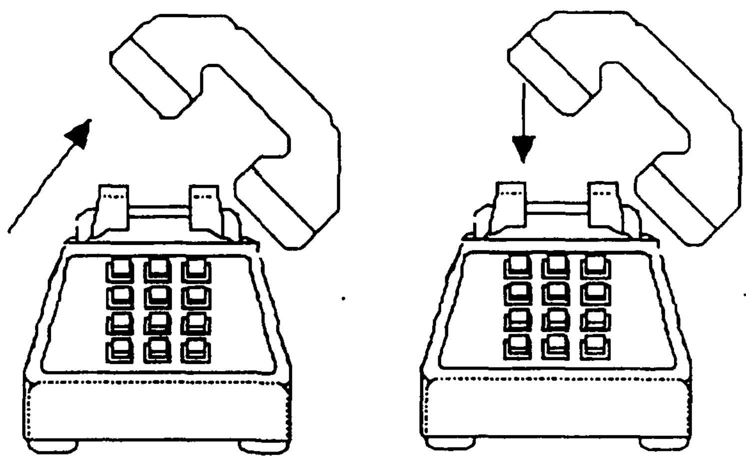


FIGURE 6.7: Example of Mirrored Icons

6.3.9.9 Three-Dimensional Icons

If three-dimensional icons are used, allow the user to manipulate them through rotation so they can be viewed from different vertical and horizontal viewing angles.

6.3.9.10 Icon Highlighting

When an icon is selected, it should be highlighted.

6.3.10 Icon Size

6.3.10.1 Size of Icons

Icons should be at least 1/4 inch in height to reduce the time required for positioning the cursor on the target and performing the required controlling actions.

6.3.10.2 Size Coding of Icons

Up to five sizes of icon can be used for coding, but no more than three are recommended for command and control systems.

6.3.10.3 Icon Size When Used for Coding

When size coding of icons is used by command and control systems, the larger icon should be 1.5 times as large as the next smallest.

6.3.10.4 Icon Figure-Ground Size Ratio

Icons should have a size ratio to the background of 1:1.5.

This Page Intentionally Left Blank

REFERENCE LIST

Paragraph	References
6.0	Helander (1988)
6.1	Smith and Mosier (1986) para 3.1.8-5
6.1.1.1	Ziegler and Fähnrich (1988) p. 127
6.1.1.2	Ziegler and Fähnrich (1988) p. 127
6.1.1.3	Ziegler and Fähnrich (1988) p. 128
6.1.2	Ziegler and Fähnrich (1988) p. 128
6.1.3	Ziegler and Fähnrich (1988) p. 129
6.1.4	Bowser (1991)
6.2	Lewis and Fallesen (1989) p. 91
6.2.1	OSF (1990) p. 531; Lewis and Fallesen (1989) p. 92
6.2.2	Lewis and Fallesen (1989) p. 91
6.2.3	Gittins (1986) p. 528
6.2.4	OSF (1990) p. 531
6.2.5	Lewis and Fallesen (1989) p. 91; Gittins (1986) p. 540
6.2.6	Lewis and Fallesen (1989) p. 92; Gittins (1986) pp. 530 and 540
6.2.7	OSF (1990) p. 530
6.2.8	Lewis and Fallesen (1989) p. 91; Gittins (1986) p. 540
6.3.1	Ziegler and Fähnrich (1988) p. 127
6.3.2.1	Smith and Mosier (1986) para 3.1.8-5
6.3.2.2	Gittins (1986) p. 526
6.3.3	Gittins (1986) p. 540
6.3.4	Ziegler and Fähnrich (1988) p. 128
6.3.5	Smith and Mosier (1986) para 2.4-13

REFERENCE LIST (Cont.)

Paragraph	References
6.3.6	Lewis and Fallesen (1989) p. 91; Gittins (1986) p. 539
6.3.7.1	Lewis and Fallesen (1989) p. 89
6.3.7.2	Lewis and Fallesen (1989) p. 89
6.3.7.3	Bullinger et al. (1987) p. 90; Gittins (1986) p. 536
6.3.8.1	Ziegler and Fähnrich (1988) p. 128
6.3.8.2	Lewis and Fallesen (1989) p. 91
6.3.9	Ziegler and Fähnrich (1988) p. 128; Smith and Mosier (1986) para 2.4-13; Shneiderman (1987) p. 200; Lewis and Fallesen (1989) p. 89; Lansdale (1988) p. 148
6.3.9.2	Ziegler and Fähnrich (1988) p. 128; Shneiderman (1987) p. 200; Lewis and Fallesen (1989) p. 89; Gittins (1986) p. 528
6.3.9.3	Lewis and Fallesen (1989) p. 91; Gittins (1986) p. 539
6.3.9.4	Lewis and Fallesen (1989) p. 91
6.3.9.5	Lewis and Fallesen (1989) p. 90; Gittins (1986) p. 539
6.3.9.6	Gittins (1986) p. 539
6.3.9.7	Gittins (1986) p. 539
6.3.9.8	Lewis and Fallesen (1989) p. 90
6.3.9.9	Lewis and Fallesen (1989) p. 92

REFERENCE LIST (Cont.)

Paragraph	References
6.3.10.1	Lewis and Fallesen (1989) p. 90
6.3.10.2	Lewis and Fallesen (1989) p. 90
6.3.10.3	Lewis and Fallesen (1989) p. 90
6.3.10.4	Lewis and Fallesen (1989) p. 90

This Page Intentionally Left Blank

7.0 WINDOWING

Resizing windows is performed through "hooking" the edge or corner and dragging the cursor to reduce or increase the window size, or by using buttons typically located in the upper right corner of a window. See Figure 7.1.

A window provides the visual means by which the user can interact with an application program. Results of the command or data input by keyboard, mouse, or other device will be displayed in the window. A window display screen is analogous to a window in a wall that allows one to see into a room; the window display screen allows the user to see into a software program. Typically, a window is rectangular and can cover part or all of a display screen. In addition, multiple windows can be open at one time on a display. Figure 7.1 illustrates a typical window. The arrows and labels identify key parts of the window.

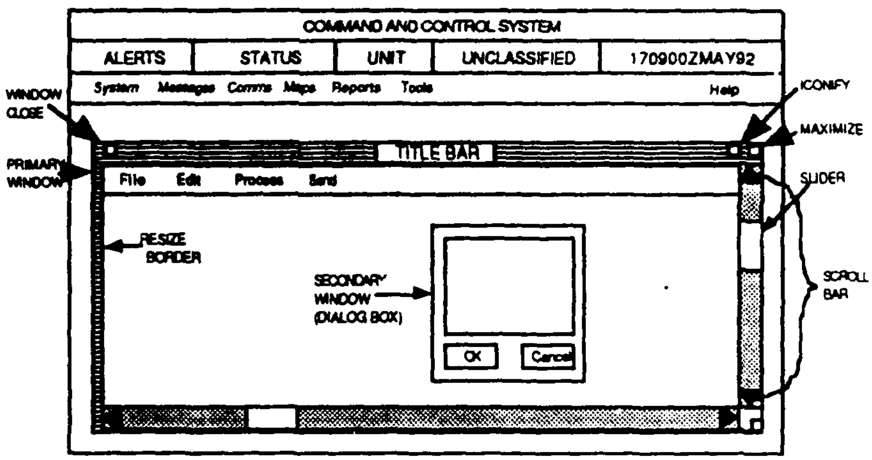


FIGURE 7.1: Example of a Typical Windowing Screen

The user controls window operation using a number of basic functions. By opening and closing a window, a task or application is started, stopped, or removed from the screen. Scrolling allows the user to view

"Moving" refers to the user's ability to reposition windows on the screen.

"Designation" refers to the means of identifying the active window.

the information within a window, including that which is outside the normal boundaries of the window. Windows can be stacked on top of each other like paper on a desk. Good designs should provide the user the capability to reshuffle and move windows relative to each other (BRING-TO-FRONT AND PUSH-TO-BACK commands).

The two basic approaches to simultaneous window presentation are tiling and overlapping. In the tiling approach, multiple windows do not overlap but lie on the same plane. Their borders are flush, and the primary control operations that can be applied to each window are opening and closing, designation, scrolling, moving, and sizing. Figure 7.2 illustrates this type of window design.

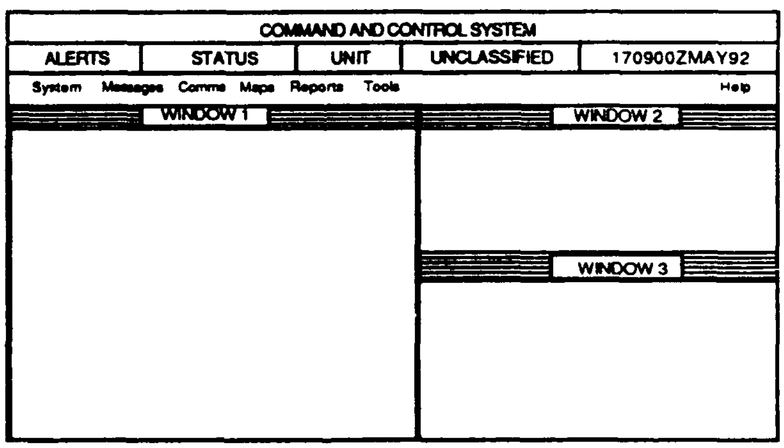


FIGURE 7.2: Example of the Tiling Approach

Using the overlapping method, windows are presented on multiple planes and appear to be three-dimensional. Windows can overlap or even obscure each other, like pieces of paper on a desk top. The user can apply all of the control functions previously discussed to overlapping windows. Figure 7.3 illustrates this type of window design.

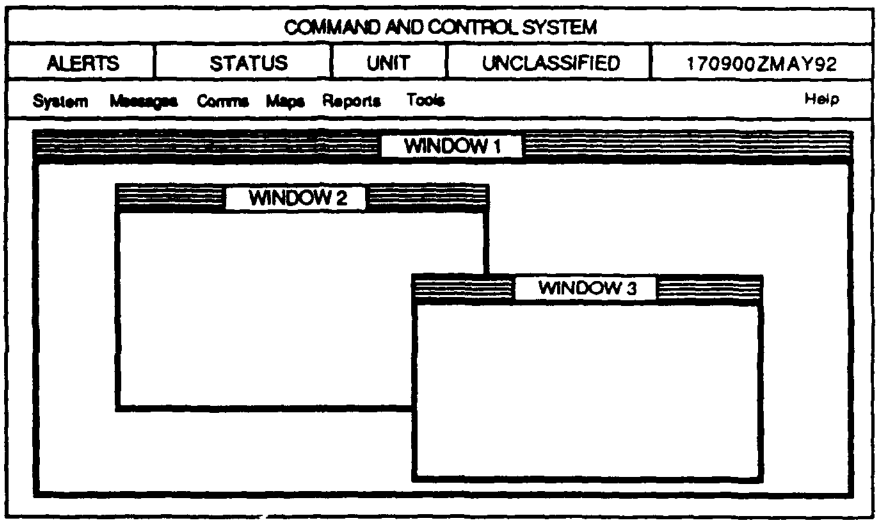


FIGURE 7.3: Example of the Overlapping Approach

One difficulty in developing design guidelines for windowing is the lack of information, primarily because the design phenomenon is so recent. The depth and breadth of research on how windows impact user performance are not as great as they are for other design areas (Billingsley 1988). In general, the following generic guidelines can be applied to the design of windowing.

- Be consistent in how the windows look and "act."

- Ensure the display device has the resolution and size to support properly a windows approach to information presentation.

- Ensure the central processor has the power, in terms of memory and speed, to utilize effectively a windows approach. Without proper hardware, the speed of information presentation will be significantly degraded by slow system-response time.

- Use windows if the user needs to perform multiple tasks or see different sets of data concurrently.

This section provides more detailed guidance for designing windows for command and control systems.

7.1 GENERAL GUIDANCE

7.1.1 Hardware Limitations on the Use of Windowing

The designer must recognize the limitations that the hardware imposes on the usefulness of windowing software.

Windowing should be avoided when the hardware has the following limitations:

a. small screen size, resulting in frequent manipulation of the screen by the user

b. slow processing speed, resulting in slow operation of tactical functions performed by the computer

c. low screen resolution, resulting in less effective visual coding, especially for tactical map graphics symbols and icons.

7.1.2 Flexibility of Window Specification

A key to the effective design of a windowing user-computer interface is the flexibility the user has in customizing window content and format. A balance must be achieved between user-specified windows and preformatted windows.

7.1.2.1 User-Specified Windows

a. When the need to view several different types of data jointly cannot be determined in advance, allow a user to specify and select separate data windows that will share a single display frame.

b. Where the information required for decision making may vary according to the tactical situation, allow the user to specify what information to include in a display.

7.1.2.2 Predefined Windows

a. When content of particular tactical displays can be determined during interface design, provide the user with preformatted windows, such as standard message texts for data entry and display.

b. Allow the user to display several of these windows concurrently, according to the tactical need.

7.1.3 Temporary Window Overlays

7.1.3.1 Use of Temporary Window Overlays

Temporary window overlays are especially effective for providing a menu of alternatives for field entry in preformatted tactical messages and database queries.

When it is necessary temporarily to add requested data or other features to a current display, provide window overlays for that purpose.

7.1.3.2 Masking the Active Window

Ensure a temporary window overlay does not completely cover the active window, thereby obscuring critical control information and command entry widgets, soft keys, or other activation points.

7.1.3.3 Nondestructive Overlay

Each open window requires system resources in terms of memory and processing speed. A limit on the maximum number of windows that can be effectively opened for each system should be determined through experimentation.

When a window overlay temporarily obscures other displayed data, ensure the obscured data are not permanently erased but will reappear when the overlay is later removed.

7.1.4 Number of Allowable Open Windows

To ensure system response time is not compromised, design into the command and control system a defined upper limit on the number of windows allowed to be open at one time.

7.1.5 Window Physical Design

7.1.5.1 Reducing Visual Clutter

Avoid visual clutter in designing windowing systems.

7.1.5.2 Tiled Windows

For tiled window systems, minimize the clutter at the edges caused by scroll bars, etc. See Figure 7.4. Figure 7.2 illustrates an uncluttered display.

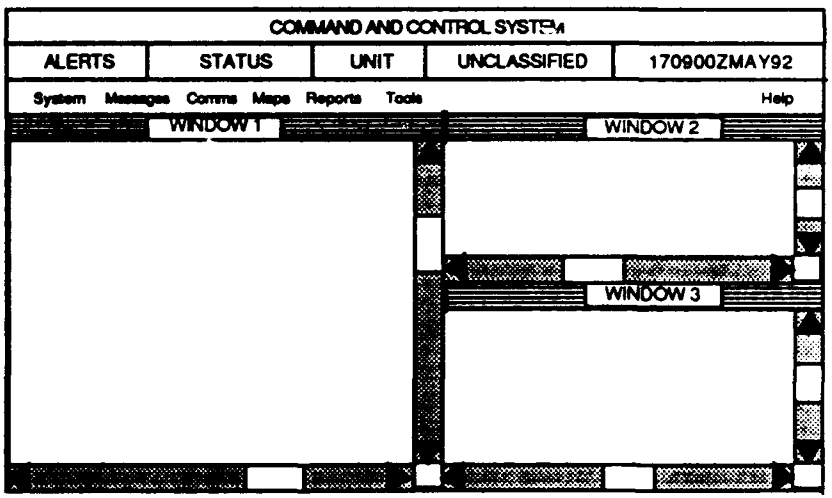


FIGURE 7.4: Example of a Cluttered Window Design for Tiled Windows

7.1.5.3 Overlapping Windows

For overlapping window systems with multiple windows, keep the background pattern neutral, rather than using complex patterns. See Figure 7.5 for an example of a cluttered display. Figure 7.3 illustrates an uncluttered display.

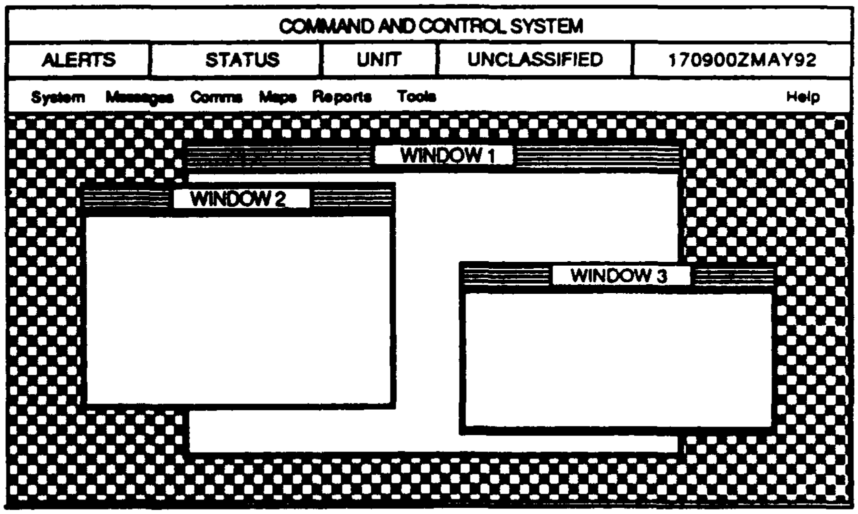


FIGURE 7.5: Example of Cluttered Window Design for Overlapping Windows Induced by a Complex Background Pattern

7.1.5.4 Adequate Window Size

When a display window must be used for scanning data that exceed more than one line, ensure the window can display more than one line of data.

7.1.5.5 Command Entry, Prompts, Messages at Bottom

When the system provides an area within a window for command entry, messages, or prompts, this area should be placed at the bottom of the window display.

7.1.5.6 Dialog Box Design

Dialog boxes should be designed to look and function consistently for all applications and systems. To achieve this, use the following recommendations (see Figure 7.6):

a. Control buttons used to input a command from a dialog box should be located at the bottom of the window, making it consistent with the user's natural task flow.

b. The button used to input the selected or default command (usually an OK) should be located on the left side of the box. The CANCEL button should be located on the right side. Any additional control buttons should be located between the OK and CANCEL buttons.

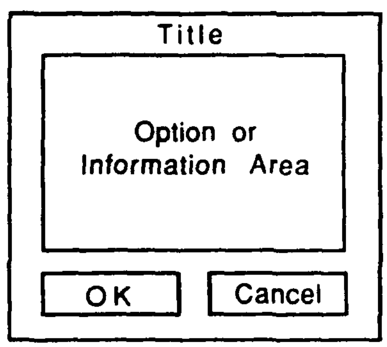


FIGURE 7.6: Example of How a Dialog Box Should Be Designed

7.2 WINDOW CONTROL

Control refers to how the user manipulates the window, not how the application operates within the window. Guidelines for the design of window control fall into five basic topics: general guidelines, opening and closing, moving, sizing, and scrolling. These are discussed as follows.

7.2.1 General

7.2.1.1 Consistent Control Within Windows

When a user may perform control actions (such as command entry) while working within a window, ensure those control actions will be consistent from one window to another.

7.2.1.2 Consistent Window Control

Ensure the means provided to the user for controlling the size, location, and characteristics of window overlays operate consistently from one display to another for each type of overlay.

7.2.1.3 Easy Suppression of Window Overlays

Provide an easy means, such as iconization or closing, for the user to suppress the display of window overlays.

7.2.1.4 Menu Bars for Multitasking Windows

Where different applications are operating concurrently in open windows (e.g., multitasking), provide a separate menu bar for each application window. See the example in Figure 7.7.

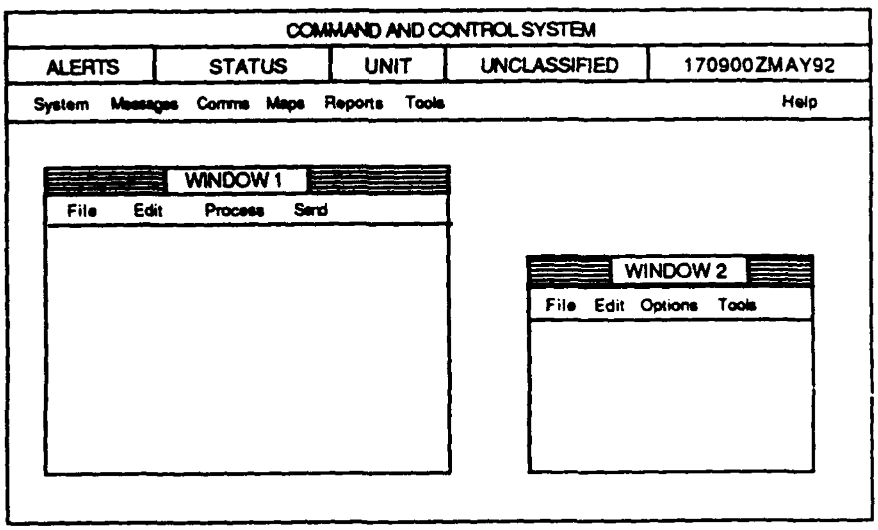


FIGURE 7.7: Example of Applications Menu With Menu Bars

7.2.2 Opening and Closing Windows

Windows are opened and closed by icons, menu selections, or a close-button widget (a small, push-button control object usually located in the upper left corner of a window). When designing the opening and closing operations, consider the following guidelines.

7.2.2.1 Animation Showing Icon Opening/Closing

The software should provide an animated depiction of the opening and closing of a window by portraying the window shrinking to an icon and vice versa. This helps the user relate the window, icon, and action. See Figure 7.8.

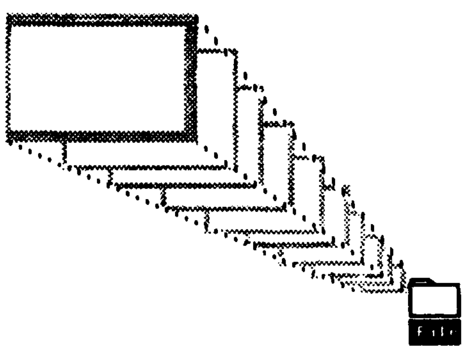


FIGURE 7.8: Example of Window Animation

If the primary window is minimized into an icon, the subordinate windows should be temporarily closed and reopened when the window is reopened.

7.2.2.2 Closure of All Subordinate Windows

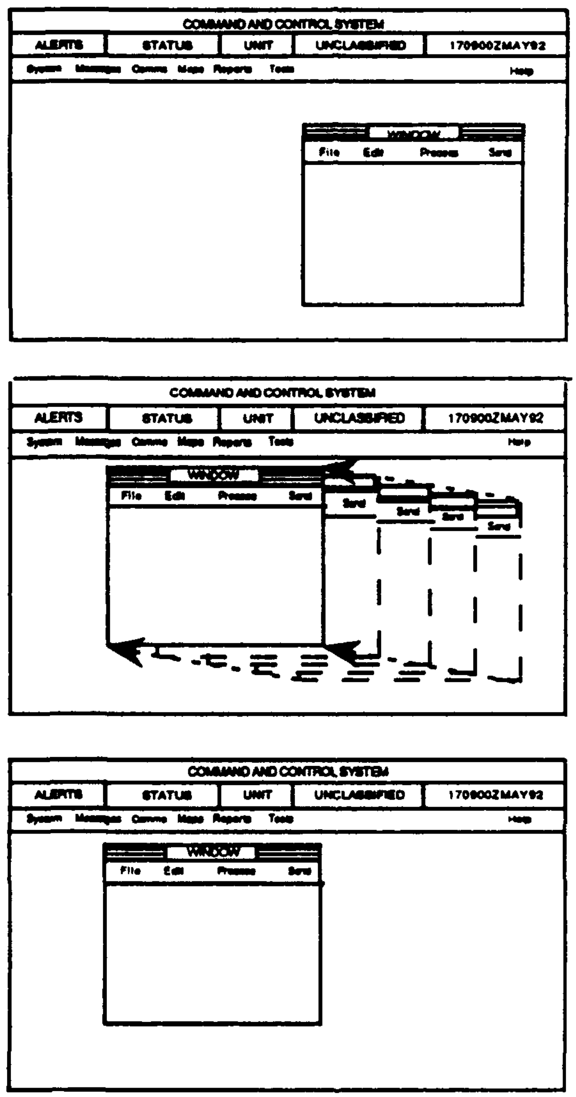
When a main applications window is closed by the user, all associated subordinate windows and dialog boxes should also close.

7.2.3 Moving Windows

Moving windows can be performed by either of two methods: 1) by "hooking" the title banner and dragging the window, or 2) by selecting a move option from a special menu, designating which specific window is to be moved, and dragging the window to its new position.

7.2.3.1 Window Movement Feedback

Do not indicate window movement by an outline only. Provide either full movement of the window (see Figure 7.9) or move an outline, leaving the window visible on the screen.



Original Location

Movement Indication

Final Location

FIGURE 7.9: Example of a Screen Move

7.2.3.2 Cursor Shape for Moving Window

When the user must select a specific move function to relocate a window on a screen, ensure the cursor indicates this by a change in shape. Figure 7.10 illustrates one type of cursor change.

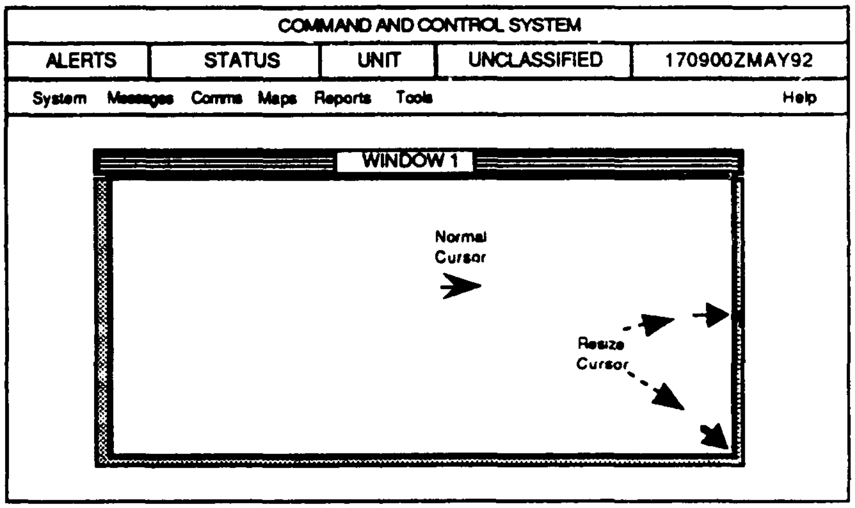


FIGURE 7.10: Example of a Cursor Changing Shape

7.2.4 Resizing Windows

7.2.4.1 System Protection

Provide system protection against obscuring critical control information during window manipulation, especially during user maximization of the window. This means system protection for both the data being retrieved through dialog boxes and system-level control information, such as alert indications. See Figures 7.11 and 7.12.

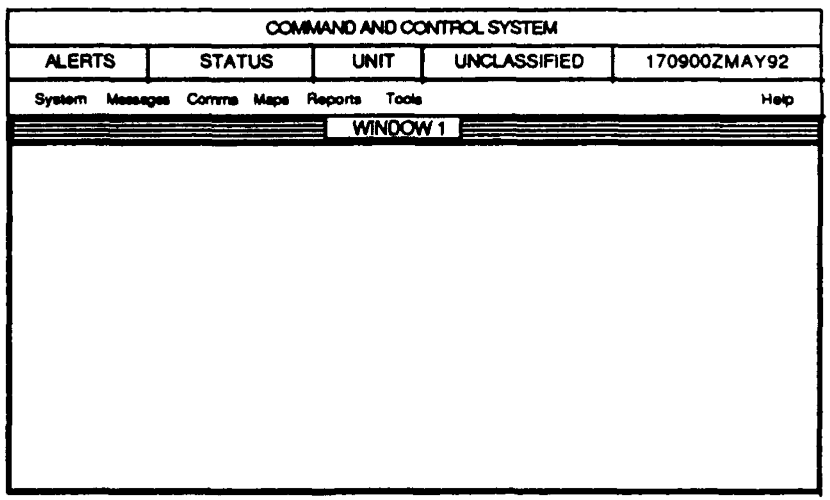


FIGURE 7.11: Maximum Size of Window

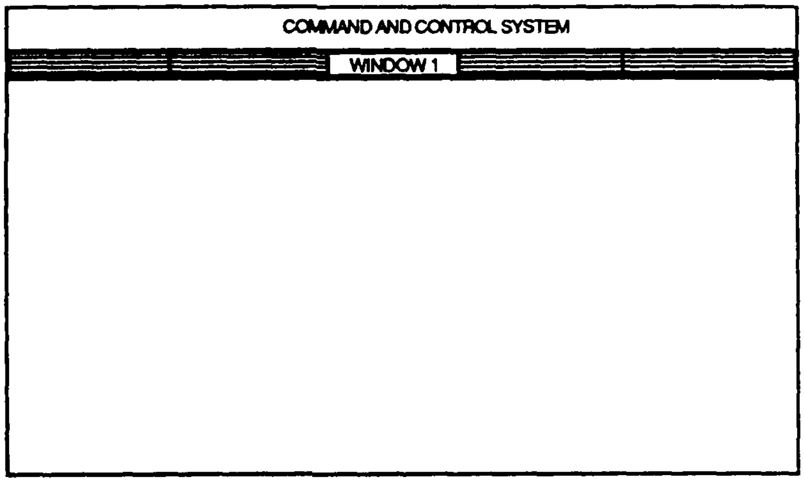


FIGURE 7.12: Window Size Too Large, Covering Critical Information

7.2.4.2 Visual Indication of Resizing

Keeping the contents visible will reduce the number of steps required by the user (e.g., resize, view, etc.).

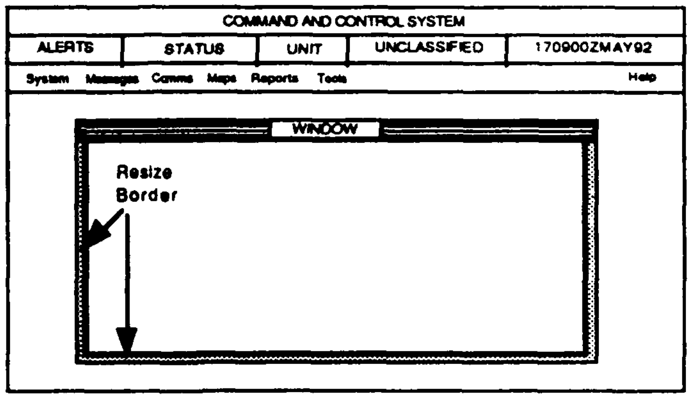
When a window is resized, ensure the window contents remain visible during the resizing to provide a visual indication of the effect on the window contents (e.g., visibility and integrity of the image), rather than providing just an outline.

7.2.4.3 Tiled Window Resizing

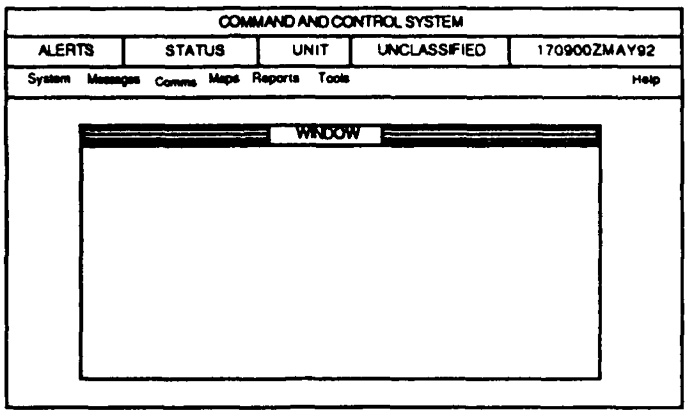
Resizing of tiled windows by a user is not recommended. If resizing is absolutely required for a tiled window system, ensure the system automatically resizes all other open windows when one is resized by the user.

7.2.4.4 Removing Resize Border

Most windows have a resize border (see Figure 7.13) located at the peripheral edge. If a window cannot be resized, this resize border should be removed to provide a positive indication to the user that the window size is static.



Window with a Resize Border



Window with Resize Border Removed

FIGURE 7.13: Resize Border Removal

7.2.5 Scrolling Windows

Scrolling a window can be performed two ways: 1) Move the window over the data, where upward movement of the scroll bar causes data to appear to move down. 2) Move the data past the window, where upward movement of the scroll bar causes data to appear to move up.

7.2.5.1 Scrolling Indication, Window Movement Versus Data Movement

For scrolling, the system should move the window over the data, as this is consistent with the general convention in industry and the OSF/Motif Style Guide.

7.2.5.2 Scroll Bar Slider Size

The distance the slider moves on a scroll bar should be proportional to the distance traveled through the file in a window to assist the user in determining current location relative to the total file.

7.2.5.3 Excessive Scrolling

Design window displays to preclude excessive scrolling. If possible, use a single screen for the full display, unless it causes reading difficulty due to reduction of screen character size.

7.3 DESIGNATION

Designation is the process of selecting and indicating with visual cues which window the user can use. This window is called the active window.

7.3.1 Positive Indication of the Active Window

When more than one window is open, provide a clear, positive indication to the user of the active window by means of a more complex border, subtle change in color hue, or change in labeling. This active window should be distinct, yet not distract the user's attention from window activity. See Figure 7.14 for an example.

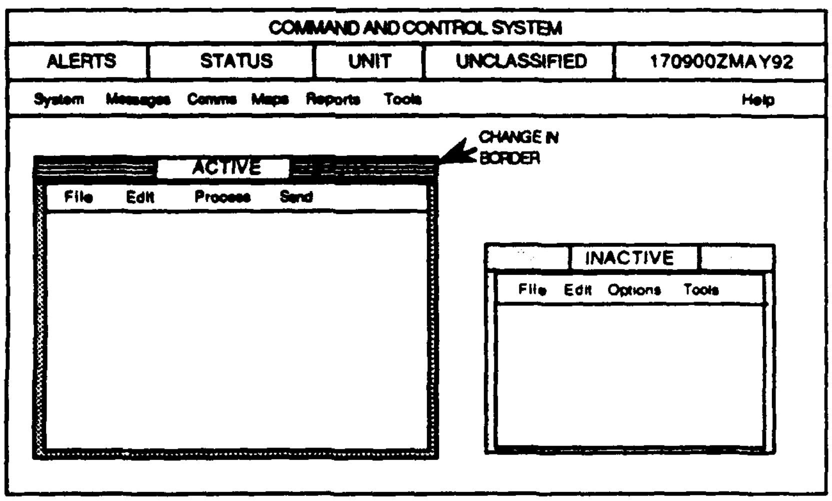


FIGURE 7.14: Example of Active Window Designation

7.3.2 Easy Shifting Among Windows

If several window overlays are displayed at once, provide some easy means for the user to shift among them to select which window will be currently active. For example, shift the cursor with the mouse, then press the mouse button to designate the active window.

7.4 LABELING

7.4.1 Labeling Windows

Window overlays, dialog boxes, or subordinate windows should be assigned an identifying label. This label should describe briefly the contents, purpose of the window, or the menu path. For example, Standard Message Formats.

7.4.2 Window Label Location

Window titles should be located at the top of the window display, with at least one blank line between the title and the body of the window.

7.4.3 Format of Subordinate Window Labels

Titles of subordinate windows should match the menu selection items from the supraordinate window menu.

7.4.4 Window Titles

Location of window titles should be consistent.

Navigation, in terms of windows, refers to the user's ability to move among the various windows that are open on a display.

When an overlapping system is used, the user may have windows open that obscure others. This can lead to disorder and confusion as to the number of windows open.

7.5 OPEN WINDOW NAVIGATION

7.5.1 Open Window Map

Applications should, when using an overlapping window structure, provide a user-requested iconic or text map/indication of all open windows to allow the user to easily identify all open (especially the hidden) windows. Figure 7.15 shows three presentations of an open window map.

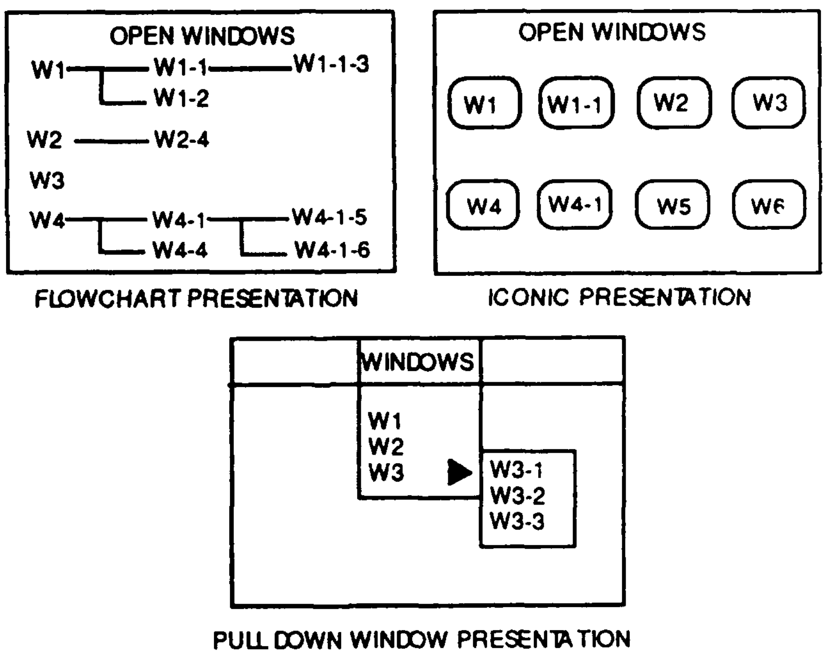


FIGURE 7.15: Examples of Open Window Maps

7.5.2 Active Designation from Open Window Map

Provide the user the capability to designate the active window through the iconic or text open window map by highlighting the window representation.

7.5.3 Expanded Window Explanation of Open Window Map

If possible, allow the user to query an open window map for expanded information (e.g., date created, size, description of subject or application, etc.) on the file or application operating in the window.

7.5.4 Window Forward Function With Window Map

When an iconic or text map is provided for determining the numbers and names of open windows in an overlapping system, allow the user to bring a window forward from the map without having to resize or move other windows.

This Page Intentionally Left Blank.

REFERENCE LIST

Paragraph	References
7.1.1	Billingsley (1988) p. 416
7.1.2.1	Avery et al. (1990); Smith and Mosier (1986) paras 2.7.5-3 and 2.5.8
7.1.2.2	Avery et al. (1990); Smith and Mosier (1986) para 2.7.5-2
7.1.3.1	Smith and Mosier (1986) para 2.7.5-1
7.1.3.2	Slominski and Young (1988); Avery et al. (1990)
7.1.3.3	Smith and Mosier (1986) para 2.7.5-10
7.1.4	Billingsley (1988) p. 421
7.1.5.1	Billingsley (1988) p. 421
7.1.5.2	Smith and Mosier (1986) para 2.5-9
7.1.5.3	Bowser (1991) p 12; Smith and Mosier (1986) para 2.5-11
7.1.5.4	OSF (1990) Chapter 7; Avery et al. (1990)
7.2.1.1	Smith and Mosier (1986) para 2.7.5-9
7.2.1.2	Smith and Mosier (1986) para 2.7.5-4
7.2.1.3	Smith and Mosier (1986) para 2.7.5-5
7.2.1.4	Nielson (1987) p. 247
7.2.2.1	Billingsley (1988) p. 428
7.2.2.2	OSF (1990) p. 3-4
7.2.3.1	Billingsley (1988) p. 428
7.2.3.2	Billingsley (1988) p. 428
7.2.4.1	Slominski and Young (1988); Avery et al. (1990)
7.2.4.2	Billingsley (1988) p. 429

REFERENCE LIST (Cont.)

Paragraph	References
7.2.4.3	Billingsley (1988) p. 421; OSF (1990) p. 4-5, Avery et al. (1990)
7.2.4.4	OSF (1990) p. 3-6
7.2.5.1	Billingsley (1988) p. 430; OSF (1990)
7.2.5.2	Billingsley (1988) p. 430
7.2.5.3	Slominsky and Young (1988) p. 5; Avery et al. (1990)
7.3.1	Billingsley (1988) p. 431; Lewis and Fallesen (1989) p. 94; Smith and Mosier (1986) paras 2.7.5-7
7.3.2	Smith and Mosier (1986) para 2.7.5-8
7.4.1	Smith and Mosier (1986) para 2.7.5-6 and 2.5-10, OSF (1990) p. 3-4
7.4.2	Smith and Mosier (1986) para 2.5-10
7.4.3	Slominski and Young (1988) p. 2, 54 p. 3-4
7.4.4	Slominski and Young (1988) p. 2
7.5.1	Billingsley (1988) p. 420
7.5.2	Billingsley (1988) p. 431
7.5.3	Billingsley (1988) p. 420
7.5.4	Billingsley (1988) p. 420

8.0 MAPS AND SITUATION DISPLAYS

Graphical presentation of data is a critical feature of many emerging command and control applications. This section provides guidelines for presenting data in graphical formats. The applications discussed here include tactical graphics (overlays, symbology, and terrain representation) and pictographic representations (digitized maps, pictures, etc.). Guidelines pertaining to graphical characteristics of the user interface (e.g., screen design, windows, icons, buttons, etc.) are presented in other sections of this document.

Most of the guidelines presented in this section were obtained from Lewis and Fallesen (1989) and Smith and Mosier (1986), who included information gathered from relevant Military Standards and other key documents. Additional guideline materials were obtained through literature reviews.

8.0 Map and Situation Displays - Introduction

The designer of map graphic displays should be aware of the following overarching guidelines that are relevant to electronic map displays:

- The design of maps, including the use of symbology, should be consistent with the user's expectations.
- The level of detail should be consistent with the operational need. Too much or too little detail limits the usefulness of the map.
- Map graphics should have tools built in that allow the user to move easily around the map, to include zooming, panning, insets, registration, and keys for scale.

8.1 GENERAL

Maps refer to projected representations of geographic data, usually on flat surface displays. Maps include both natural and man-made features and text and/or graphics and colors used to describe or code those features. Situation displays provide a means of relating changing conditions or events to geographic features represented on maps. Figure 8.1 illustrates a typical map graphic display.

8.1.1 Curvature

Be consistent in projecting the earth's curvature on flat surface maps when displaying large geographic areas.

8.1.2 Situation Display Presentation

Provide a means of presenting situation displays as overlays on related map backgrounds.

8.1.3 Map Label Position

Position map labels consistently (e.g., beneath or within the feature). Where possible, label all significant features without cluttering the display.

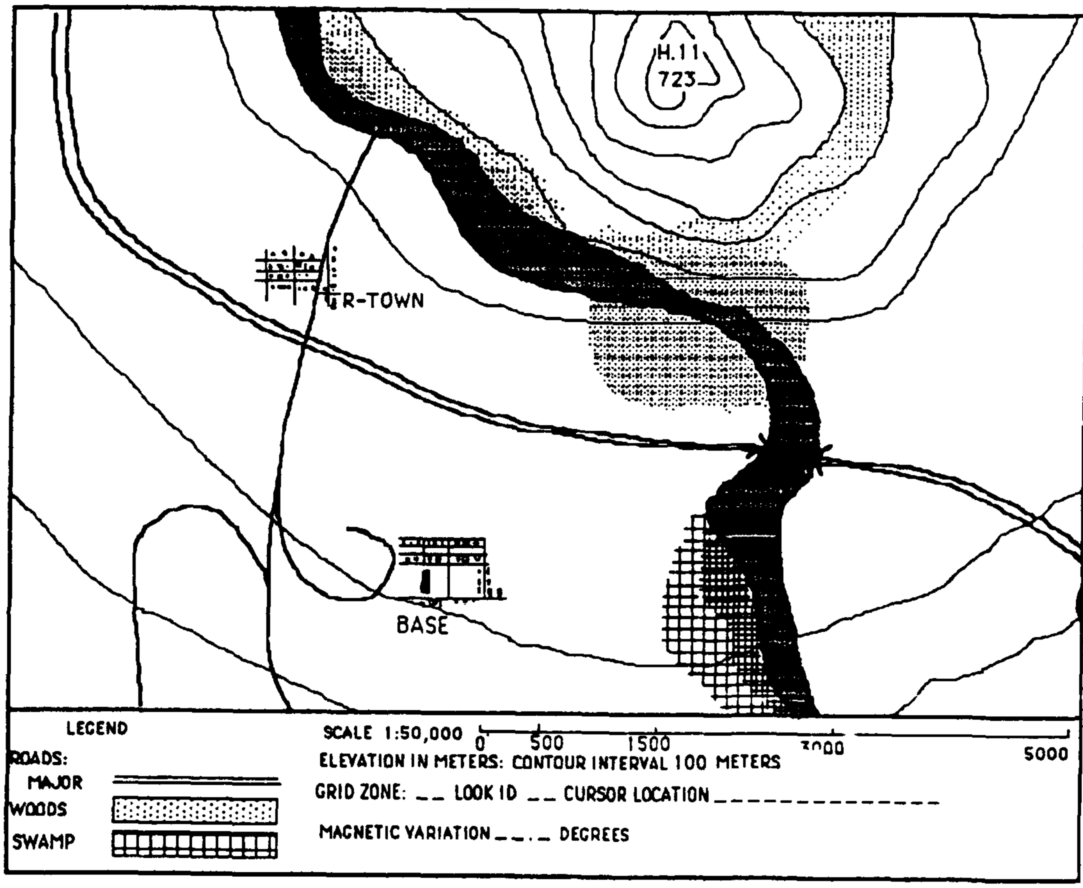


FIGURE 8.1: Typical Electronic Map, in Black and White

8.1.4 Map Orientation

Use a consistent map orientation when more than one map will be displayed (e.g., north consistent for all maps). See Figure 8.2.

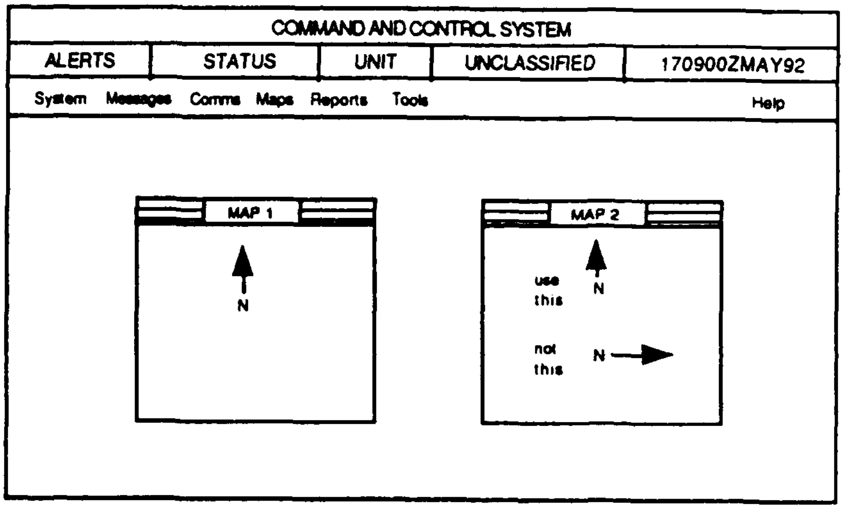


FIGURE 8.2: Example of Consistent Map Orientation

8.1.5 Designating Map Areas

Consider using color, shading, texture patterns, or highlighting to define map areas of special interest. Shades (tones) of a single color are preferable to multiple colors when observers must make relative comparisons between or among areas. When using shades of color or texture patterns, the gradation of shades from dark to light should correspond to variation in the variable that is represented (see Paragraph 11.3).

8.1.6 Automated Tools

Provide automated tools for complex map analyses. The specific tools should be based upon the user's needs. For example, avenue of approach, line-of-sight, and trafficability are needed by some but not all users. The user requirements should be determined and appropriate tools provided.

8.2 STATIC DISPLAY ATTRIBUTES

8.2.1 Coverage Area and Resolution

As a minimum, maps must cover the areas of responsibility of the user at each organizational level and provide all essential details required to conduct operations. Map displays should be large enough to permit the simultaneous presentation and visual integration of information required by the user. Small electronic displays may be panned and zoomed (see Section 8.3) to increase map coverage. However, at present, such displays have significant visual limitations when compared to traditional, large-format, paper maps.

8.2.1.1 Map Feature Representation

All critical map features must be represented.

8.2.1.2 Map Label Legibility

Labels must remain legible at all display resolutions.

8.2.1.3 Reduction of Clutter

Provide a means for reducing clutter while preserving essential information.

8.2.1.4 Area of View on Maps

Maneuver commanders at each echelon should be able to view their own areas of operation, activities one echelon above and two echelons below, and activities of friendly adjacent (flanking) units. The activities of adjacent and deep enemy units that oppose displayed friendly forces should also be displayed. See Figure 8.3.

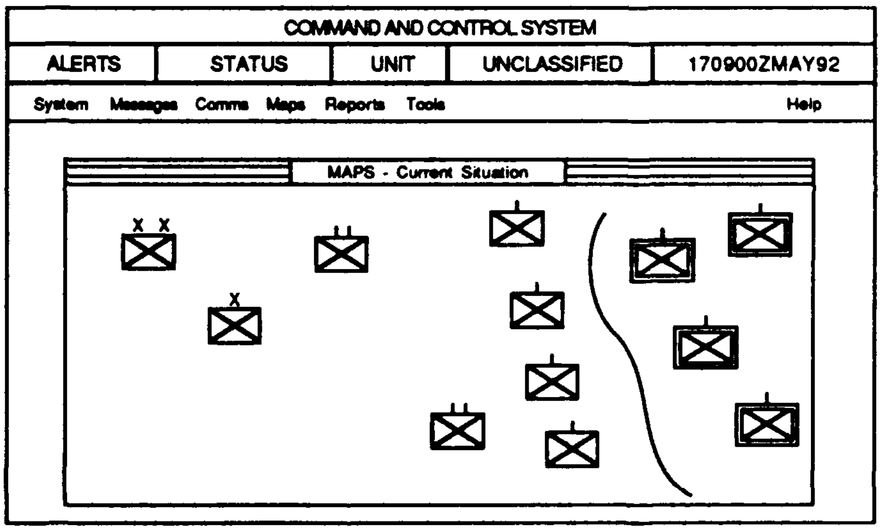


FIGURE 8.3: Example of How a Brigade's Map Display Allows Viewing of One Echelon Higher and Two Echelons Lower

8.2.2 Accuracy of Location

8.2.2.1 Connecting Symbols to Location

Symbols should be accurately placed on the map or connected to the desired location using arrows, lines, or other pointing devices.

8.2.2.2 Automatic Registration

Provide an automated means of registering graphic data with background map information at all display scales.

8.2.3 Symbology

8.2.3.1 Standard Military Symbols

Use standard military symbols in accordance with doctrine when preparing maps and overlays. For example, use the current edition of FM 101-5-1, Operational Terms and Symbols.

8.2.3.2 Symbol Identification Key

Provide a means by which the user may obtain help in identifying unknown symbols or other map information. For example, the user could highlight a symbol and query its meaning through a context-sensitive help feature.

8.2.3.3 Symbol Color Coding

Use standard military map color codes and provide a user-prompted key defining the color codes which are used. See Section 11.

8.2.3.4 Overlap of Symbols

Map symbols should not be allowed to overlap, particularly if this would obscure their identity. Where overlap is unavoidable, provide a means for moving background symbols to the foreground or otherwise revealing masked symbols.

8.2.3.5 Symbol Labeling

Essential labels (for example, unit identification) should be displayed with the symbol; otherwise, provide a means by which the user can display information related to selected symbols. Figure 8.4 illustrates how a user could query a symbol for more detail.

8.2 Map and Situation Displays -Static Display Attributes

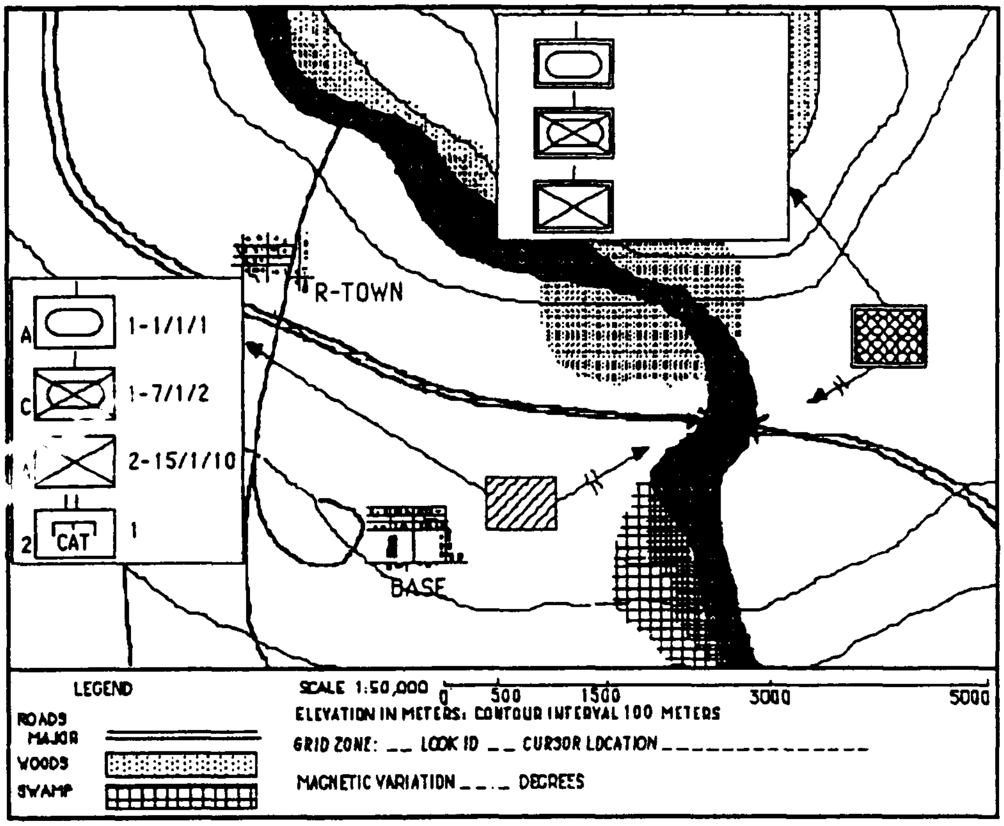


FIGURE 8.4: Querying a Summary Symbol for Detailed Information

8.2.3.6 Auxiliary Alphanumeric Labels

Consider the auxiliary use of alphanumeric coding where graphic data are not already so labeled.

8.2.3.7 Symbol Label Position

Position symbol labels consistently in accordance with doctrinal guidance.

8.2.3.8 Terrain Representation

Digital terrain data available for some versions of electronic map (e-map) allow alternative methods of portraying terrain features. In addition to traditional topographic contour intervals, digital terrain data can present map backgrounds depicting road networks, drainage, vegetation, and soil type. Shading, coloring, or other visual cues can also be used to accentuate terrain features.

8.2.4 Location of Displayed Section

Where location information is frequently used, a constantly visible display of coordinates associated with the cursor should be displayed in user-selectable coordinate units that can also be conveniently changed. The continuous display of location should be augmented with the capability to fix (point on the map) a location to facilitate moving overlay displays.

8.2.4.1 Availability of Symbol/Map Feature Coordinates

Provide to the user a means of obtaining the exact map coordinates for a selected symbol or map feature by means of querying the symbol or feature. The recommended method of querying an item is to use a pointing device, such as a mouse or trackball cursor.

8.2.4.2 Larger Map Inset

When the entire map is not displayed, provide an inset that shows where the displayed portion is within the larger map. See Figure 8.5.

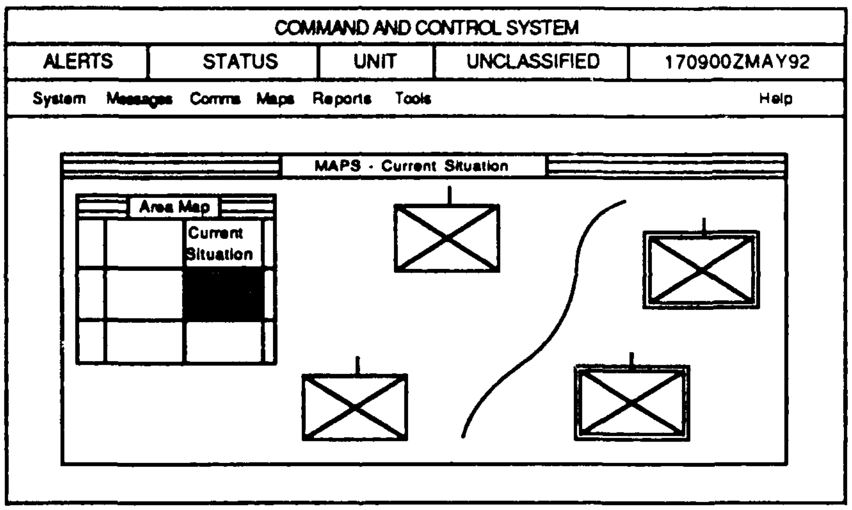


FIGURE 8.5: Example of a Map Inset

8.2.4.3 Distance Determination

Provide an automated means for readily determining the distance between points.

8.2.4.4 Bearing Determination

Provide a means for readily determining the bearing between points.

8.3 DYNAMIC CHARACTERISTICS

Because of the limited screen size of many displays, a method is needed to scan and change the scales of the maps. In addition, changes in the tactical situation require updates to various map overlays. The following guidelines should be considered when implementing dynamically changing maps.

8.3.1 Panning

8.3.1.1 Use of Panning

Permit the user to change the displayed area by moving a window over the map in any direction. Panning operations may be continuous (preferable) or discrete but should meet the user's requirements.

8.3.1.2 Position Indicator for Panning

During panning operations, provide an indicator of position in the overall display.

8.3.1.3 Return to Start Point

During panning operations, provide a means for rapidly returning to the starting point.

8.3.2 Zooming

8.3.2.1 Use of Zooming

Provide a means for moving away from or toward the displayed area (zooming) to obtain a larger view or greater detail.

8.3.2.2 Zooming and Readability

Zooming must not cause problems in reading symbols, labels, or other map features.

8.3.2.3 Variable Level of Detail

Consider modifying the level of detail (number of symbols and features depicted) to match the degree of zooming used (i.e., more detail for close-up views and less for large-area perspectives).

When zooming, symbols should be collapsed into fewer summary symbols to declutter.

8.3.2.4 Method of Zooming

Of the two methods of zooming (i.e., continuous and discrete), continuous is preferable. Whichever method is used must be satisfactory to the user.

8.3.2.5 Return to Default

Provide a means for quickly returning to the normal display size when zooming.

8.3.2.6 Indication of Changing Scales

When changing scales through zooming, provide an indicator that continually shows the appropriate scale.

8.3.2.7 Expanded Sector Position Indication

It is recommended that an inset or window be provided that shows the maximum available map coverage. An example of map coverage (Figure 8.5) would be a graphic square on the inset map that indicates the position of the map currently displayed. In the most useful form, this inset would be interactive and used to set parameters for calling up a screen map display.

8.3.3 Automatic Updating

Automatic updating, editing, and distributing map data are among the primary advantages offered by electronic displays. The following guidelines address considerations in implementing these capabilities.

8.3.3.1 Selecting Information for Update

As appropriate, allow the user to select categories of information that will be automatically updated.

8.3.3.2 Stable Reference Elements

Provide stable reference elements (e.g., terrain features, boundaries, etc.) when displays are automatically updated.

8.3.3.3 Identification of Updates

Provide a means for readily identifying updates or changes. Critical changes must be easily recognized and distinguishable from other changes to the display. For example, highlight the update until the user acknowledges it.

8.3.3.4 Control of Update Frequency

The user should be permitted to control how often the display is updated and should be able to freeze the display to prevent further updates.

8.3.3.5 Update Rate

The rate of display update should match the perceptual abilities of the observer to permit successful visual integration of the changing patterns.

8.3.3.6 Availability of Update Freeze

Permit the user to freeze the display to prevent further updates. Provide a warning while the automatic display updating is suspended and when resuming automatic updating. Provide an option to either resume at the current time or at the time updating was suspended.

8.3.4 Sequencing

Display sequencing refers to two practices: 1) selectively presenting and removing displayed data, such as a series of overlays with different information. This can act as an aid for decluttering a display. 2) illustrating temporal changes in the information of historical data or simulation of future events.

Display sequencing may be used to reduce clutter (e.g., presenting map overlays in succession), to reproduce temporal changes in the display database (e.g., changes in the tactical situation), and to aid in visualizing simulated changes in the battlefield situation.

8.3.4.1 Rate of Sequencing Control

Where possible, allow the user to control the rate of sequencing.

8.3.4.2 Sequencing Pause or Suspend

Provide a capability to pause or suspend sequencing operations and provide an indicator of the status of sequencing operations..

8.3.4.3 Forward and Reverse Sequencing

As appropriate, allow the user to present sequenced displays in forward or reverse order.

8.3.4.4 Return to a Specific Display in a Sequence

Provide a means for the user to return quickly to a selected display within a sequence of displays.

8.3.4.5 Use of Animation in Sequencing

Consider using animation as an aid to the pictorial display for complex objects.

8.3.5 Grid Overlay

Provide a user-selectable grid overlay that is keyed to the coordinate system of the map. It should be easy for the user to turn the grid on and off. Coordinate keying of the overlays must be clearly specified and easily operated by the user.

8.3.6 Dynamic Map Legend

The map display should have an associated window giving relevant information in a continuous display. The information should include map scale, cursor location, graphic of map coverage, and status (i.e, "working," "computing," "available," etc.).

8.4 CREATING AND EDITING MAP GRAPHICS

8.4.1 Standard Symbol Library

Provide a library of standard symbols and a means of transferring and manipulating symbols.

8.4.2 Labeling Symbols

Provide an easy means of labeling symbols. Consider automated means of aiding the user in labeling and enforcing labeling conventions.

8.4.3 Building Symbols and Overlays

Provide automated tools to assist the user in constructing new symbols and graphics overlays.

8.4.4 Printing Preview

When preparing graphics displays for printing, allow users to preview displays as they will appear when printed.

8.4.5 Display Editing

8.4.5.1 Addition and Deletion

The user should be able to add or delete symbols, labels, or other features without destroying background information.

8.4.5.2 Area Expansion for Data Placement

Allow the user to expand an area of the display as required for accurate placement of critical data.

8.4.5.3 Graphic Element Designation

Provide a means for designating graphic elements for editing. Highlight selected items to provide a visual cue of forthcoming subsequent actions.

8.4.5.4 Repositioning Elements

Allow the user to reposition selected elements on the display.

8.4.5.5 Remove/Restore Elements

Allow the user to remove and restore selected elements.

8.4.5.6 Selection from Existing Options

Allow the user to select from displays of available options when making changes to display attributes, such as color, symbols, line types, textures, etc. Selection should be made by pointing rather than by naming the options.

8.4.5.7 Attribute Identification

Provide an easy means for the user to identify attributes currently selected.

8.4.5.8 Attribute Change

The user should be able to change the attributes of selected graphic elements.

8.4.5.9 Storage of Graphic Display

Provide an easy means for naming, storing, and retrieving graphics displays and elements. Also, provide a means for reviewing and selecting from stored graphics files.

8.5 MAP DISPLAY CHARACTERISTICS

8.5.1 Map as a Base Screen

When an application is map intensive, it is recommended that the map be used as the background or base screen, which should be the maximum display size possible to promote readability.

8.5.2 Map Readability

It is beneficial to ensure the readability of map features, since the map is the focus of the user. The screen design should avoid displays that cover the map when possible, and windows should not obscure the map.

8.5.3 Map Cursors

Map cursors should use a crosshair design that has high contrast with the background. It is recommended that cursor size subtend 20 minutes of visual angle so the average user can easily locate it on the map.

8.5.4 Graphic Overlays

The preselection or filtering of graphic overlays is a recommended feature. The decluttering of graphic displays (especially maps) should be assisted.

8.5.4.1 Filters

Labels and titles used for filters should be carefully reviewed to ensure items are understandable. The filters should be extended to map features, such as roads, cities, vegetation, topography, and political data. The intensity of the map should be controllable to allow fadeout of the map without losing all the map features.

8.5.4.2 Labeling of Graphic Overlays

It is understood that graphic overlays will overlap map features, but text information should not be obscured. The text should be offset with arrows to preserve map legibility.

8.5.5 Color Use with Graphic Overlays

Using color to identify symbols is encouraged, but redundant coding that does not use color should also be used. This caution is especially true for friend-enemy or danger-safe designations. Dots, dashes, shapes, and video effects are recommended. Care must be taken to avoid visual color illusions caused by color blending (i.e., adjacent red and blue lines are seen as one purple line).

This Page Intentionally Left Blank.

REFERENCE LIST

Paragraphs	References
8.1.1	Smith and Mosier (1986) para 2.4.8-3
8.1.2	Smith and Mosier (1986) para 2.4.8-3
8.1.3	Smith and Mosier (1986) para 2.4.8-15
8.1.4	Smith and Mosier (1986) para 2.4.8-4 and 2.4.8-5
8.1.5	Smith and Mosier (1986) para 2.4.8-7 through 2.4.8-9
8.1.6	Smith and Mosier (1986) para 2.4.8-13
8.2.2.1	Lewis and Fallesen (1989) para 2.5.6.1.1.2
8.2.2.2	Smith and Mosier (1986) para 1.6-17
8.2.3	U.S. Department of the Army (1985b)
8.2.3.1	Lewis and Fallesen (1989) para 2.5.6.1.1.1; U.S. Department of the Army (1985b)
8.2.3.4	Lewis and Fallesen (1989) para 2.5.6.1.1.3
8.2.3.5	Lewis and Fallesen (1989) para 2.5.6.1.1.4
8.2.3.6	Smith and Mosier (1986) para 2.6-8
8.2.3.7	Lewis and Fallesen (1989) para 2.5.6.1.1.4.6; U.S. Department of the Army (1985b)
8.2.3.8	Lewis and Fallesen (1989) para 2.5.6.1.3
8.2.4	Bowser (1991) p..13

REFERENCE LIST (Cont.)

Paragraphs	References
8.2.4.1	Lewis and Fallesen (1989) para 2.5.6.1.1.2 and 2.5.6.2.5
8.2.4.3	Smith and Mosier (1986) para 2.4.8-12
8.3	Bowser (1991) p..13
8.3.1.1	Lewis and Fallesen (1989) para 2.5.6.2.3
8.3.1.2	Smith and Mosier (1986) para 2.4.8-11
8.3.2.5	Lewis and Fallesen (1989) para 2.5.6.2.4
8.3.2.6	Smith and Mosier (1986) para 2.4-16
8.3.2.7	Smith and Mosier (1986) para 2.4-17
8.3.3.2	Smith and Mosier (1986) para 2.4.8-18
8.3.3.5	Smith and Mosier (1986) para 2.7.3-4
8.3.4.5	Smith and Mosier (1986) para 2.4- 18
8.3.5	Bowser (1991) p..14
8.3.6	Bowser (1991) p..14
8.4.2	Bowser (1991) p..15
8.4.3	Smith and Mosier (1986) para 2.4- 20
8.4.4.2	Smith and Mosier (1986) para 1.6-5
8.4.4.3	Smith and Mosier (1986) para 1.6-6 and 7
8.4.4.5	Smith and Mosier (1986) para 1.6-9
8.4.4.6	Smith and Mosier (1986) para 1.6.10 and 11
8.4.4.7	Smith and Mosier (1986) para 1.6-12
8.4.4.8	Smith and Mosier (1986) para 1.6-13

REFERENCE LIST (Cont.)

Paragraphs	References
8.4.4.9	Smith and Mosier (1986) para 1.6-15 and 16
8.5.1	Bowser (1991) p. 14
8.5.2	Bowser (1991) p. 14
8.5.3	HFS (1988); Bowser (1991) p..14
8.5.4	Bowser (1991) p. 14
8.5.4.2	Bowser (1991) p..15
8.5.5	Bowser (1991) p. 15

This Page Intentionally Left Blank

9.0 PRESENTATION GRAPHICS (GRAPHS, PICTURES, AND DIAGRAMS)

The goal of presentation graphics is to communicate effectively to the user. The idea, information, or concept communicated should be clear and unambiguous when presented in a visual form, or alternate communication modes should be used. There are emerging technological capabilities that allow the direct manipulation of elements of graphic objects within an application. These capabilities should be included in applications designed to interactively create graphics. *Guidelines* will address three aspects of presentation graphics: graphs, pictures, and diagrams.

Graphs should be used where necessary to visualize relationships among two or more variables, to facilitate comparing sets of data, to aid the observer in visualizing trends in data, and to aid in extrapolating future values of the underlying data. Graphs are also useful when comparing actual data to predicted values, when comparing actual values to established limits in control processes, for representing rapidly changing data, and for interpolating values between known points. In general, graphs have advantages over tabular data in summarizing complex relationships among variables and facilitate information processing and understanding.

9.0 Presentation Graphics - Introduction

Pictures are becoming an increasingly important form of graphic presentation. The multimedia capabilities of developing computer systems have increased the availability of pictures within computer applications. The most frequent command and control picture is a map. The use of scanned maps has transferred the tactical planning focus to the computer interface. The use of pictures on computer screens must be done with great care to avoid misleading the user.

Use diagrams (schematics) when the user requires information concerning the spatial relationship among objects but does not require the level of detail required by pictures. Schematic representations can be used as an aid to understanding relationships among components of complex systems and as a means of conveying status information concerning the operation of systems and their components. They also provide a medium through which users may manipulate designs and observe consequent actions on modeled systems.

9.1 GENERAL

9.1.1 Complex Formats

Avoid complex formats, such as three-dimensional presentations and artistic embellishments (pictures, shading, colors, decorative items), which detract from the intended purpose of the graphic.

9.1.2 Clarity Preservation

Design graphics to preserve clarity when the graphics must be reproduced or reduced in size. Window sizing should be controlled so no graphic shows partial lines.

9.1.3 Appropriateness of Formats

Provide formats (presentation styles) appropriate for the user's level of training and experience. Graphics should utilize user-expected symbols.

9.1.4 Data Specific to Task

Provide only those data the user needs for a specific task.

9.1.5 Alternative Style Selection

Allow users a selection of alternative presentation styles.

9.1.6 Querying Data Elements

Provide a means by which the user can select data elements on the graph and display the associated values.

9.1.7 Graphical Versus Tabular

Consider allowing the user to select between graphical and tabular data formats.

9.1.8 Consistency

Be consistent in design, format, labels, etc. for each presentation style.

9.1.9 Labeling

Clearly label the displayed graphics.

9.2 CREATING AND EDITING

9.2.1 Computer-Aided Entry

Provide computer aids for the entry and organization of complex graphic data.

9.2.2 Data Entry Validation

Validate data entries. The validation may include comparison to a standard range and/or the use of rules for relationships among variables. The validation process should be part of the application software.

9.2.3 Data Entry Aids to Plotting

When plotting formats are known, provide templates or other data entry aids to facilitate the entry of graphic data.

9.2.4 Automated Plotting of Stored Data

Automate plotting of stored data, and provide the user with automated editing and construction capabilities.

9.2.5 Automated Production of Scales

Automate the production of scales, and/or provide the user with automated aids for scaling graphic data.

9.2.6 Lines

9.2.6.1 Automated Aids for Lines

Provide automated aids for drawing straight and curvilinear line segments.

9.2.6.2 Use of Rubberbanding

Use rubberbanding (i.e., provide a visible line that connects a starting point to the current cursor position), which can be made permanent when selected.

9.2.6.3 Automated Aid in Line Joining

Provide automated assistance in joining and intersecting line segments.

9.2.6.4 Line Segment Designation

Allow the user to identify and select line segments for moving and editing. Typically, this is done through highlighting and dragging the line.

9.2.6.5 Optional Grid References

Provide optional, adjustable, grid references to aid the user in aligning horizontal and vertical lines.

9.2.7 Rule Specification by the User

Allow the user to specify rules for attributes, relationships, and design, and have the computer apply those rules automatically during the design process. For example, straighten hand-drawn lines, adjust angles between intersecting lines, and complete details of graphic elements.

9.2.8 Computer-Aided Drawing

Provide computer-aided methods for drawing figures and a system of prompts or other means to aid the user in the design process.

9.2.9 Automatic Scale Reduction

Allow the user to edit or create drawings using a large scale, which will later automatically reduce to the desired scale.

9.2.10 Object Manipulation

Provide a basic set of capabilities to resize, copy, move, and rotate displayed objects.

9.2.11 Mirror Imaging

Provide a means of producing mirror images (reflecting) as an aid in producing symmetrical graphic displays.

9.2.12 Grouping Elements

Permit the user to select and group graphic elements that will be edited in common.

9.2.13 Area Fill Capability

Provide an automatic means of filling enclosed areas with selected attributes (e.g., color or texture).

9.2.14 Computer Models for Graphical Display Generation

Provide computer models that can generate graphical displays in response to parameters provided by the user. See Figure 9.1.

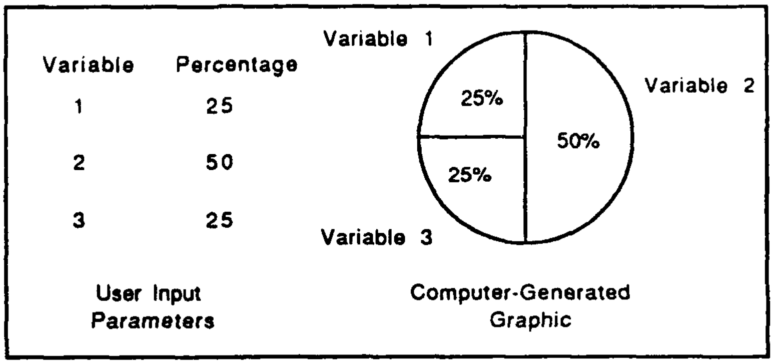


FIGURE 9.1: Example of How a Computer Model Can Generate Graphics from User Input

9.3 SCALES, LABELS, AND CODING

9.3.1 Standard Scaling Conventions

Use standard scaling conventions: values increase when moving away from the origin; independent variables (time or causal events) are plotted against the horizontal axis; and dependent variables (effects) are plotted against the vertical axis.

9.3.2 Standard Meanings

Use or assign standard meanings to graphic symbols, and apply them consistently.

9.3.3 Color and Pattern Coding

Users prefer colors to patterns for coding lines or filling areas of graphs on visual displays. Good design requires redundant coding be used. See Section 11 for color usage guidelines. Printed outputs should use texture for coding, since in most cases color will not be available.

9.3.4 Texturing Displays

If texturing must be used, use simple hatching or shading, and avoid patterns that produce visual illusions of vibration and motion. See Figure 9.2 for examples.

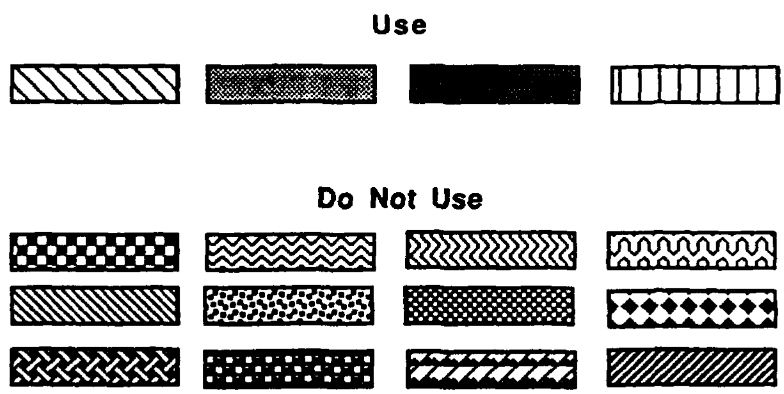


FIGURE 9.2: Texture Patterns

9.3.5 Axes Breaks in Expanded Scales

When expanding scales to emphasize a limited range of data, provide breaks in the axes to indicate discontinuities with the origin. See Figure 9.3.

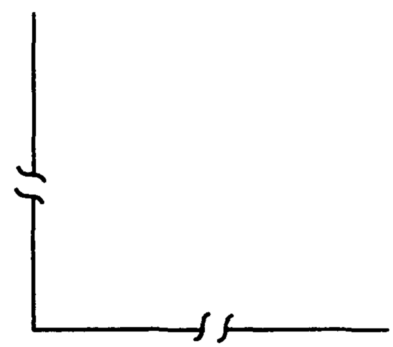


FIGURE 9.3: Example of Breaks in a Graph's Axes When Scales Have Been Expanded

9.3.6 Duplicating Axes

When scaled data contain extreme values, it may be difficult for the user to comprehend the scale values in relation to the data. To aid readability, add a copy of the X-axis at the top and a copy of the Y-axis at the right of the graph. Extreme values and data are thus in proximity throughout the graph.

9.3.7 Avoid Exaggerated Scales

Avoid the use of exaggerated scales that distort or suppress trends in the data. See Figure 9.4.

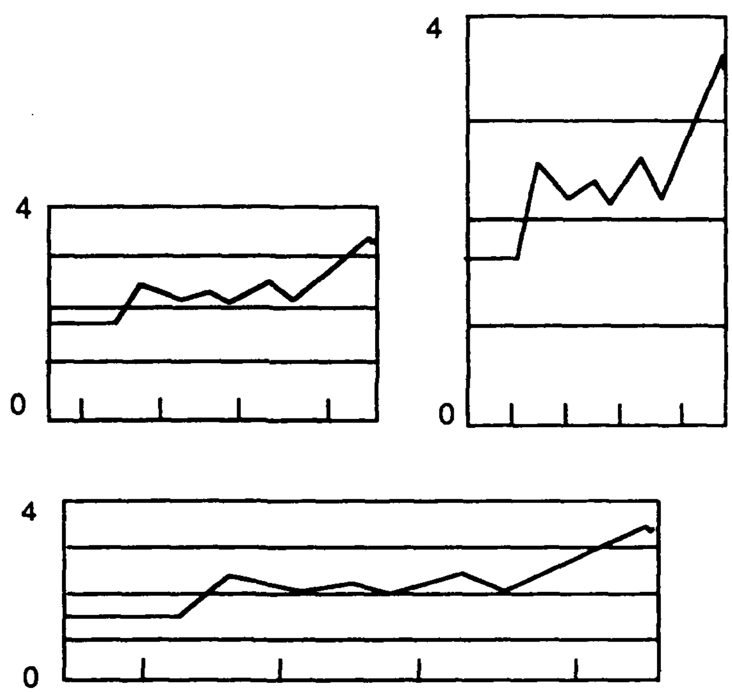


FIGURE 9.4: Example of the Distortion Induced in Data Trends by Exaggerated Scales.

9.3.8 Formats for Graphic Comparison

Provide identical formats and scales when comparisons are required between separate graphs or plot different sets of data on the same graph.

9.3.9 Using Linear Scales

Linear scales should be used in preference to non-linear scales wherever practical. For example, see Figure 9.5.

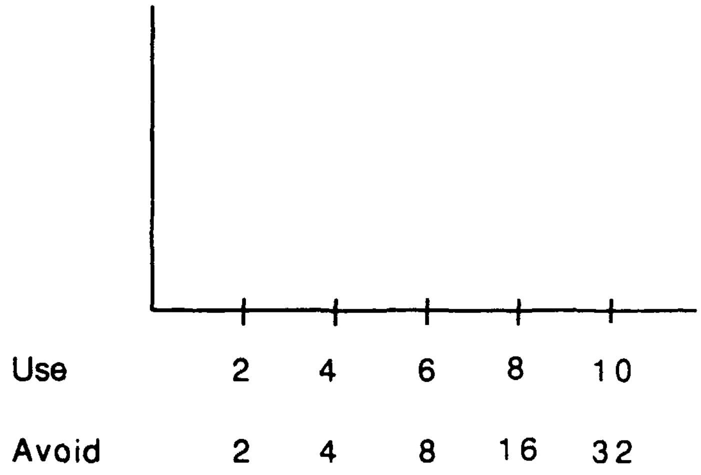


FIGURE 9.5: Example of Linear Versus Non-Linear Scales

9.3.10 Using Logarithmic Scales

Logarithmic scales may be used where comparisons of rates of change and percentages are required.

9.3.11 Multiple Entries

Avoid multiple scales on the axes of a single graph.

9.3.12 Labeling Tick Marks

Tick marks corresponding to major scale divisions on axes should be numbered or labeled, and each axis should include a label containing descriptions and units of measurement.

9.3.13 Numbering Scale Divisions

Where practical, use no more than 10-12 major scale divisions separated by up to 9 subdivisions. Whenever the appearance of the display will not be degraded, major scale divisions should be decimal multiples of whole numbers, cover the entire range of the data, and start from zero.

9.3.14 Numeric Scale Division

Numeric data scales should begin with zero when users must use displays to compare quantities or different series.

9.3.15 Label Format

Labels should use upper and lower case sans serif fonts and be oriented left-to-right for normal reading.

9.3.16 Use of Labels

Labels should be used in preference to legends or keys when it is necessary to identify plotted data elements. Labels should be oriented horizontally and located adjacent to the referenced elements. Arrows, lines, or similar pointing conventions may also be used to connect labels to their respective data elements.

9.3.17 Location of Legends and Keys

Legends or keys used to identify graphic data elements should be located within the rectangular bounds of the graph, unless such positioning would interfere with interpretation of the displayed data.

9.4 IDENTIFYING CRITICAL DATA

9.4.1 Displaying Values

Display reference or baseline values when users are required to make comparative evaluations against a fixed standard.

9.4.2 Using Supplementary Text

Consider using supplementary text to emphasize features of data requiring user attention. See the example in Figure 9.6.

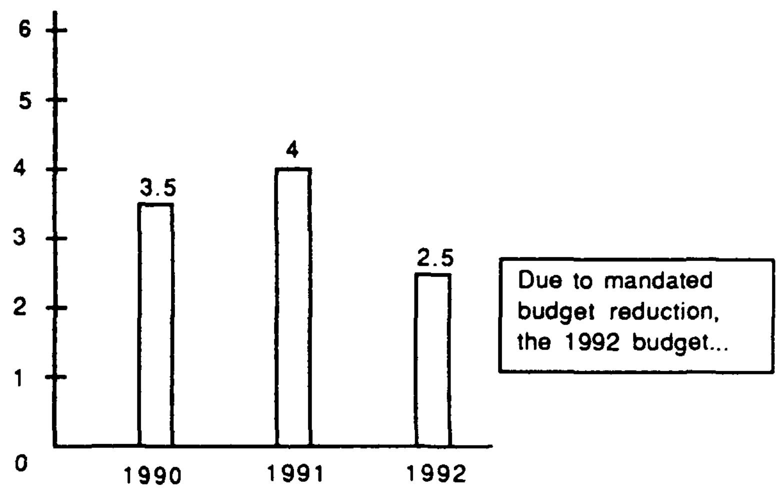


FIGURE 9.6: Example of the Use of Supplementary Text and Actual Values in a Graph

9.4.3 Displaying Data Values with Graphics

Display actual data values in addition to the graphic display where precise readings of values are required, as illustrated in Figure 9.6.

9.4.4 Position of Text Used for Labeling

When labeling graphic data, position text consistently with respect to graphic elements.

9.5 GRID LINES

A grid is the set of horizontal and vertical lines, including the labeled and scaled axes, which form a rectangular boundary around the graph. Additional horizontal and vertical grid lines corresponding to scale values partition the bounded area of the graph and provide a visual aid in locating and reading points on the graph(s). A grid and grid lines should be used, as appropriate, when presenting data graphically.

9.5.1 Grid Line Visibility

Grid lines must be easily distinguishable and should not obscure graphed data.

9.5.2 Using Grid Lines

Avoid excessive use of grid lines. Locate grid lines using the guidelines for placement of major scale values. Consider using more grid lines where greater precision is required or where the size of the display will permit their use.

9.5.3 User Display of Grid Lines

Where practical, allow the user to determine whether or not grid lines will be displayed.

9.6 TYPES OF PRESENTATION GRAPHICS

9.6.1 Curve and Line Graphs

9.6.1.1 Use of Curved and Straight Lines

Use smoothed curves or straight lines connecting data points (line graphs) when displaying relationships between two continuous variables (e.g., when showing time variation in some quantity).

9.6.1.2 Labeling Multiple Curves

When a single graph contains multiple curves, designate each curve with an adjacent label. If it is necessary to use a legend, list legend codes in the order in which curves occur in the graphs.

9.6.1.3 Highlighting Curves

When displaying multiple curves, highlight a curve containing critical data.

9.6.1.4 Using Line Coding with Curves

Use line coding to distinguish among multiple curves on the same graph, and use coding consistently when the same types of data appear on different displays.

9.6.1.5 Coding for Projecting Values Beyond Data Set

Use a distinct line code (e.g., dashed or dotted lines) when projecting values beyond the actual data set.

9.6.1.6 Cyclic Data

For cyclic data, provide at least one full cycle of data.

9.6.1.7 Plotting Difference Data

Consider plotting the difference between two series where comparisons are necessary.

9.6.2 Surface Charts

Surface charts provide a means of visualizing the relative contributions of individual elements to the sum of their individual parts, often as a function of time. Figure 9.7 uses a surface chart to illustrate how the total number of items (i.e., the sum of the numbers within each category) varies with time.

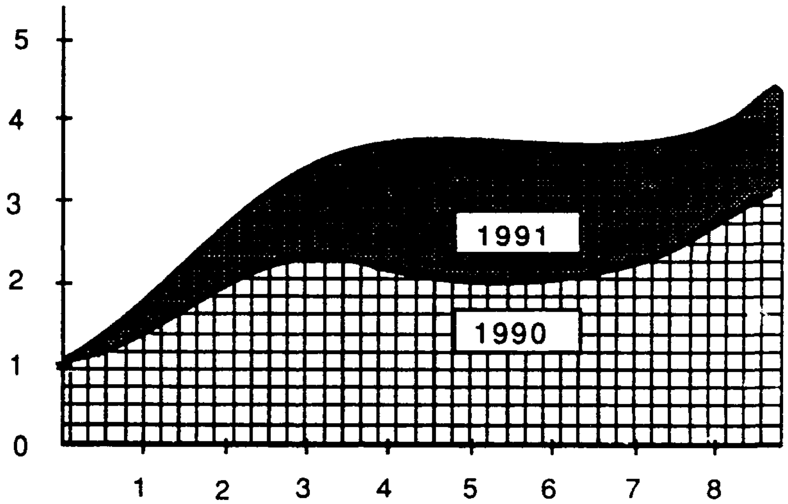


FIGURE 9.7: Example of a Surface Chart

9.6.2.1 Area Between Curves

Use texture or shading to indicate the area between curves.

9.6.2.2 Curve Stacking

Stack the series with the least variable series at the bottom and the most variable at the top.

9.6.2.3 Labeling Surface Charts

Place labels within the textured or shaded bands if space is available.

9.6.3 Bar Graphs

Bar graphs represent the magnitudes of numeric data by the lengths of parallel bars. Bars may be vertically or horizontally oriented and are usually spaced apart along an axis containing discrete reference points (e.g., months, mid-points of sample intervals, non-numeric categories, etc.). Histograms, or stepcharts, are bar graphs without spacing between bars and are used when a large number of intervals must be plotted. Figure 9.8 illustrates a bar graph. Graphic presentations should be designed to conform to user expectations.

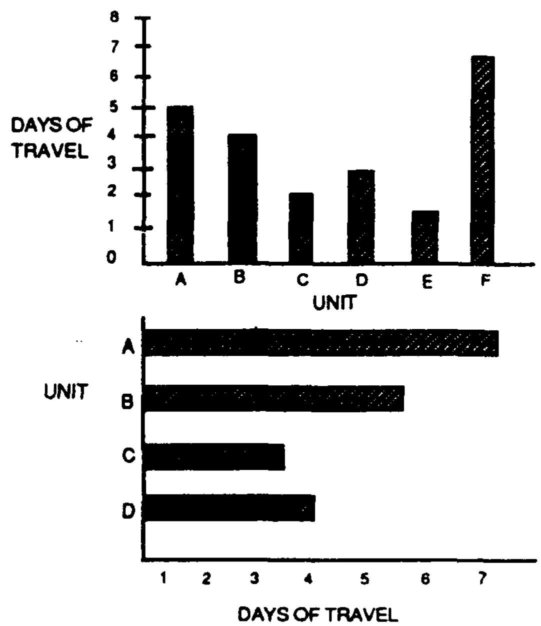
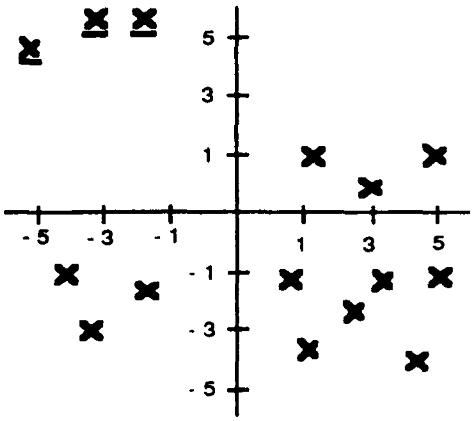


FIGURE 9.8: Example of Bar Graphs

9.6.4 Scatterplots

Scatterplots present data as a two-dimensional distribution of points and should be considered when it is necessary to show how variables are related or to represent the spatial distribution of data (for example, impacts on a target). Highlight data points that are particularly significant. Figure 9.9 illustrates a scatterplot.



Target Distribution of X missile

FIGURE 9.9: Example of a Scatterplot

9.6.5 Pie Charts

Bar graphs are preferable when accurate judgments of magnitudes are required.

Pie charts, like bar graphs, are used to show the proportional distribution of categories with respect to the sum of the categories. See the example of a pie chart in Figure 9.10.

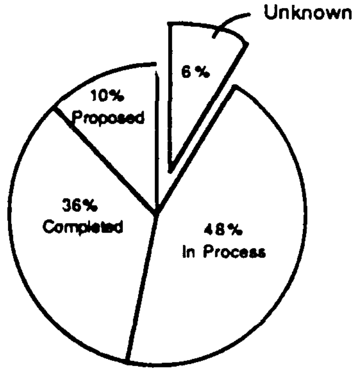


FIGURE 9.10: Example of a Pie Chart

9.6.5.1 Labeling Pie Charts

Place labels in a normal orientation on the segments of pie charts. Segment labels should include numbers that indicate percentages and/or absolute numbers represented by each segment of the display.

9.6.5.2 Highlighting Pie Chart Elements

Segments requiring emphasis should be highlighted or displaced slightly from the rest of the pie chart, as illustrated in Figure 9.10.

9.7 PICTURES

9.7.1 Using Pictures

Consider using pictures when a very detailed representation of objects is required. For example, the scanned map in Figure 9.11.

9.7.2 Automated Aids for Pictures

Provide automated aids when users must perform detailed analyses of image data.

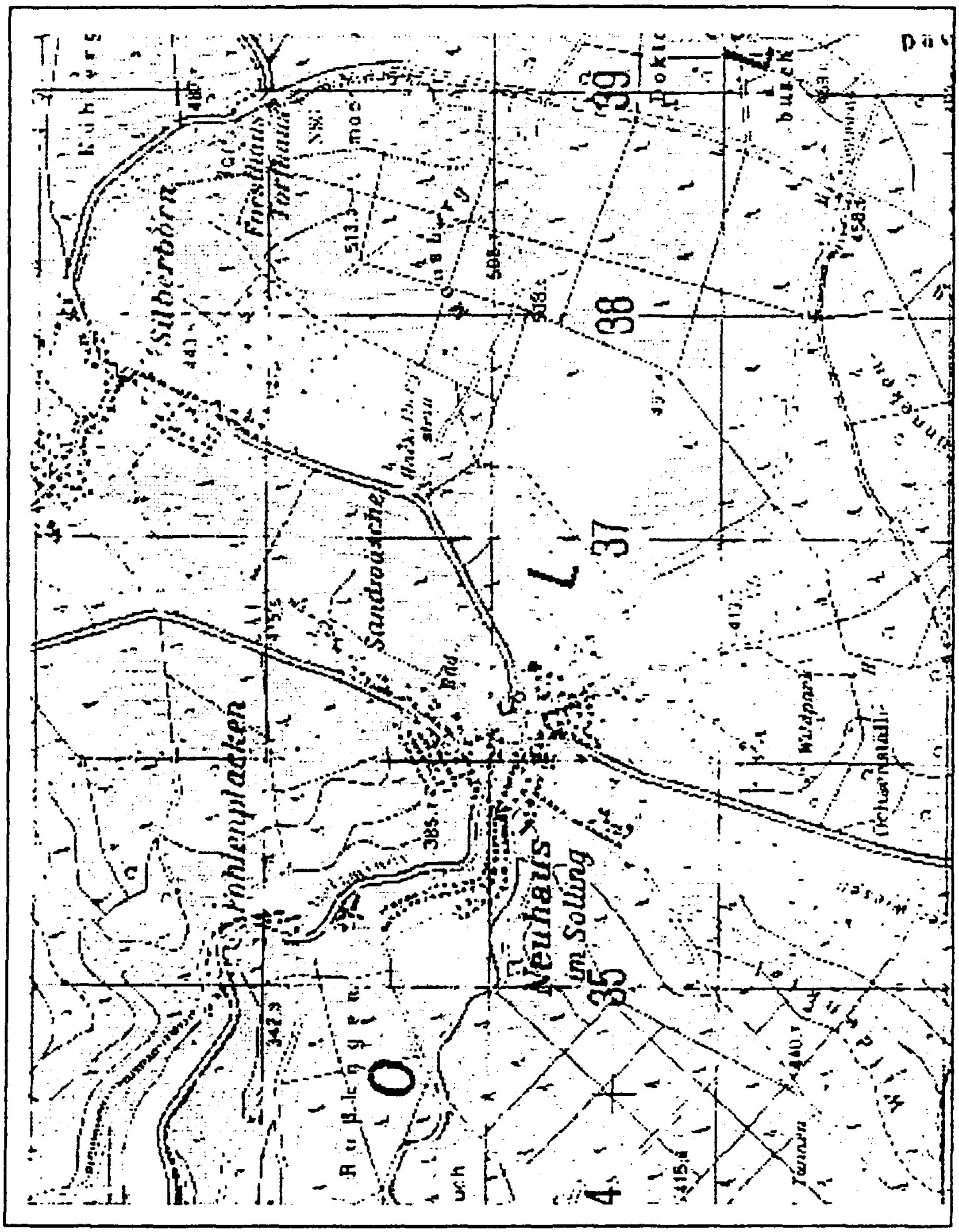


FIGURE 9.11: Example of a Graphic Picture (Scanned Map)

9.8 DIAGRAMS (SCHEMATICS)

9.8.1 Diagrams General

Use diagrams when the user requires information concerning the spatial relationship among objects but does not require the level of detail provided by pictures.

9.8.1.1 Consistent Notation in Diagrams

When diagrammed data are presented in separate sections, use consistent notations across sections, provide an easy means for users to move among sections, and provide an overview of the entire diagram represented by the individual sections.

9.8.1.2 Highlighting Diagrams

Highlight portions of diagrams requiring special user attention.

9.8.1.3 Rotation Capability for Diagrams

Provide a capability for the user to rotate displayed diagrams where it is necessary to view the object from different perspectives.

9.8.2 Flowcharts

Use flowcharts to provide a schematic representation of sequential processes. They may also be used as aids to problem solving when solutions can be reached by answering a series of questions. Figure 9.12 illustrates a typical flowchart.

9.8.2.1 Ordering Flowchart Elements

As appropriate, sequence flowchart elements in a logical order; otherwise, when designing flowcharts, minimize path lengths to reduce size.

9.8.2.2 Orientation of Flowchart Paths

The layout of flowchart paths should conform to standard orientation conventions (i.e., left to right, top to bottom, or clockwise).

9.8.2.3 Consistent Coding of Flowcharts

Consistently apply the coding schemes for flowchart elements.

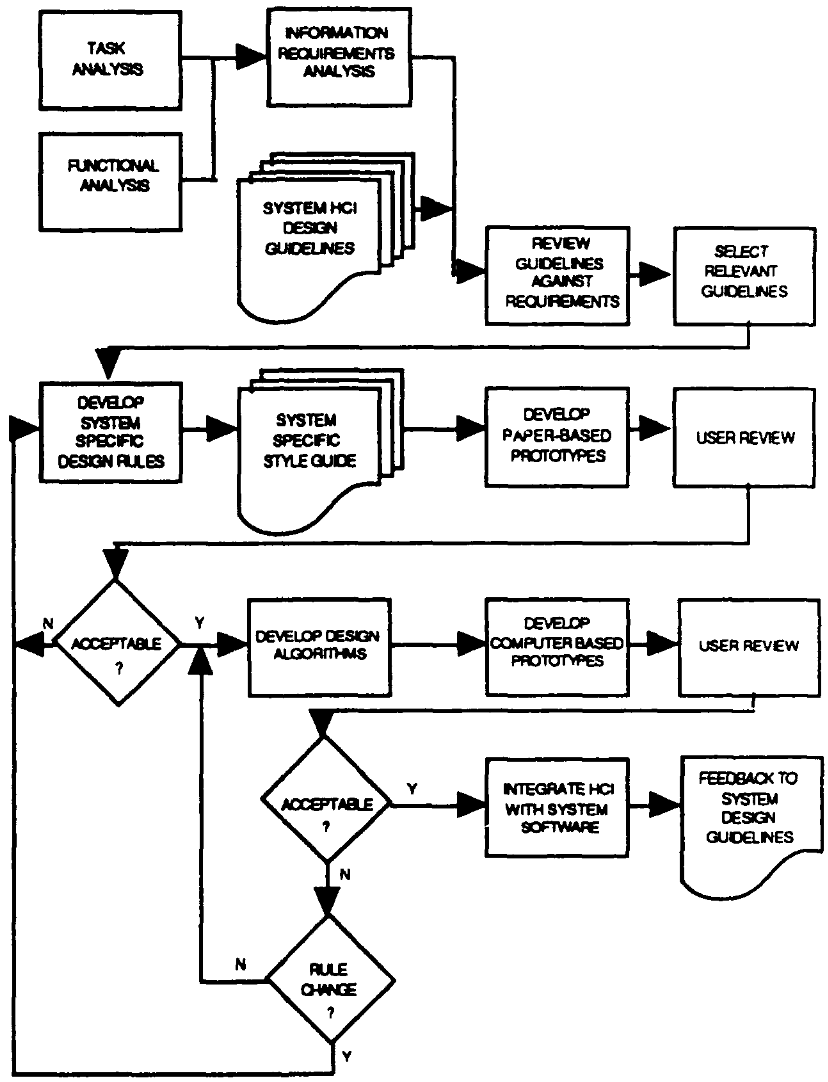


FIGURE 9.12: Sample Flowchart

9.8.2.4 Directional Conventions for Flowcharts

Use standard directional conventions when using arrows to connect elements of flowcharts.

9.8.2.5 Highlighting Flowcharts

Use highlighting to direct a user's attention to elements of particular significance.

9.8.2.6 Flowchart Design for Decision Aids

When using flowcharts as decision aids, require only one decision at each step, and provide the user with a logically ordered list of available options at each step.

9.8.2.7 Consistent Wording for Flowcharts

Use consistent wording for options displayed at decision points.

This Page Intentionally Left Blank

REFERENCE LIST

Paragraph	Reference
9.0	Smith and Mosier (1986) para 2.4.6-2
9.0	Smith and Mosier (1986) para 5.2
9.1	Lewis and Fallesen (1989) para 2.1.1-3
9.1	Brown (1989) para 5.3-8, 5.10
9.1	Brown (1989) p. 88
9.1.1	Lewis and Fallesen (1989) para 2.3.6
9.1.2	Bowser (1991) p. 15; Lewis and Fallesen (1989) para 2.3.9
9.1.4	Smith and Mosier (1986) para 2.4-5
9.1.6	Lewis and Fallesen (1989) para 2.3.2.2b
9.1.7	Lewis and Fallesen (1989) para 2.3.2.2c
9.1.8	Smith and Mosier (1986) para 2.4-4
9.2.1	Smith and Mosier (1986) paras 1.6.1-1 and 1.6.1-18
9.2.2	Smith and Mosier (1986) para 1.6-19
9.2.3	Smith and Mosier (1986) para 1.6.1-2
9.2.4	Smith and Mosier (1986) paras 1.6.1-2 and 1.6.1-4
9.2.5	Smith and Mosier (1986) paras 1.6.1-5 and 1.6.1-6
9.2.6.1	Smith and Mosier (1986) para 1.6.2-1
9.2.6.2	Smith and Mosier (1986) para 1.6.2-2
9.2.6.3	Smith and Mosier (1986) para 1.6.2-3
9.2.6.5	Smith and Mosier (1986) paras 1.6.2-4 and 1.6.2-5
9.2.7	Smith and Mosier (1986) paras 1.6.2-6, 1.6.2-7 and 1.6.2-18

REFERENCES LIST (Cont.)

Paragraph	Reference
9.2.8	Smith and Mosier (1986) paras 1.6.2-8 and 1.6.2-9
9.2.9	Smith and Mosier (1986) para 1.6.2-11
9.2.10	Smith and Mosier (1986) paras 1.6.2-10, 1.6.2-12 and 1.6.2-13
9.2.11	Smith and Mosier (1986) para 1.6.2-14
9.2.12	Smith and Mosier (1986) para 1.6.2-15
9.2.13	Smith and Mosier (1986) para 1.6.2-17
9.2.14	Smith and Mosier (1986) para 1.6.2-19
9.3.1	Smith and Mosier (1986) para 2.4.1-1
9.3.2	Smith and Mosier (1986) para 2.4-12
9.3.3	Brown (1989) para 4.2.7
9.3.4	Smith and Mosier (1986) para 2.4-14
9.3.4	Lewis and Fallesen (1989) para 2.4.7.1
9.3.5	Smith and Mosier (1986) para 2.4.1-7
9.3.6	Smith and Mosier (1986) para 2.4.1-8
9.3.9	Lewis and Fallesen (1989) para 2.3.1.3
9.3.10	Lewis and Fallesen (1989) para 2.3.1.4
9.3.11	Lewis and Fallesen (1989) para 2.3.1.5
9.3.12	DoD (1989a) para 5.15.3.6.4
9.3.13	Lewis and Fallesen (1989) para 2.3.1.8-9
9.3.14	Smith and Mosier (1986) para 2.4.1-7
9.3.15	Lewis and Fallesen (1989) para 2.3.3
9.3.16	Lewis and Fallesen (1989) para 2..4-5
9.3.17	Lewis and Fallesen (1989) para 2.3.5
9.4.1	Smith and Mosier (1986) para 2.46

REFERENCE LIST (Cont.)

Paragraph	Reference
9.4.2	Smith (1986) para 2.4-7
9.4.3	Smith and Mosier (1986) para 2.4-8
9.4.4	Smith and Mosier (1986) para 2.4-9
9.4.5	Smith and Mosier (1986) para 2.4-10
9.5	Lewis and Fallesen (1989) para 2.3.2
9.5.1	Lewis and Fallesen (1989) para 2.3.2.1
9.5.2	Lewis and Fallesen (1989) para 2.3.2.1.1
9.5.3	Lewis and Fallesen (1989) para 2.3.2.2a
9.6.1.1	Smith and Mosier (1986) para 2.4.3-1
9.6.1.2	Smith and Mosier (1986) para 2.4.3-3
9.6.1.2	Smith and Mosier (1986) para 2.4.3-4
9.6.1.3	Smith and Mosier (1986) para 2.4.3-5
9.6.1.4	Smith and Mosier (1986) paras 2.4.3-6 and 2.4.3-7
9.6.1.5	Smith and Mosier (1986) para 2.4.3-8
9.6.1.6	Smith and Mosier (1986) para 2.4.3-10
9.6.1.7	Smith and Mosier (1986) para 2.4.3-11
9.6.2.1	Smith and Mosier (1986) para 2.4.3-12
9.6.2.2	Smith and Mosier (1986) para 2.4.3-13
9.6.2.3	Smith and Mosier (1986) para 2.4.3-14
9.6.3	Bowser (1991) p. 15; Smith and Mosier (1986) para 2.4.4-3

REFERENCE LIST (Cont.)

Paragraph	Reference
9.6.4	Smith and Mosier (1986) para 2.4.2-3
9.6.5	Smith and Mosier (1986) para 2.4.5-1
9.6.5.1	Smith and Mosier (1986) paras 2.4.5-2 and 2.4.5-3
9.6.5.2	Smith and Mosier (1986) para 2.4.5-5
9.7	Smith and Mosier (1986) para 2.4.6-1
9.7.1	Smith and Mosier (1986) para 2.4.6-6
9.8.1.1	Smith and Mosier (1986) para 2.4.6-3
9.8.1.2	Smith and Mosier (1986) para 2.4.6-3
9.8.1.3	Smith and Mosier (1986) para 2.4.6-5
9.8.2	Smith and Mosier (1986) paras 2.4.7-1 and 2.4.7-2
9.8.2.1	Smith and Mosier (1986) paras 2.4.7-3 and 2.4.7-4
9.8.2.2	Smith and Mosier (1986) para 2.4.7.5
9.8.2.3	Smith and Mosier (1986) para 2.4.7-6
9.8.2.4	Bowser (1991) p. 16; Smith and Mosier (1986) para 2.4.7-7
9.8.2.5	Smith and Mosier (1986) para 2.4.7-8
9.8.2.6	Smith and Mosier (1986) paras 2.4.7-9 and 2.4.7-10
9.8.2.7	Smith and Mosier (1986) para 2.4.7-12

10.0 SCREEN DESIGN

Screen design refers to how information is arranged and presented on a display screen. It is difficult to develop standard guidelines for screen design for command and control systems, primarily because of the differences in tasks being performed by users. Screen design requirements can vary extensively, depending on the function being performed by the system. Some systems are actually information management systems that rely heavily on databases and do not require immediate user response to information displayed on their screens. On the other hand, real-time tactical display and control systems require the user to make immediate decisions and to input commands from the information on the display screen. Each system has different screen design requirements based on its primary function. The designer needs to understand the primary function of the system being developed to provide an effective screen design.

Certain common, general principles of human factors engineering (HFE) design should be incorporated into the screen design, regardless of the system function. They are:

- Organization of information should be guided by Gestalt principles of perception, such as rules of proximity and similarity. These are discussed in greater detail in the introduction to Section 6.

- Display formats should be designed to provide optimum transfer of information to the user by the use of information coding, density, grouping, and enumerating.

- Information should be presented simply and in a well-organized manner for high information transfer.

- The user's performance is improved by the following screen features:

    - an orderly, clutter-free appearance

    - information present in expected locations

    - plain, simple language

    - a simple way to move through the system

    - a clear indication of interrelationships.

- Displays should be formatted to group data items on the basis of some logical principle, considering trade-offs derived from task analysis.

- Screen design should minimize pointer and eye movement requirements within the overall design. The goal to minimize eye and pointer movement must be considered within general task considerations, with logical trade-offs taken into account.

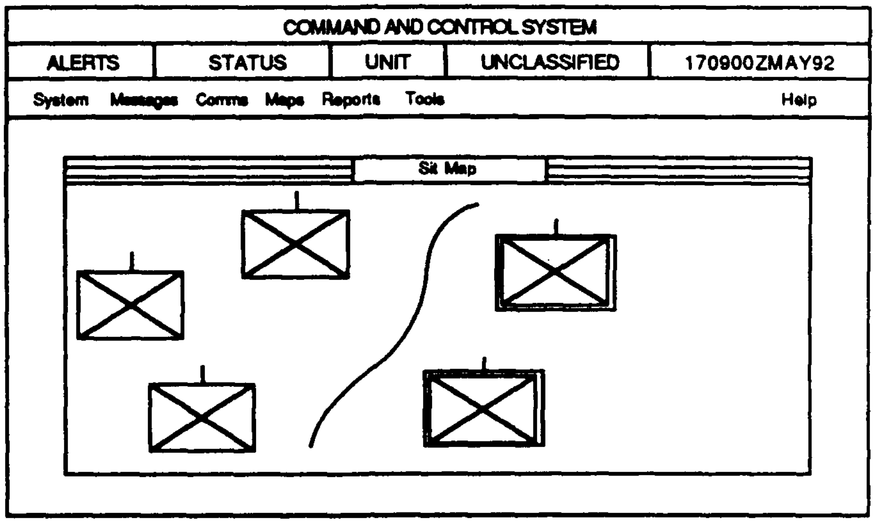
The following guidelines represent design considerations that should be applied, for the most part, to the screen design of all command and control systems.

This Page Intentionally Left Blank

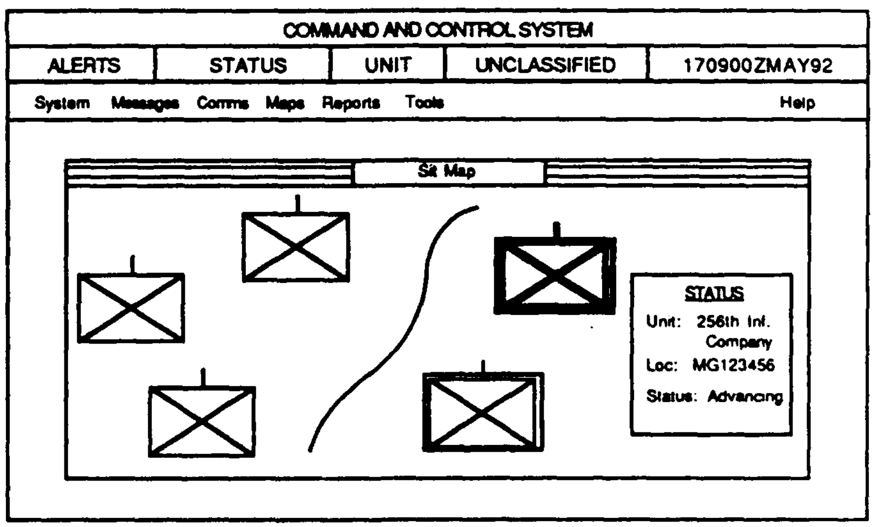
10.1 GENERAL

10.1.1 Information Display Based on Criticality

A criterion should be established for prioritizing different levels of displayed information. For example, critical tactical information should always be displayed, whereas optional information may be made available by request. See Figure 10.1.



Display Screen Design Where Critical Information is Always Present



How a Display Screen Design Can Provide Optional Information Through User Request

FIGURE 10.1: Tactical Information Display Options

10.1.2 Display Only Critical Information

The information density on a system display should be minimized by presenting only information essential to the user at the time.

10.1.3 Integrated Display

When the user needs specific data displayed concurrently to make judgments on the tactical situation, provide those data in an integrated display, rather than partitioning them into separate windows.

10.1.4 Information Format

Present information in a directly usable form. Do not require the user to decode or interpret data.

10.2 CONSISTENCY

The structure of information presented on the screen should be consistent. This helps the user develop a perceptual model of the interface.

10.2.1 Consistent Display Structure

Create display formats with a consistent structure evident to the user, so display features are always presented in the same way.

10.2.2 Consistent Fields

Use fields, such as headers, that are constant and stay the same on every page.

10.2.3 Input Prompts

Display the input prompt at a standard location, next to the command entry area of the display.

10.3 FORMAT

10.3.1 General

10.3.1.1 Distinctive Display Elements

Make the different elements in a display format distinctive.

10.3.1.2 Visual Competition

Information on a display screen should be organized such that visual competition among distinct items of information is minimized.

10.3.1.3 Abbreviations

Make appropriate use of abbreviations, and use them in a consistent manner. Provide a key or built-in reference table. Abbreviations should conform to AR310-50 (U.S. Department of the Army 1985). Do not place periods after abbreviations. It is recommended that applications requiring extensive text input provide an on-line spell-checker that addresses abbreviations and acronyms.

10.3.1.4 Use of Contrasting Features

Use contrasting features such as inverse video and color to call attention to different screen components and urgent items.

10.3.1.5 Simple Statements

Use short, simple statements in text.

10.3.2 Screen Organization

10.3.2.1 Order of Data

The order of items should follow some principle that can be recognized and applied by the user.

10.3.2.2 Display Title at Top

Begin every display with a title or heaoer located at the top of the page, describing briefly the contents or purpose of the display. Leave at least one blank line between the title and the body of the display. See Figure 10.2 for an example.

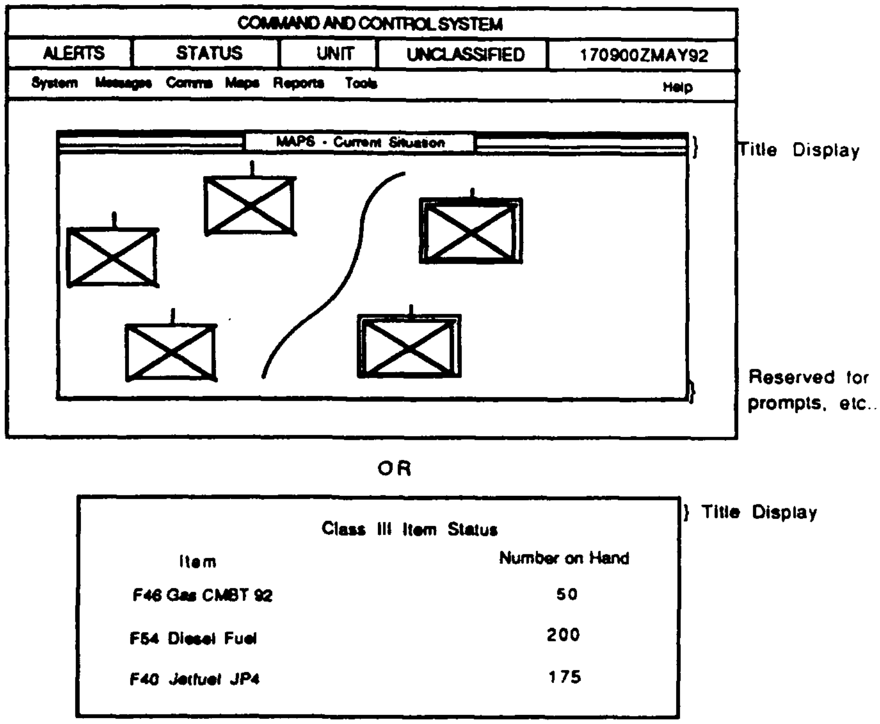


FIGURE 10.2: Example of How Specific Types of Information Should Be Located on a Display Screen

10.3.2.3 Command Entry, Prompts, Messages at Bottom

The area set aside for displaying messages should be consistent. Text systems have reserved the last several lines at the bottom of displays for status and error messages, prompts, and command entry, when appropriate. See Figure 10.2 for an example. This area can also be used for a supporting data menu bar, including such items as a user note pad.

There are insufficient data regarding screen density for graphical user interfaces and map graphic type displays. Research is needed in these areas.

10.3.2.4 Screen Density

For text displays, screen density (i.e., ratio of characters to blank spaces) should not exceed 60% of available character spaces.

10.3.2.5 Instructions

Instructions on how to use a screen should be highlighted at the top of the text, preceding the response options, as illustrated in Figure 10.3. Instructions on the disposition of a completed screen should be at the bottom.

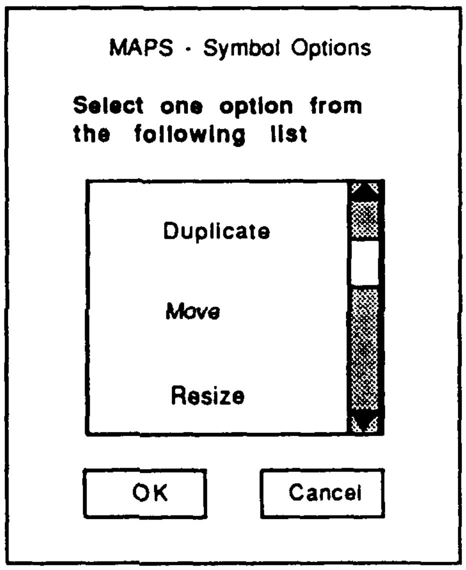


FIGURE 10.3: Example of the Proper Location of Display Screen Instructions

10.3.2.6 Use of Functional Fields

Assign functional fields for particular kinds of data, such as program messages, error messages, system messages, and alarms.

10.3.3 Data Organization

10.3.3.1 Display Capacity

Large portions of text should be broken into smaller, meaningful groups to minimize the amount of information to be attended to at one time. See Figure 10.4.

Poor	Good
The 3rd Bn is currently located at 32UNA100100, moving to contact in sector 8, with 80% strength, supported by an armor platoon.	3 Bn Status o At 32UNA100100 o Moving to Contact o 80% Strength o Supported by an armor platoon

FIGURE 10.4: Example of Breaking Information into Smaller Groups

10.3.3.2 Spacing to Structure Displays

Use blank space to structure a display.

10.3.3.3 Use of Two Text Columns

For screens containing large amounts of text, consider using two columns of text to improve readability.

10.3.3.4 Placement of Labels

Ensure labels are sufficiently close to their related data fields but separated by at least one space.

10.3.3.5 Spacing Between Words and Lines

Provide adequate spacing between words and lines of text for better legibility. Separate paragraphs with a blank line.

10.3.3.6 Vertical Presentation of Data

A series of data elements should be presented vertically, not horizontally, in text. See Figure 10.5.

POOR

Class I	Item 1	Item 2	Item 3
Class II	Item 1	Item 2	Item 3
Class III	Item 1	Item 2	Item 3

GOOD

Class I	Class II	Class III
Item 1	Item 1	Item 1
Item 2	Item 2	Item 2
Item 3	Item 3	Item 3

FIGURE 10.5: Example of How Data Should Be Presented Vertically as Opposed to Horizontally

10.3.3.7 Starting Point

Provide an obvious starting point for information.

10.3.3.8 Justification of Columns

Columns should be justified, as noted below and as illustrated in Figure 10.6.

a. Left justify alphanumeric columns to permit rapid scanning.

b. Right justify numerical data without decimals.

c. Justify numerical data with decimal points by the decimal.

Poor	Good
Artillery	Artillery
Tanks	Tanks
Trucks	Trucks
Aircraft	Aircraft
400	400
4210	4210
38	38
39111	39111
1.5	1.5
10.38	10.38
1.365	1.365

FIGURE 10.6: Example of How Data Should Be Justified

10.3.4 Line Organization

10.3.4.1 Format for Detailed Reading

Use only 70-character positions on the standard 80-character line to increase reading efficiency. This is most important when detailed reading is the user's primary task.

10.3.4.2 Line Length

Display no more than 35 to 40 characters on each line for information presented in columns.

10.3.5 Character Design

10.3.5.1 Capital Lettering

a. Capital letters should be used for typographic coding, headlines, indicating new paragraphs, and/or captions and labels.

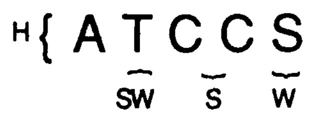
b. Do not use all capital letters in running text or tables. It impairs word recognition, reduces readability, and limits space between text lines.

10.3.5.2 Spacing Between Characters and Lines

Spacing between characters should be 20-50% of character height. Spacing between lines should be equal to character height. See Figure 10.7.

10.3.5.3 Character Height and Width

Minimum height of displayed characters should be 1/200 of viewing distance. For example, a viewing distance of 36 inches requires a 0.18-inch character height on the display screen. Character width should be 50-100% of character height. Character stroke width minimum is 10-12.5% of character height (see Figure 10.7). Maximum text size should not exceed 10% of the available vertical display area on a full-size screen.



Height	=	H	=	X
Spacing	=	S	=	20-50% of X
Width	=	W	=	50-100% of X
Stroke Width	=	SW	=	10-12.5% of X

FIGURE 10.7: Example of Character Size and Spacing

10.3.5.4 Dot Matrix Construction

Characters should contain a minimum 7 x 9 dot matrix construction for better readability.

10.3.5.5 Readability

Command and control applications require the characters on the screen be legible to a person standing behind the user. The screen viewing distance referred to in Paragraph 10.3.5.3 should reflect the anticipated maximum viewing distance. Using large fonts with broad stroke widths is recommended to improve readability. Selecting background color and contrasting foreground (text) color should ensure sufficient contrast for good readability.

10.3.5.6 Font Size

The usual designation of font size is given in points. The display of text fonts on screens is proportional to point size, but the actual size of displayed text is related to screen size and application software. The font point size only controls the actual size of printed output. It is recommended that screen text size be reviewed and adjusted in relation to the objective hardware system.

10.3.6 Scrolling of Data

10.3.6.1 Avoid Constant Scrolling

Display text information statically on the screen, rather than constantly scrolling it across the screen.

10.3.6.2 Format for Scrolling Text

If text is meant to be scanned by constant scrolling, columns with 35 characters or fewer per line are preferred.

10.4 GROUPING

10.4.1 Grouping for Data Comparison

If users must analyze sets of data to discern similarities, differences, trends, and relationships, structure the format of displays so the data are consistently grouped.

10.4.2 Important Data Placement

Where displayed data are used in some spatial or temporal order, consider grouping those data by sequence of use to preserve that order.

10.4.3 Primary Viewing Area

When data and terms are particularly important, require immediate user response, or are more frequently displayed, group them in the primary viewing area of the user.

10.4.4 Data Grouped Alphabetically or Chronologically

When there is no appropriate logic for grouping data by sequence, function, frequency or importance, adopt some other principle, such as alphabetical or chronological grouping.

10.4.5 Arrangement of Data on Screen

Arrange and group data on command and control system display screens to differentiate among instructions and data and to facilitate observation of similarities, differences, and trends for the most common uses.

10.4.6 Provide Cohesive Groupings

Provide cohesive groupings of command and control screen elements by using blank space, surrounding lines, different intensity levels, etc.

10.5 MULTIPLE SCREENS

10.5.1 Paging Crowded Displays

When a display contains too much data for presentation in a single frame, partition the data into separately displayable pages. Refer to Section 7, Windowing.

10.5.2 Related Data on Same Page

When partitioning displays into multiple pages, take into account the type of data being partitioned, and display functionally related data items together on one page.

10.5.3 Multiple Pages Labeling

In a multipage display, label each page with a unique identifier that shows its relation to the other pages.

This Page Intentionally Left Blank

REFERENCE LIST

Paragraph	Reference
10.1.1	Slominski and Young (1988) p. 2
10.1.2	Licktieg (1989) p.9; Brown et al. (1983) p. 1-10; Galitz (1984) p. 99 & 102; Tullis (1988) p. 382
10.1.3	Smith and Mosier (1986) para 2.5-7
10.1.4	Galitz (1984) p. 103; Shneiderman (1987) p. 327
10.2.1	Williams (1987b) Appendix A p. A-1; Smith and Mosier (1986) para 4.0-6; Brown et al. (1983) p. 1-11; Shneiderman (1987) p. 327; Smith and Mosier (1986) para 2.5-1; Licktieg (1989) p.10; Brown et al. (1983) p. 1-1 & 1-11, Tullis (1988) pp. 393 & 336; Hamel and Clark (1986) p. 26; Slominski and Young (1988) p. 2
10.2.2	Brown et al. (1983) p. 1-1
10.2.3	Williams (1987b) Appendix A p. A-2; MacGregor and Lee (1988) p. 10; Galitz (1984) p. 103; Shneiderman (1987) p. 336
10.3.1.1	Smith and Mosier (1986) para 2.5-2

REFERENCE LIST (Cont.)

Paragraph	Reference
10.3.1.2	Hamel and Clark (1986) p. 28; Slominski and Young (1988) p. 3-4
10.3.1.3	Bowser (1991) p. 16; Tullis (1988) p. 385
10.3.1.4	Galitz (1984) p. 103
10.3.1.5	Shneiderman (1987) p. 327
10.3.2.1	Nes (1986) p. 105
10.3.2.2	Smith and Mosier (1986) para 2.5-10; Licktieg (1989) p. 10; Brown et al. (1983) p.1-5 & 1-12; Shneiderman (1987) p. 327
10.3.2.3	Bowser (1991) p. 16; Smith and Mosier (1986) para 2.5-11; Galitz (1984) p. 102
10.3.2.4	Tullis (1988) p. 382
10.3.2.5	Galitz (1984) p. 102; Licktieg (1989) p. 9
10.3.2.6	Brown et al. (1983) p. 1-4
10.3.3.1	Williams et al. (1987b) Appendix A p. A-1
10.3.3.2	Smith and Mosier (1986) para 2.5-3
10.3.3.3	Nes (1986) p. 103; Shneiderman (1984) p. 104; Brown et al. (1983) p. 1-6
10.3.3.4	Shneiderman (1987) p. 327

REFERENCE LIST (Cont.)

Paragraph	Reference
10.3.3.5	Nes (1986) p. 101; Brown et al. (1983) p. 1-13; Shneiderman (1987) p.105; Tullis (1988) p. 398-399; Grabinger and Amedeo (1988) p. 198
10.3.3.6	Tullis (1988) p. 395
10.3.3.7	Galitz (1984) p. 102
10.3.3.8a	Shneiderman (1987) p. 327
10.3.3.8b	Brown (1989) p. 28
10.3.3.8c	Brown (1989) p. 29
10.3.4.1	DoD (1985) p. 3-3
10.3.4.2	Tullis (1988) p. 399; Galitz (1984) p. 104; DoD (1985) p. 3-3
10.3.5.1a	Shneiderman (1987) p. 104; Nes (1986) p. 112; Tullis (1988) p. 397
10.3.5.1b	Nes (1986) p. 112; Brown et al. (1983) p. 1-9; Shneiderman (1987) p.104; Tullis (1988) p. 397
10.3.5.2	Galitz (1984) p. 184
10.3.5.3	Galitz (1984) p. 184
10.3.5.4	Shneiderman (1987) p. 184
10.3.5.5	Bowser (1991) p. 16
10.3.5.6	Bowser (1991) p. 16
10.3.6.1	DoD (1985) p. 3-2
10.3.6.2	DoD (1985) p. 3-3

REFERENCE LIST (Cont.)

Paragraph	Reference
10.4.1	Smith and Mosier (1986) paras 2.5-13 and 2.5-15; Tullis (1988) p. 387; Shneiderman (1987) p. 336
10.4.2	Smith and Mosier (1986) para 2.5-14
10.4.3	Smith and Mosier (1986) paras 2.5-16 and 2.5-17
10.4.4	Smith and Mosier (1986) para 2.5-18
10.4.5	Brown et al. (1983) p. 1-11
10.4.6	Galitz (1984) p. 102
10.5.1	Smith and Mosier (1986) para 2.5-4
10.5.2	Smith and Mosier (1986) para 2.5-5; Galitz (1984) p. 103; Shneiderman (1987) p. 327
10.5.3	Smith and Mosier (1986) para 2.5-6; Shneiderman (1987) p. 327; Brown et al. (1983) p. 1-5

11.0 COLOR USAGE

As emerging command and control systems implement graphical user interfaces and high resolution color graphics displays, the use of color as an information discriminator has become crucial. Color can be a very effective discriminator, for example, by decluttering a display and improving task performance. Color can also induce the very clutter and performance degradation it attempts to reduce. For these reasons, color in a display must be used very carefully.

A designer must be sensitive to the many factors that affect how a person perceives and reacts to color as an information discriminator. An in-depth discussion on visual perception, color, and human performance is beyond the scope of this document; nevertheless, Table 11.1 provides definitions of key terms associated with the subject.

The designer should be aware of the following important factors:

- Both brightness and type of lighting (e.g., incandescent versus fluorescent) can affect how colors are perceived. For example, bright ambient light desaturates display colors, leading to degraded color identification and discrimination. It may shift the eye's adaption, also reducing the ability to discriminate color (Thorell and Smith 1990). In essence, identically colored objects can be perceived as being dissimilar under different lighting conditions.

- How the color of a foreground object is perceived is directly related to its background color.

- Visibility and readability are a direct result of the contrast between the foreground and the background.

Additional overarching guidelines for using color in computer display systems are:

- Color should be used sparingly as an information discriminator. Color rapidly loses meaning and, when overused, may impede rather than enhance human performance.

- Colors should be used consistently within a display and across a set of displays for an application.

- The meaning of color should be consistent with user expectation.

- When using color to impart a specific meaning to the user, utilize an additional, redundant form of coding, such as shape. This ensures the correct meaning will be conveyed should the user have a color vision deficit or should color be unavailable on the screen.

- Whereas flexibility in color coding schemes may be desirable for a terminal dedicated to a single user, color coding should be standardized for military applications. Because of the variety of users on a tactical terminal, a terminal with a uniquely customized color coding scheme may be very difficult to interpret. The user should be allowed to select colors schemes for aspects of the application that do not involve coding or status. A default scheme should be easily available to restore the interface for subsequent users.

- A requirement for adjustable colors is created because of portable applications among hardware configurations. Each hardware system display has different color perceptual values and color names. Portable applications must be able to accommodate these differences. Status colors should be assigned during installation; the user should not be allowed to adjust these colors.

The following paragraphs provide more detailed guidelines for using color in command and control systems. It should be noted that many of the guidelines contained in this section are most relevant to using color in text-based software, primarily because the majority of the past research in color usage was done with text applications. Whereas results of research with graphical user interfaces and graphical presentation of information are emerging, more research is needed on using color in tactical graphics applications, especially in foreground/background combinations for colored map graphic displays.

The designer should also be aware that it is easier to define color combinations to avoid than to identify a single best way to utilize color. Color choice/combination tends to be a matter of personal preference. For example, high contrast between foreground text and background is crucial but can be accomplished with a number of color combinations. This ambiguity becomes all the greater when developing design guidance for the different command and control applications represented across military systems. The designer should utilize those guidelines most relevant to the particular application.

TABLE 11.1: Definitions of Key Terms for Color Usage

TERM	DEFINITION
Achromatic	Colorless; lights that have no definite hue.
Brightness	The perceptual (psychological) correlate of intensity that ranges from dark to bright.
Chromatic	Highly colored.
Discrimination	Degree to which a human visual system can sense differences in the physical characteristics of an image.
Hue	The psychological attribute of color sensation associated with the physical property of visible wavelengths.
Legibility	Ability to identify an alphanumeric character or symbol. A criterion of image quality.
Luminance	The amount of light reflected or emitted by a surface, measured in foot-lamberts.
Luminance Contrast	Ratio of the foreground brightness compared to the background brightness.
Monochromatic	Consisting of one color or hue.
Recognition	Ability to recognize or interpret the meaning or association of an image.
Saturation	The degree to which a hue differs from a gray of the same lightness.

This Page Intentionally Left Blank

11.1 GENERAL

11.1.1 When to Use

Color should be used when a basic monochromatic presentation of tactical information needs to be augmented for the user to gain a more effective understanding of the information being presented.

Color should be used, when necessary, to:

a. attach specific meaning to tactical information presented in the form of text or symbology

b. direct the user's attention to the most important or time-critical information on the screen (i.e., information category headings, system and user errors, information requiring immediate attention, key data items, and window titles)

c. enable a user to differentiate rapidly among several types of information, especially when the information is dispersed on the display

d. increase the amount of information portrayed on a graphic display by adding color in addition to shape

e. indicate changes in the status of graphical data.

11.1.2 Constraints on Use

Color should be used carefully. As a coding method, color can rapidly lose its effectiveness.

The following should be considered when including color in display screens.

11.1.2.1 Adding Color to Formatted Displays

Add color coding only after displays have already been designed as effectively as possible in an achromatic format.

11.1.2.2 Using Color

Color only logically related information with similar hues. Consider spacing or highlighting instead of, or in addition to, color.

11.1.2.3 Relative Brightness of Colors

When emphasizing tactical information by means of color, use a color for more important information that is brighter than adjacent color coded information. Figure 11.1 provides an illustration of color in descending order of brightness when driven to the same level of energy. Ensure the choice of colors is consistent with the user's expectations for the information being coded. See Section 11.2.

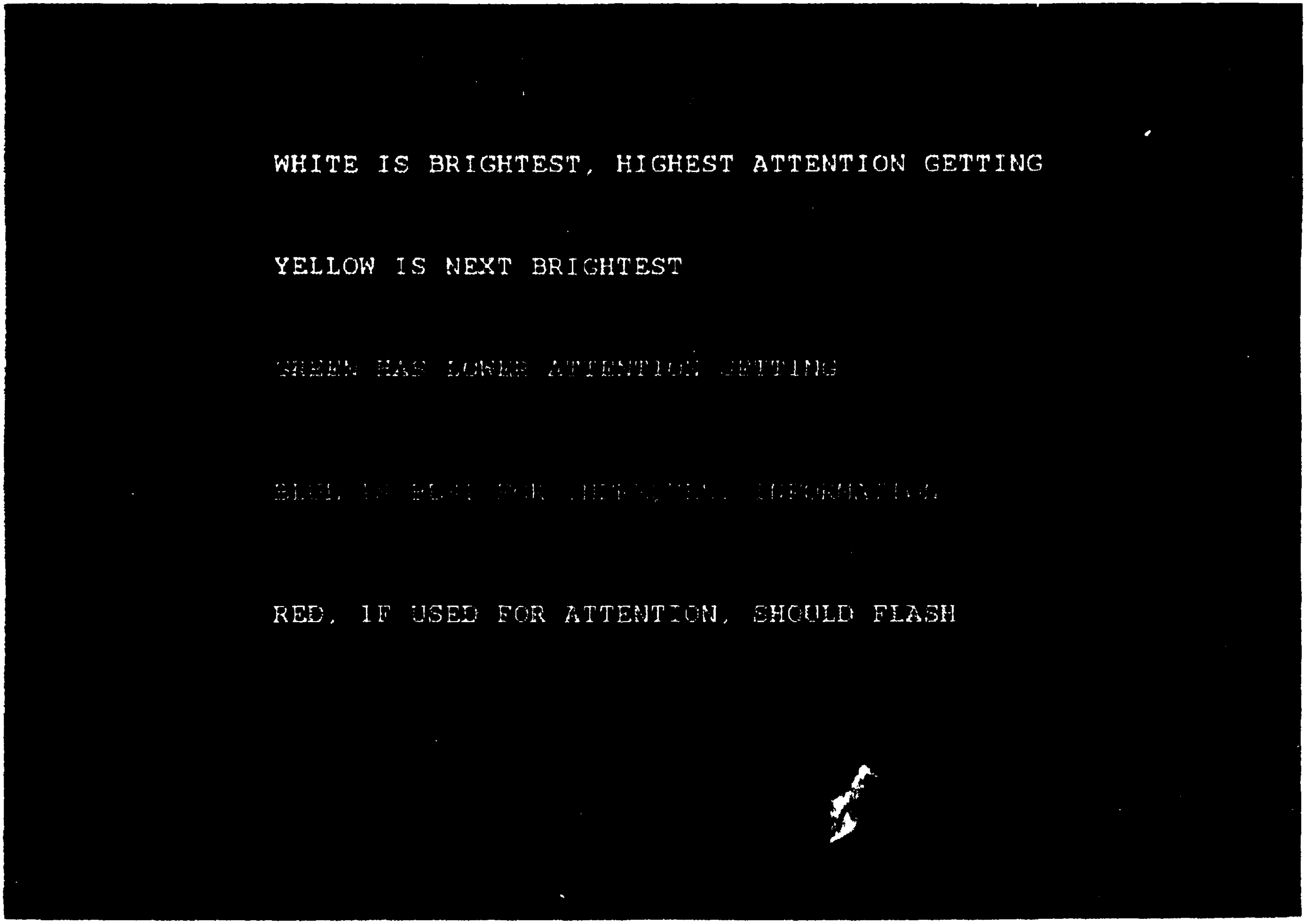


FIGURE 11.1: Relative Brightness of Colors, in Descending Order

This Page Intentionally Left Blank

11.1.2.4 Inappropriate Use of Color Coding

The user with defective color vision will have difficulty discriminating among the colors.

Do not use color coding when it might confuse users with defective color vision or when the use of color reduces screen readability. If color must be used, consider the following:

a. When the user must compare data, such as those contained in graphs based on color, avoid using green, yellow, and red as comparison colors for command and control information requiring important or frequent discriminations. If possible, use yellow and blue, or red and cyan.

b. When green, yellow, and red must be used, provide other cues, such as brightness and saturation, to enhance discriminability.

11.1.2.5 Redundant Coding with Color

Do not code solely by color. Make color coding redundant with some other display feature, such as symbology or actual text content.

11.1.2.6 Color and Small Areas of the Display

Avoid requiring the user to discriminate between colors in small areas of the display. Small, color-coded areas are subject to loss and bleeding of colors. Use achromatic colors (i.e., black or white) if coding must be done in small areas.

11.1.2.7 Highlighting

Use white to highlight data that require particular attention. Do not excessively use white as a highlighter, as it can create a glaring brightness that may interfere with screen legibility. When status changes are signaled by color, that color should not be used to highlight text. Status changes using color coding should be signaled by a ball or box next to the text.

11.1.2.8 Enhancing Color Discrimination

Contrast should be high between the text or graphical object and its background to enhance screen readability. Generally, the color foreground should differ from its background by a minimum of 100 ΔE (CIE Yu'v') distances. Minimum luminance contrast ratios are required for specific tasks. For discrimination and legibility, acceptable ratios of foreground-to-background luminance contrast range from 6:1 to 10:1. Table 11.2 provides guidance for specific conditions. Using pure white or black as a background color is not recommended. Unsaturated hues provide the best background contrast.

TABLE 11.2: Recommended Luminance Contrast Ratios

Condition	Ratio of Foreground to Background
Bright Ambient Illumination	>7:1
To Attract Attention	>7:1
To Sharpen Edges	>7:1
Continuous Reading	3:1 to 5:1
Dark Ambient Illumination	3:1 to 5:1
Camouflage Images or Smooth Edges	<3:1

11.2 COLOR SELECTION

11.2.1 General

11.2.1.1 Easily Discriminable Colors

When selecting colors for coding discrete categories of information displayed on a screen, ensure those colors are easily discriminable in all expected operational environments.

11.2.1.2 Color Discrimination

To aid in color discrimination, use colors that are as widely spaced along the visible color spectrum as possible. The following colors, listed by their wavelengths in millimicrons, are spaced widely enough for easy discrimination from one another.

Red	700
Orange	600
Yellow	570
Yellow-green	535
Green	500
Blue-green	493
Blue	470

11.2.1.3 Color Selection, Infrequently Used Data

Use an unobtrusive color to display information used infrequently on a screen. Unobtrusive colors have shorter wavelengths.

11.2.1.4 Use of Warm and Cool Colors

Use warm colors (colors with a longer wavelength, such as red or orange) to convey action or the requirement for a response. Use cool colors (colors with a shorter wavelength, such as blue or green) to convey status or background information.

11.2.1.5 Unique Assignment of Color Codes

Each color should represent only one category of displayed data, as defined in paragraph 11.2.6.

11.2.2 Consistency

11.2.2.1 Consistent Use of Color

A mismatch of color and color association slows recognition time and increases misidentification of words.

Color should be applied consistently from screen to screen and from application to application to ensure the user can make the proper interpretations. This is applicable both within and across military command and control systems. For example, do not use status colors as window borders unless status coding is intended.

11.2.2.2 Label Wording and Color Associations

Color coding should be consistent with the interaction of the label's color and the color associations of the words in the label. As shown in Figure 11.2, the word ENEMY, if color coded, should be red rather than green.

11.2.2.3 Conventional Assignment of Color Codes

Choose colors for coding based on conventional associations with particular colors. These should conform, if possible, to those specified in Army documents, such as <u>FM 101-5-1: Operational Terms and Symbols</u>.

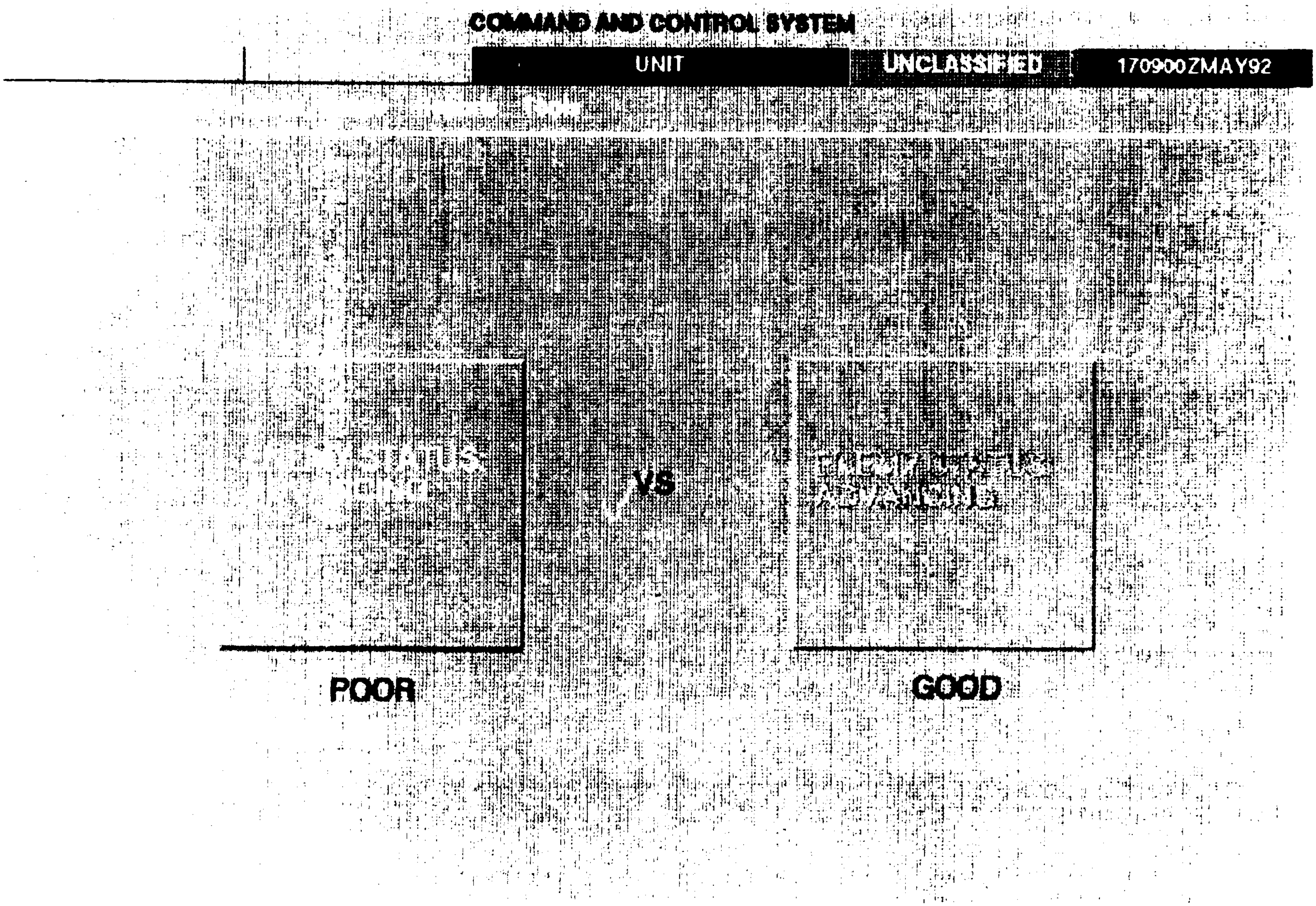


FIGURE 11.2: Example of Color Coding Used Inconsistently and Consistently with the Word's Color Association

This Page Intentionally Left Blank

11.2.3 Number of Colors to Use

11.2.3.1 Conservative Use of Color

Implement color coding conservatively, using relatively few colors to designate critical categories of displayed data and only where it will help user performance. Figure 11.3A illustrates overuse of color. Figure 11.3B illustrates the same display, redesigned to minimize the use of color.

11.2.3.2 Color Coding on Alphanumeric Screens

Use no more than four colors at one time when using alphanumeric screens, with a maximum of seven total for all screens.

Amplifying the coding scheme for graphics can be achieved by using three hues of each color, though this should be done very conservatively.

11.2.3.3 Color Coding on Graphical Screens

Use four standard colors, with others reserved for occasional use. Only eight or nine highly saturated colors can be easily discriminated, so do not exceed this number.

This Page Intentionally Left Blank

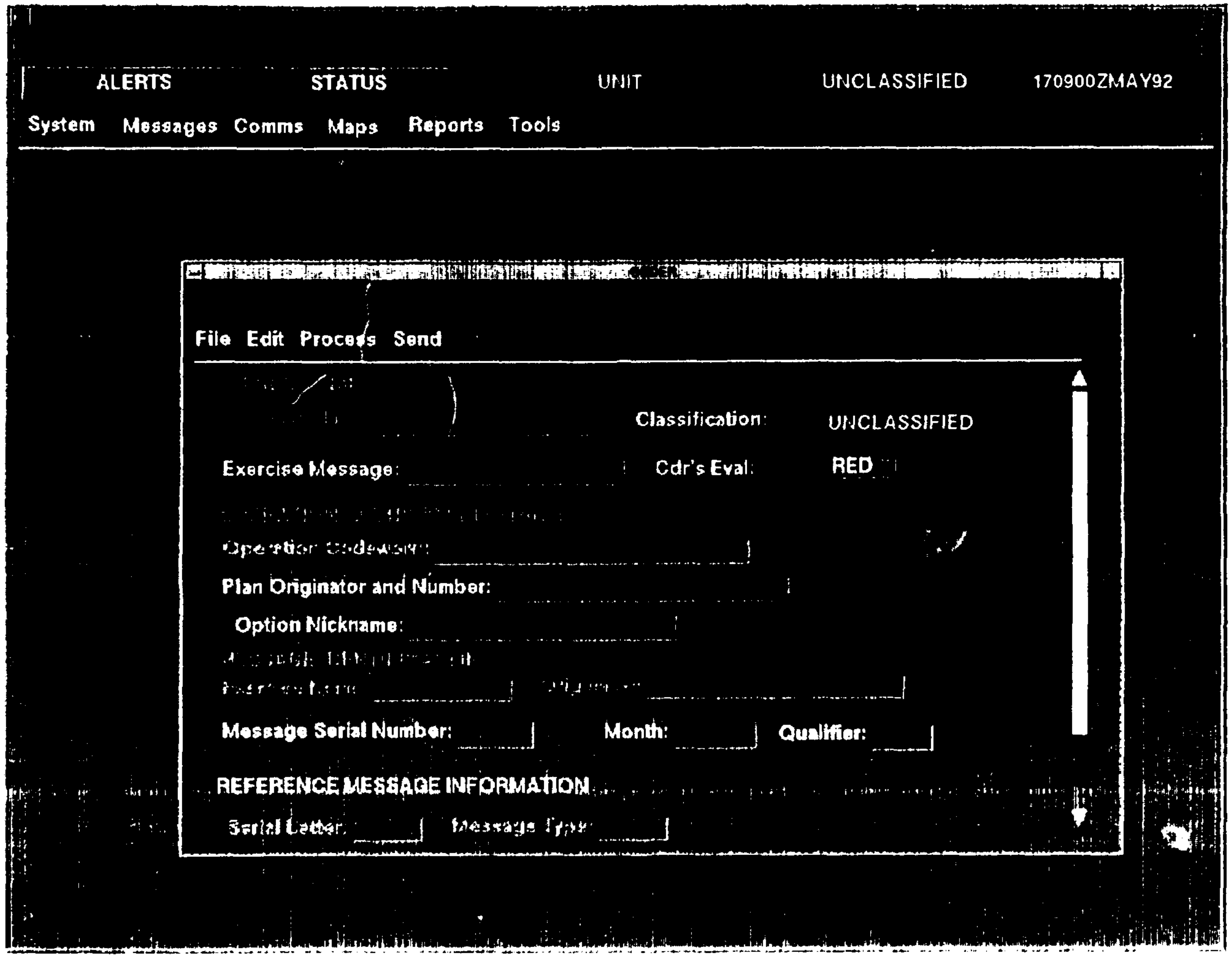


FIGURE 11.3A: Example of Overuse of Color Coding

This Page Intentionally Left Blank

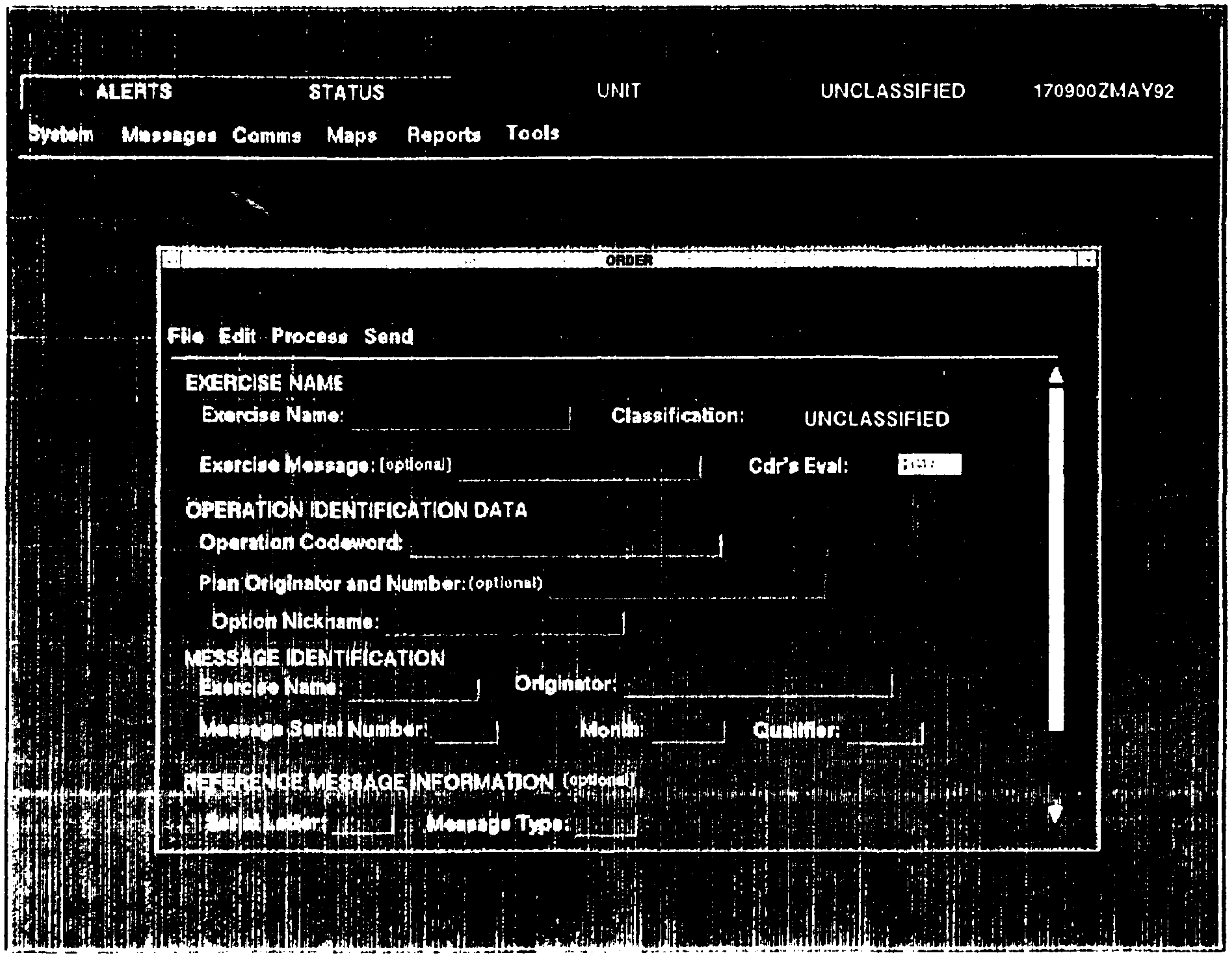


FIGURE 11.3B: Example of Display Design without Color Overuse

This Page Intentionally Left Blank

11.2.4 Pairing of Colors

Colors should be carefully paired on a screen to maximize human performance.

These color pairs have wavelengths that focus at different distances from the retina. When used together they will cause frequent eye refocusing and other performance difficulties.

11.2.4.1 Use of Saturated Colors

Avoid simultaneous use of highly saturated, spectrally extreme color pairs on a display screen. Examples include such color pairs as red and blue, yellow and purple, or magenta and green. The affect is most significant with red and blue. If they must be used, desaturate them. See Figure 11.4.

11.2.4.2 Color Coding of Different Items

To emphasize different tactical information in text and presentation graphics displays, use contrasting colors such as red and green or blue and yellow. However, in color choice, be consistent with the guidance provided in other paragraphs of this section.

11.2.4.3 Color Coding of Similar Items

To convey similarity in tactical information in text and presentation graphics displays, use similar colors, such as orange and yellow or blue and violet.

This Page Intentionally Left Blank

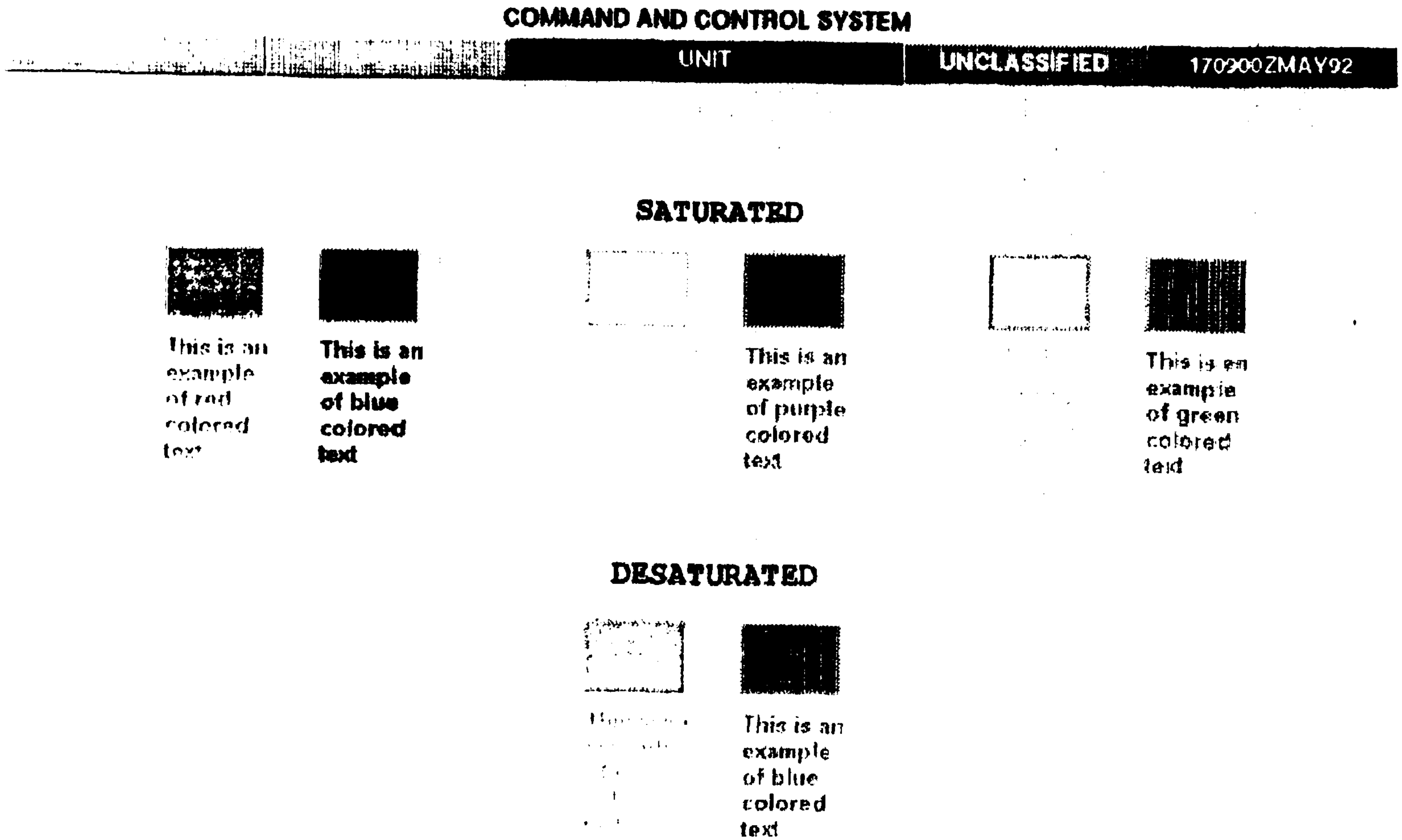


FIGURE 11.4: Example of How Pairing of Certain Highly Saturated Colors Causes One to Appear Closer to the Viewer and How Desaturation Reduces the Effect

This Page Intentionally Left Blank

11.2.4.4 Foreground and Background Combinations

Contrast between foreground objects and background displays should be high. Black on light blue and blue on white are good general foreground and background combinations. However, blue should not be used for the foreground when fine detail resolution is required, such as in target recognition. More specific guidance is illustrated in Figures 11.5A and 11.5B.

This Page Intentionally Left Blank

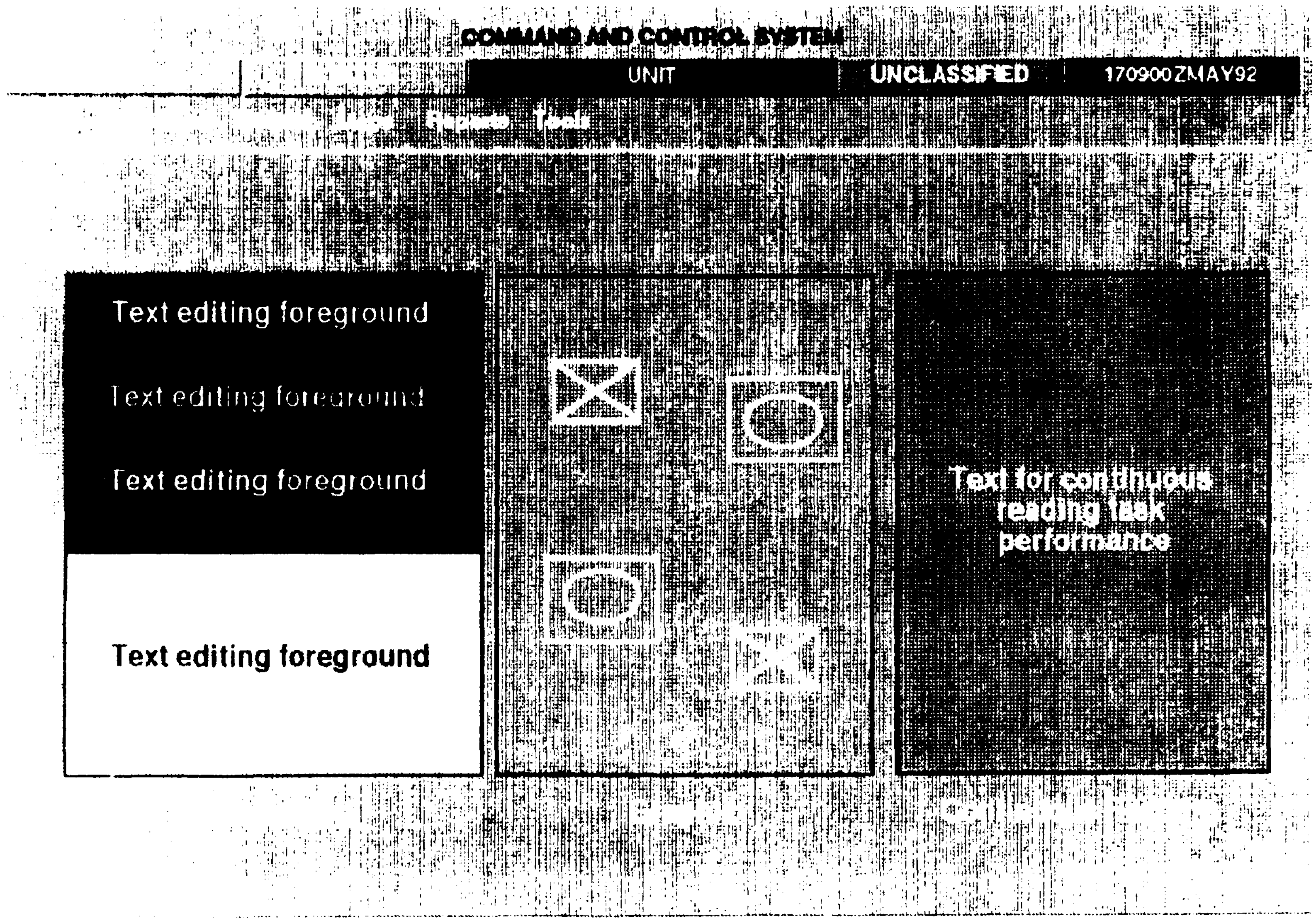


FIGURE 11.5A: Recommended Foreground and Background Color Combinations

This Page Intentionally Left Blank

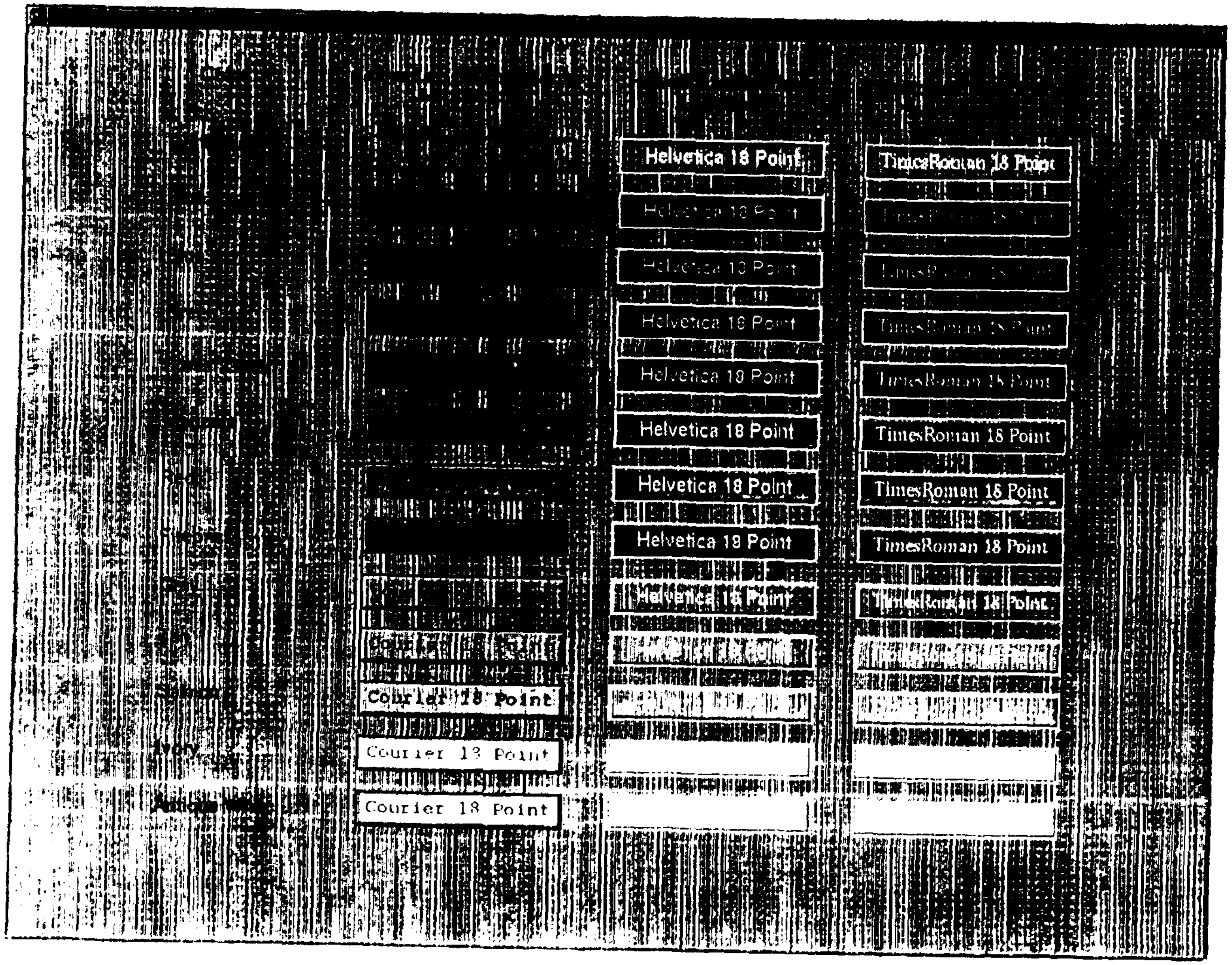


FIGURE 11.5B: Examples of Background/Foreground Color Combinations on Text Legibility

This Page Intentionally Left Blank

Extensive coloring creates a brighter-than-necessary display, with subsequent negative impact on user performance.

11.2.4.5 Extensive Background Coloring

Avoid using extensive coloring (e.g., many different colors) for the background, segments of the background, or particular regions surrounding individual characters or symbols. See Figure 11.6.

11.2.5 Color Selection and Ambient Illumination

The level of ambient illumination directly affects the perceived brightness and hue of a color. Consider the following when designing a color display.

11.2.5.1 Using Green

Green provides good general visibility over a broad range of intermediate luminances.

11.2.5.2 Using Red

Use red under high ambient lighting but not in low lighting.

11.2.5.3 Using Yellow

Yellow provides good general visibility over a broad range of luminances.

This Page Intentionally Left Blank

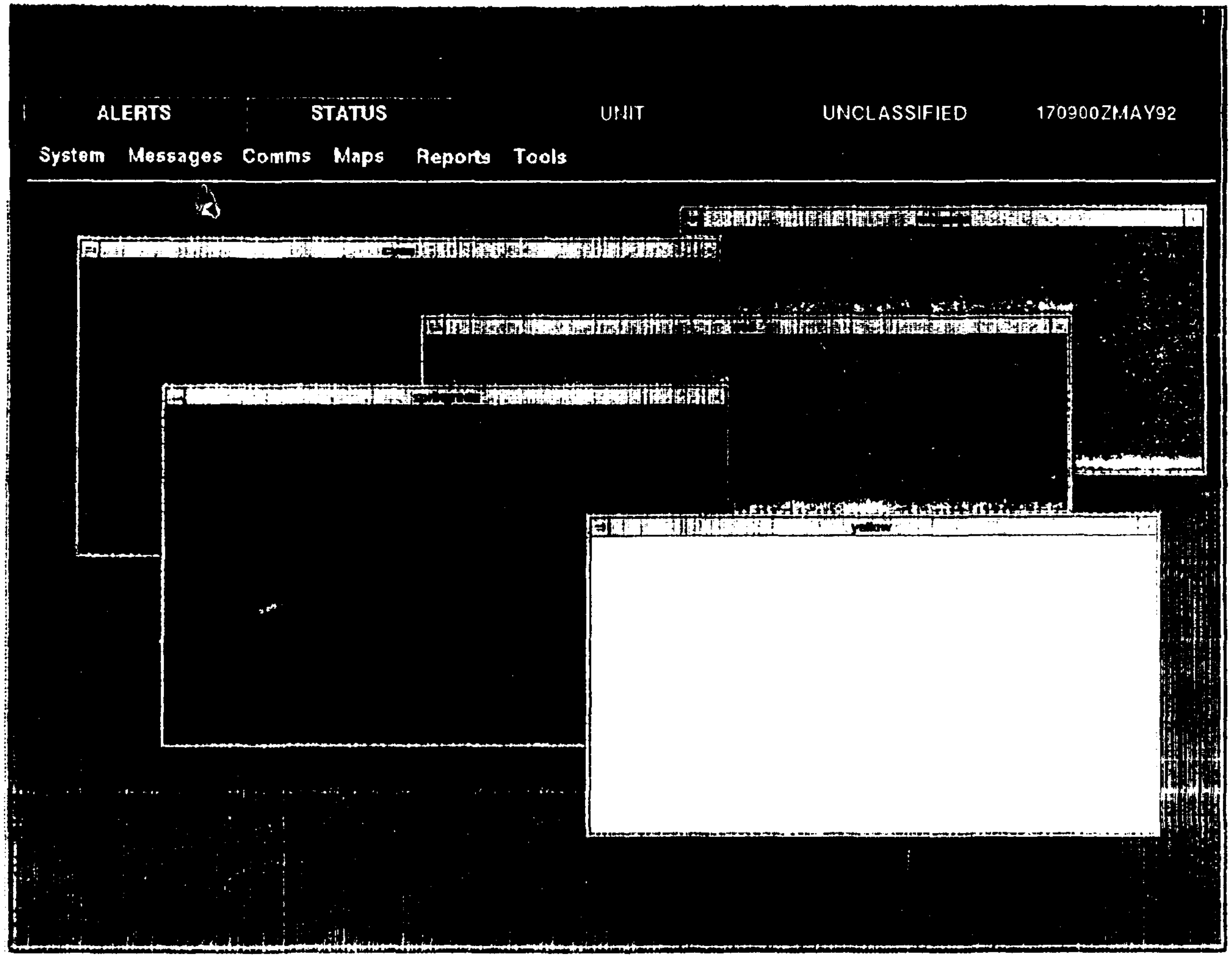


FIGURE 11.6: Extensive Coloring of Backgrounds Causes the Display to Become Much Brighter, Reducing User Performance

This Page Intentionally Left Blank

11.2.6 Specific Color Meanings

Use the colors and associated meanings listed in Table 11.3 for designing military color coding. The military intelligence community uses the classification Bar Colors listed in Table 11.4.

TABLE 11.3: Color Codes and Associated Meanings

Color	Meaning
Green	Non-alert, forces or situation at acceptable condition, obstacles on map graphics.
Blue	Friendly forces symbology.
Red	Alert, forces or situation at critical condition, enemy symbology.
Yellow	Forces or situation at marginal condition, caution, NBC areas on map graphics.
Black	Friendly forces symbology.

TABLE 11.4: Classification Bar Color Codes and Associated Meanings

Bar Color	Meaning
Green	Unclassified
Blue	Confidential
Red	Secret
Yellow	Top Secret
Purple	Sensitive Compartmental Information

11.2.7 Using Blue

Blue as a background color is most effective for tasks performed at close distances.

11.2.7.1 Using Saturated Blue on a Dark Background

Because the eye is relatively insensitive to blue, blue lines or dots will be very difficult to resolve. Avoid using saturated blue for small lines or dots when the background is dark.

11.2.7.2 Saturated Blue for Background Color

Use saturated blue only for background features in a display, not for critical data.

11.2.8 Use of Color Keys

If the screen designer must deviate from the color meanings provided in Table 11.3 or use other colors, a key that explains the color meaning (Figure 11.7) should be included on the display. This key should:

a. be readily visually accessible on the display without having to scroll or expand the screen or window

b. appear in the color being defined.

11.2.9 Large Screen Display Periphery Colors

Avoid the use of red and green in the periphery of a large-scale display. Yellow and blue are good periphery colors.

This Page Intentionally Left Blank

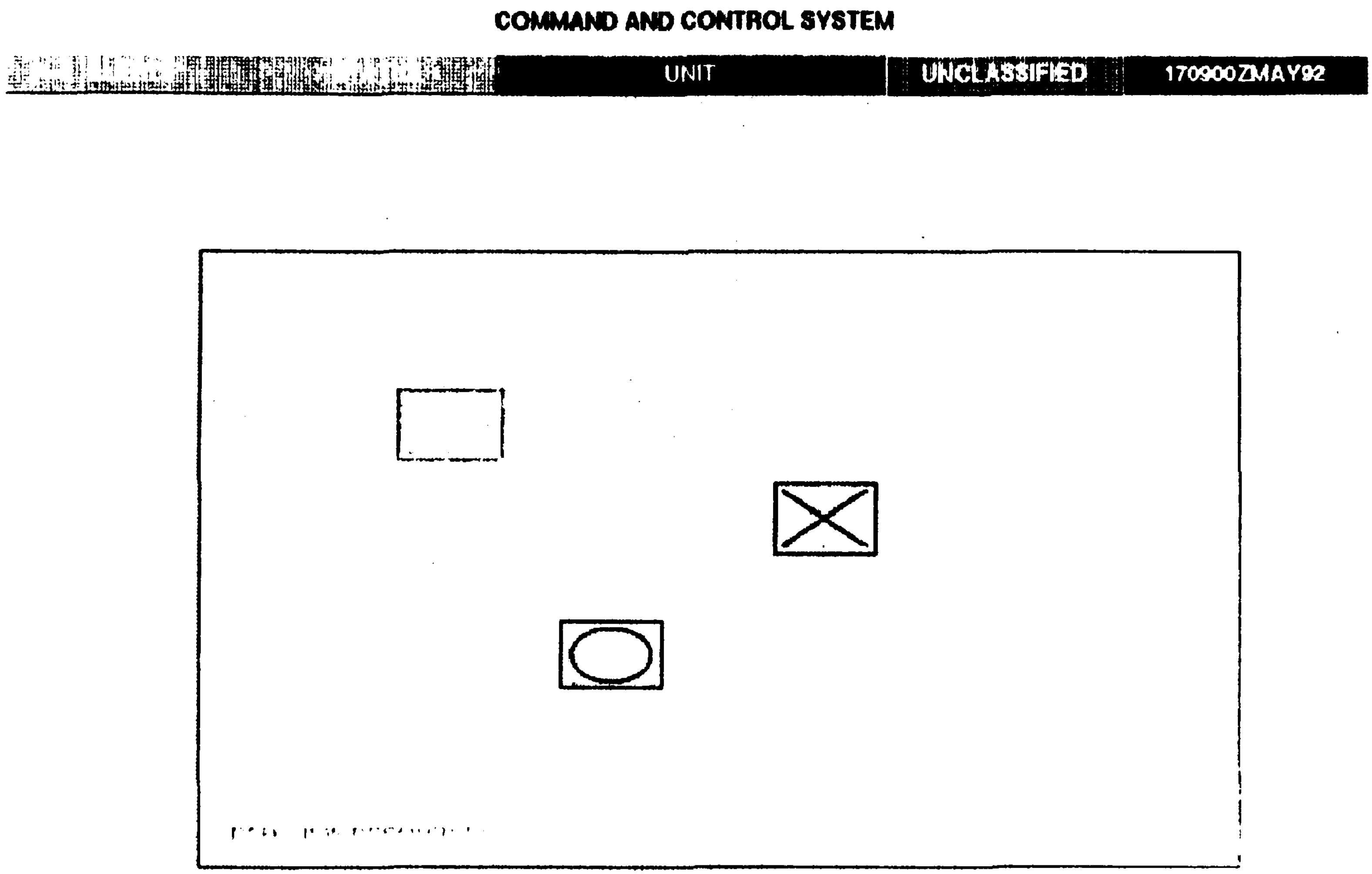


FIGURE 11.7: Visible Color Keys When Color Coding is Used in Ways that Differ from the User's Expectation

This Page Intentionally Left Blank

11.3 TONAL COLOR CODING

11.3.1 Color Coding for Relative Values

When relative rather than absolute values of a variable are important, display gradual color changes of a single color as a tonal code to show the relative values of a single variable. Display a monochromatic shading rather than spectral codes (different colors). See Figure 11.8.

11.3.2 Ordered Coding

Where different areas of a map are coded by texture patterns or tonal variation, order the assigned code values, such that the darkest and lightest shades correspond to the extreme values of the coded variable.

This Page Intentionally Left Blank

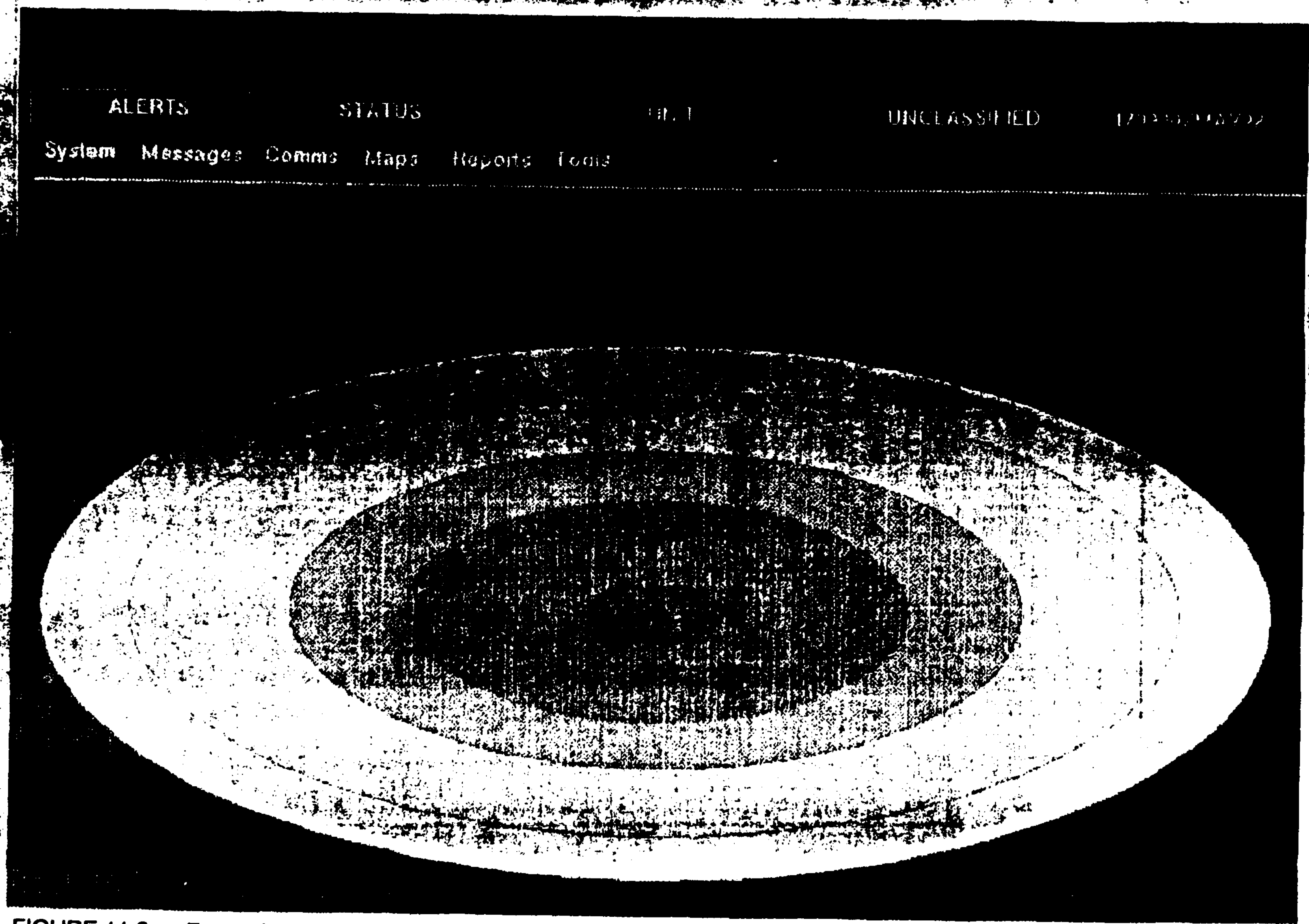


FIGURE 11.8: Example of How Tonal Changes Can Indicate Gradual Changes in Relative Value, in this Case Radiation Contamination Levels

This Page Intentionally Left Blank

11.4 COLOR-CODED SYMBOLS

Symbols that are color coded should use the following guidelines:

11.4.1 Color-Coded Symbol Size

Color-coded symbols should subtend a minimum of 20 minutes of visual arc. The designer must determine the maximum viewing distance from the display, then calculate the minimum size of the object, using the formula:

$$\text{Visual Angle (Min.)} = \frac{(57.3)\ (60)L}{D}$$

where L = size of the object, and D = distance from the eye to the object. See Figure 11.9.

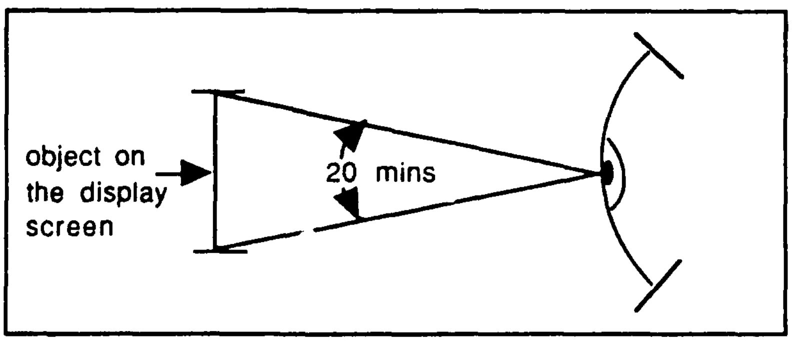


FIGURE 11.9: Visual Arc Subtended

11.4.2 Color-Coded Symbol Brightness

Color-coded symbols should have a minimum luminance of one foot-lambert.

11.4.3 Refresh Rates

The minimum refresh rate for color-coded symbols should ensure a flicker-free display.

11.5 MAP GRAPHICS AND COLOR

11.5.1 Functional Versus Decorative Color Coding

On map graphic displays, use color coding that provides a specific meaning to the user, rather than colors that are decorative only. These meanings should be in accordance with FM 21-26 (e.g., green for vegetation, brown for topographic relief, etc.)

11.5.2 Differences in Color Perceived Distance

The designer should be aware of how different colors focus at different distances relative to the user's retina due to wavelength. To the user, different colors will appear to be closer than others, especially the more saturated colors (see Paragraph 11.2.4.1).

This Page Intentionally Left Blank

REFERENCE LIST

Paragraph	References
11.1	Lickteig (1989) p.10; Nes (1986) paras 4.2.3 and 4.2.4; Galitz (1984) p. 122; Brown et al. (1983) para 7.2; Lickteig (1989) p. 10; Shneiderman (1987) p. 341; Bailey (1982) p. 421; Lewis and Fallesen (1989) p. 20
11.1.2.1	Smith and Mosier (1986) para 2.6-29
11.1.2.2	Shneiderman (1987) p. 72 and 337; Tullis (1988) p. 390
11.1.2.3	Lewis and Fallesen (1989) p. 23; Nes (1986) para 4.1.1; DoD (1989) para 9-1.3.3; Galitz (1984) p. 126 and 127; Slominski p. 4; Matthews (1987); Sidorsky p. 6.3-15 q
11.1.2.4	Galitz (1984) p. 126-127
11.1.2.5	Sidorsky p. 2.3.6 q-6,7; Galitz (1984) p. 121; Bailey (1982) p. 43; Lewis and Fallesen (1989) p.25
11.1.2.6	Smith and Mosier (1986) para 2.6-30; Brown para 7.4; DoD (1989a) para 5.4.1.4.5.5; Bailey (1982) p. 63, Lewis and Fallesen (1989) p. 20; Lickteig (1989) p. 10
11.1.2.7	Brown et al.(1983) para 7.7.6

REFERENCE LIST (Cont.)

Paragraph	References
11.1.2.8	Bowser (1991) p.18; Lewis and Fallesen (1989) p. 21; HFS (1988)
11.2.1.1	Smith and Mosier (1986) para 2.6-27
11.2.1.2	Galitz (1984) p. 125; Lewis and Fallesen (1989) p. 22
11.2.1.3	Galitz (1984) p. 123
11.2.1.4	Lewis p. 22
11.2.1.5	Smith and Mosier (1986) para 2.6-26, 31; Sidorsky p. 6.3-15 o-1; Brown para 7.6.1; Nes (1986) para 4.2.2; Galitz (1984) p. 122; DoD (1989) para 5.15.3.3.7; Lickteig (1989) p. 10; Shneiderman (1987) p. 339
11.2.2.1	Bowser (1991) p. 18; Brown (1983) para 7.6.2; Galitz (1984) p. 125; Shneiderman (1987) p. 340, Lewis and Fallesen (1989) p. 21; Bailey (1982) p. 263
11.2.2.2	Bailey (1982) p. 246

REFERENCE LIST (Cont.)

Paragraph	References
11.2.2.3	Smith and Mosier (1986) para 2.6-32; Sidorsky p. 6.3-15 o-2 and 2.3.6 q-3, 4; Brown (1983) para 7.7.1; Galitz (1984) p. 125; DoD (1989) para 5.2.2.1.18; Shneiderman (1987) p. 340; Hamel p. 5; Bailey (1982) p. 246, Lickteig (1989) p. 10
11.2.3.1	Smith and Mosier (1986) para 2.6-28; Brown (1989) para 7.1; Galitz (1984) p. 127; Lickteig (1989) p. 10; Smith and Mosier (1986) para 2.6-28; Chao (1987) p. 361; Lewis and Fallesen (1989) p. 20
11.2.3.2	Galitz (1984) p. 127; Nes (1986) para 4.2.2 and 4.2.5; Slominski p. 4; Shneiderman (1987); p. 338
11.2.3.3	Sidorsky p. 6.3-15 o-3; Galitz (1984) p. 125; Lickteig (1989) p. 10; Bailey (1982) p. 421; Shneiderman (1987) p. 71
11.2.4.1	Lewis and Fallesen (1989) p. 22
11.2.4.2	Galitz (1984) p. 126
11.2.4.3	Galitz (1984) p. 126

REFERENCE LIST (Cont.)

Paragraph	References
11.2.4.4	Lewis and Fallesen (1989) p. 22; Shneiderman (1987) p. 341; Snyder (1988) p. 465; Sidorsky para 2.3.6; Thorrell p. 2 and 3
11.2.4.5	Shneiderman (1987) p. 341; Snyder (1988) p. 465; Matthews (1987) p. 23
11.2.4.7	IBM (1984) p. 19
11.2.5a	Galitz (1984) p. 127
11.2.5b	Galitz (1984) p. 127
11.2.5c	Galitz (1984) p. 127
11.2.6	Sidorsky p. 2.3.6 q-3 and 4; DoD (1989) p. 256; U.S. Department of the Army (1985) p. 2.2
11.2.7	Lewis and Fallesen (1989) p. 23
11.2.7.1	Snyder (988) p. 465
11.2.7.2	Smith and Mosier (1986) para 2.6-34; Brown (1983) para 7.7.5; Galitz (1984) p. 127
11.2.8	Sidorsky p. 6.3-15 o-4; Brown para (1983) 7.6.1; Nes (1986) para 4.2.4; Galitz (1984) p. 123; Lickteig (1989) p. 10; U.S. Department of the Army (1985) p. 2-2
11.2.9	Lewis and Fallesen (1989) p. 22

REFERENCE LIST (Cont.)

Paragraph	References
11.3.1	Smith and Mosier (1986) para 2.6-25
11.3.2	Smith and Mosier (1986) para 2.4.8- 7
11.3.3	Smith and Mosier (1986) para 2.4.8-8
11.4.1	Durrett (1987) p. 186; Van Cott and Kinkade (1984) p. 47
11.4.2	Breen et al. (1987) p. 207
11.4.3	Breen et al. (1987) p. 209
11.5.1	Olson (1987) p. 207; U.S. Department of the Army (1987)
11.5.2	Olson (1987) p. 209

12.0 DECISION AIDS

Military command and control systems continue to develop automated decision aids in support of military operations. Although what constitutes a decision aid has been debated, it is important to point out that decision aids assist, rather than replace, human decision-makers. Consequently, when defining decision aids, applications limited to managing information are usually excluded, as are those that make decisions in a fully autonomous mode.

Decision aids may be designed to be parts of other software or as stand-alone applications. As an example, decision aids have been designed to assist users in evaluating military courses of action. These applications present alternatives and supporting evidence, as well as assist the user in evaluating the alternatives. The user retains a major role in developing the final recommendations. Information management software such as database management, text processing, and graphics applications may support the decision process but are not usually considered decision aids. Other applications, such as engine diagnostic software, may include many relatively fixed rules derived from human experts. Such "expert systems," can include many properties of decision aids, but they place relatively more emphasis on internal rules to arrive at conclusions. Examples of autonomous systems include automatic fire control systems and robotic devices. These systems may require human supervision but rely heavily on internal rules and algorithms for their operation.

It is difficult to make a distinction between decision aids and expert systems (which include autonomous capabilities), since both require cooperation between human and automated system components. Holtzman (1989) differentiates between expert and intelligent decision systems and points out which is appropriate to use based on subject matter, circumstance, and preference of the decision-maker. Expert systems may have a relatively large knowledge base and rules that respond to constant environmental factors, whereas decision aids place more burden on the decision-maker. Decision aids provide assistance and are designed to help in uncertain or novel situations. The guidelines presented in this section are appropriate for both expert systems and decision aids. Therefore, the term "decision aid" or "aid" will be used to refer to both types of decision support application.

The question of "when to use decision aids" is reviewed in the first part of this section. Given that decision aids are to be used, the next step is defining the requirements. When the requirements are firm, the features needed to support the requirements become important. This section then deals with specific issues of decision aid interface design.

12.1 WHEN TO USE DECISION AIDS

Decision aids should be used to compensate for known limitations in human decision-making and to offset the adverse effects of external factors. In general, difficulties can arise because of the fundamental limits of human cognitive (i.e., mental) abilities and lack of experience. Difficulties also arise because of various environmental factors that both stress the decision-maker and determine the type, quantity, quality, and rate of information presented.

12.1.1 Cognitive Considerations

Consider the cognitive limitations and styles of decision-makers when designing decision aids. For example, overload (i.e., stress, information, situational, etc.) often causes users to focus on a subset of the available information. Innate abilities and learned information-processing strategies may cause additional problems. The following points describe several commonly occurring limitations.

12.1.1.1 Cognitive Limitations

Novices, individuals lacking confidence, and those performing tasks under stressful conditions will make errors that may result in less-than-optimal decisions. For example:

a. Humans often have difficulty retrieving, retaining, representing, and manipulating large amounts of information. They also may have difficulty combining multiple cues or criteria or performing computational tasks. These difficulties result in delaying performance or avoiding difficult tasks.

b. Humans often have difficulty making decisions in times of uncertainty.

c. Novices do not have previous experience, so they often fail to recognize errors.

d. When making a decision, humans often simplify decision problems by selectively perceiving data, information, and knowledge. They may set outcome objectives and then look for a decision that meets them. Thus, they may focus on confirming rather than on refuting evidence. They also may adjust decision methods to fit goals or desired results.

e. If a decision leads to a negative result, humans may attribute the outcome to chance or to the complexity of the problem, rather than to their own decision-making deficiencies.

f. Humans have limited memory available for current tasks; they will lose some information within seconds.

g. Humans have limited abilities to organize information.

h. Humans usually have difficulties with symbolic and quantitative manipulation of mental representations, and they may have difficulty formulating or dealing with abstractions.

i. Humans may have difficulty extrapolating time and space information.

j. Humans may fail to use prior experience to generalize in new situations.

12.1.1.2 Avoiding Complexity

Humans often have difficulty dealing with complexity and may, therefore, try to make a problem less complex by avoiding certain aspects of it. Consequently, they may not consider all factors when making decisions. Some strategies used to make decisions less complex are:

a. Humans often simplify decision problems by only considering a few alternatives.

b. Humans may use only part of the available information. Or they may use information that corresponds to a mental representation or model of what they imagine the solution to be, even if this means rejecting or misperceiving relevant information. They may combine or "chunk" information in various ways, rely on poor memory-search strategies, or rely on erroneously perceived correlations between data.

12.1.1.3 Cognitive Biases

Humans have biases that can carry over into the decision-making process.

a. Humans may recall information that has been recently acquired, frequently rehearsed, or semantically related to current information.

b. Humans may anchor their judgments (i.e., place greater emphasis on early evidence) and then fail to adjust when provided new information.

c. Humans may give preference to information they believe is causally related to the problem.

d. Humans may provide numeric judgments that contain systematic bias or variance.

e. Humans may select cues that are often unreliable indicators of the true situation.

f. Humans may use inappropriate analogies to generate and compare options.

g. Humans may incorrectly identify current situations with similar past events.

h. Humans may fail to detect unique features among similar cases, and inconsistent or ambiguous information may not be noticed or emphasized appropriately.

12.1.1.4 Time Allocation

Humans may fail to allocate time properly to different phases of the planning process. Too much attention to early stages of planning may leave inadequate time to evaluate derived alternatives properly.

a. Humans may perform detailed analysis early but fail to do so later.

b. Humans may fail to develop and evaluate the alternatives thoroughly.

c. Humans may fail to identify, evaluate, compare, and combine salient information and, therefore, fail to identify, prioritize, and assess goals.

d. Humans may fail to model (war-game) alternatives because of lack of time.

12.1.2 External Factors

Many external factors may influence the quality of decisions.

12.1.2.1 Information Overload

When the complexity, dynamics, and/or volume of information to assess are high (such as in battlefield operations), they may degrade decision-making performance.

12.1.2.2 Time Stress

Humans have difficulty analyzing information fast enough to meet external time constraints. When they have enough time, they often have difficulty maintaining high performance long enough to analyze all data.

12.1.2.3 Limited Information

Decision-makers may work in situations where information available to support their decisions is limited. Under such circumstances, lack of experience and human limitations in making or formulating estimates may pose problems.

12.1.3 Training

Decision-makers may not have the experience, deductive skills, or knowledge of the procedures necessary to make a decision. A decision aid can assist by performing some steps, by leading the human through the required steps, and by filling in knowledge gaps. Properly designed decision aids also train users through explanation and embedded training.

12.1.4 When to Use Decision Aids

Use decision aids to help the user overcome the difficulties previously described. For example:

12.1.4.1 Manage Complexity

Decision aids help the user cope with information overload; they focus attention. Use decision aids when the user is trying to manipulate large amounts of data or visual representations, when combining multiple criteria, when allocating resources, when managing detailed information, and when selecting and deciding among alternatives.

12.1.4.2 Improve Timeliness

A decision aid helps a user perform many time-consuming activities more quickly. Some examples include the diagnosis of the current state of a system and mathematical calculations, particularly when they are beyond the user's abilities. Providing aid when users encounter unfamiliar problems also helps improve the timeliness of the process.

12.1.4.3 Best Use of Limited Data

Use decision aids when there is uncertainty due to limited data. Decision aids help by predicting future events from limited information, by improving the accuracy and reliability of critical tasks, and by addressing critical areas beyond the ability of the user.

12.1.4.4 Overcoming Limitations

Use decision aids to overcome the human cognitive limitations described in the earlier sections. For example:

a. Use decision aids to overcome human limitations in dealing with uncertainty.

b. Decision aids are helpful in overcoming emotional components of decision-making.

c. If the quality of human performance is in question, decision aids can add greater accuracy to the process.

d. Decision aids are ideal in cases where memory- and information-retention problems exist.

e. Decision aids overcome cognitive biases very well.

12.1.5 When to Consider Alternatives

The following are circumstances under which the use of decision aids may not be advisable:

12.1.5.1 Obvious Solutions

Do not use decision aids when solutions are obvious or when an alternative clearly dominates all other options.

12.1.5.2 Time Requirements

Use decision aids only when sufficient time is available or when the user is authorized to make decisions.

12.1.5.3 Generalizing

As appropriate, defer to the human ability to generalize.

12.1.5.4 Adaptation

Recognize situations where individuals may be superior in adapting to novel situations.

12.1.6 Cautions and Limitations

Exercise caution when introducing decision aids, in particular, when they include functions that reduce the role of human judgment.

12.1.6.1 User Complacency

The decision aid could encourage users to take a less active role. This, in turn, could cause users to be inattentive and less prepared to handle sudden decreases or increases in work load, both of which may reduce accuracy.

12.1.6.2 Continued Vigilance

If the user's role becomes less active, the user may have difficulty maintaining sustained attention, which could lead to longer user response times.

12.1.6.3 Discrimination Limitations

It is necessary to recognize limitations in a user's ability to discriminate between correct and incorrect automated decisions.

12.1.6.4 Fear of Automation

Many users mistrust automation and automated decisions, preferring to believe their performance is superior to that of automated systems. User attitudes toward automation are often based on the fear of being replaced. The designer should take this into account when planning the role and degree of authority the user will have in overriding automated decisions.

12.2 DEFINING DECISION AID REQUIREMENTS

Develop decision aids or expert systems that focus on tasks that users find difficult, rather than on what is already done routinely.

12.2.1 Understand Tasks

Base designs on an in-depth understanding of both the tasks to be performed and the conditions of their performance.

12.2.1.1 Experts

The best way to define decision aid requirements is to start with experts in the field. However, it is important to choose the experts appropriately. Be sure to use more than one expert and verify that they really are experts. If they have knowledge of part of the field, be sure to consider those parts, and find other experts for the other parts. Also, ensure that common users participate with the group of experts. When obtaining information from the experts, provide a means to identify the criteria used to reach decisions.

12.2.1.2 Operating Conditions

Decision aids must be matched to operational conditions and limitations they are designed to support.

12.2.1.3 Appropriate Problems

Recognize that not all functions are appropriate for decision aids. Determine the appropriate functions and design them to be compatible with the user's decision processes.

12.2.1.4 Efficiency

Provide no more than one aid for each task.

12.2.2 Requirements

Decision aid development should be driven by requirements, not by technology.

12.2.2.1 Intended Users

Identify areas where users actually need help, then match the decision aid to the needs of the intended users.

12.2.2.2 Highest-Level Goal

Recognize the user's decision situation and goals, and focus on the highest-level goal.

12.2.2.3 Perceived Utility

Anticipate skepticism concerning automated decision support. Recognize that the dominant factor in accepting decision aids is perceived utility. The system must add new capabilities or increase efficiency in the performance of decision-making tasks.

12.2.2.4 User Characteristics

Consider characteristics of the user population in designing the decision aid and its interface.

12.2.3 Types of Aids

Types of aids and presentation formats may vary according to the phases of the decision process (i.e., alerting, acquisition, evaluation, and responding) and factors such as time stress.

12.2.4 Function Allocation Between Humans and Computers

Allocating functions between humans and computers must be based on cognitive task analysis, not on what is achievable using current technology.

12.2.4.1 Performance

Recognize that aided performance may not exceed unaided performance, even though aided methods are preferred.

12.2.4.2 Work Load

Decision aids and expert systems can enhance decision quality. However, they may increase the user's work load because users may be required to consider more variables. Seek design alternatives that prevent or minimize increased work load.

12.2.4.3 Complete

The aid must be complete for its intended purpose. All critical aspects of the decision situation must be addressed.

12.2.4.4 User Behavior

Recognize that the user's decision-making behavior is contingent upon the task and the context within which it is performed. Design the decision aid to provide decision methods suitable to probable variations in tasks and context.

12.2.4.5 User Preference

Users often prefer to perform some of the tasks and allow the decision aids to perform others. Specifically, users prefer to do the easy to moderately difficult tasks and leave difficult tasks to the decision aid. This interaction is necessary to maintain user interest and attention. Decision aids are more acceptable to users if viewed as advisors rather than decision-makers.

12.2.4.6 Meaningful Applications

Avoid applications that are trivial or lack complexity because they may undermine the value of automated decision support methods.

12.3 FEATURES OF DECISION AIDS

12.3.1 General Design Considerations

12.3.1.1 Ease-of-Use

A decision aid should be easier to use than the decision process it replaces. It must be flexible, versatile, and easy enough to benefit typical users (i.e., users don't need to be subject matter experts). A decision aid must use terminology and criteria appropriate to the target user group. It must be easy to control and understand.

12.3.1.2 Provide Timely Advice

A decision aid must be capable of responding to the user's ad hoc requests in time to allow the information to influence decisions. The interface should facilitate the exchange of information.

12.3.1.3 Match Available Resources

Tailor decision aids to the resources available to the user.

12.3.1.4 Automatic Discovery

A decision aid should automatically identify meaningful patterns and relationships and bring them to the attention of the user.

12.3.2 Provide Decision Alternatives

12.3.2.1 Application Requirements

Decision aids must be able to support development and evaluation of multiple, feasible alternatives. The aid should present a set of possible alternatives, each of which could be feasible. However, the aid should not display all of the options when that would be too complex. The decision aid also should display which goals are served by the different alternatives and applicable options.

12.3.2.2 User-Oriented Requirements

The decision aid must support user evaluation of decision options. First, the aid should generate alternatives for the user to evaluate and should allow the user to input his or her own alternative(s). Second, the aid should have a method of assigning and explaining probabilities for alternatives. The user should be able to explore different solutions, including using different decision strategies and criteria. Once the user has applied all desired options, the aid should rank-order the decision alternatives. This assistance also should include guidance in using rating procedures.

12.3.3 Prediction, Simulation, and Modeling

12.3.3.1 Prediction

The application should be able to predict future data. Historical data should be available to make comparisons, search for precedents, and assist the user in visualizing trends. The decision aid should alert the user when it predicts a future problem or opportunity upon which the user needs to act.

12.3.3.2 Simulation and Modeling

Provide a modeling and simulation capability to support "what if?" exercises and to make predictions based on current conditions.

12.2.3.3 Validate Models

Models used in decision aiding must be appropriate and validated.

12.3.4 Identify and Assess Factors Underlying Decisions

12.3.4.1 Multiple Criteria

Provide a means of obtaining and assessing weights for multiple criteria. Multiple criteria should be statistically independent, when possible. As appropriate, provide a means of combining weights from multiple sources. This refers to the technique of multi-attribute decision-making.

12.3.4.2 Causal Factors

Identify and rank causal factors by their importance and assign weights. The application should allow users to modify the decision factors and their weights and to provide and adjust risk factors used in decision models. This refers to techniques such as pair-wise comparison.

12.3.4.3 Explain Underlying Factors

The aid should be able to explain the contributions of underlying factors and support the use of sensitivity analysis for exploring those contributions. The aid must identify and assess operational constraints and provide a means of informing the user (upon request) of decision-aid boundaries or other limitations. The aid should make available to the user the assumptions underlying modes and parameters and a history of its past performance.

12.3.4.4 User Input

The decision aid should make it easy for the user to provide input into the aid's decision. The user should be able to add new decision factors and set the range of conditions (within the decision aid's set limits), the level of output detail, and the parameters for optimization. Provide a means for saving and reusing the user's modifications, but also provide a means to return to the default settings.

12.3.4.5 Visualization

Assist in visualizing interacting factors.

12.3.4.6 Validity

Provide a means for assuring the validity of elements added to the decision model, in particular those used over successive applications.

12.3.5 Handling Decision Aid Recommendations

12.3.5.1 Display Outcomes of Options

The application should be able to calculate and display results of selected decision options.

12.3.5.2 Evaluate Risk

The application should provide facilities for assessing costs, risks, and benefits of all alternatives.

12.3.5.3 Explain Recommendations

For users to trust the decision aid, it is imperative the aid explain the rationale behind outputs or recommendations. It also should provide indicators of certainty or uncertainty when making recommendations.

12.3.5.4 Identify Missing Data

When data are missing or uncertain, the aid should identify this situation and give information on the possible impact on the recommendations.

12.3.5.5 Check for Consistency

The decision aid should include internal consistency checks to prevent the system from making contradictory predictions and recommendations.

12.3.5.6 Failure Notification

The decision aid must inform the user when it cannot handle the current situation.

12.4 USER REQUIREMENTS

12.4.1 General Considerations

12.4.1.1 User-Friendly

The decision aid must be user-friendly and intelligent and must present information that is readily understood by or familiar to the users. Where possible and appropriate, the decision aid also should have sufficient "intelligence" to adjust to user task requirements.

12.4.1.2 Decision Methods

The decision aid should use decision methods acceptable to the decision-maker and should also be able to accommodate user changes. Once the decision method is determined, the user must retain control throughout the process. The aid should provide feedback on the method and the current stage of processing.

12.4.1.3 Data Entry

Reduce the user's data-entry requirements as much as possible. To do so, set defaults for data-entry fields. However, these defaults and fields must be user-changeable.

12.4.1.4 Alerts

A decision aid should automatically alert users to important new developments occurring in the database or as a result of predictive modeling.

12.4.1.5 User Cooperation

The system should encourage the user to participate in the decision process. To do this, the system should represent problems and solutions in the same way the users do. The system also should try to foster user "ownership" of decisions and allow the user to exercise judgment over the decision aid results. This includes providing sufficient information to the user both about the process and about the end result.

12.4.1.6 Guidance

The decision aid should guide the user through the process. It should provide automated guidance on how to define and analyze a problem and formulate a decision. When user input is required, the decision aid should aid in making this requirement clear. However, it should not make the user dependant, such that the process cannot be completed when the system is unavailable.

12.4.1.7 Information Overload

Avoid presenting too much data. Use aids to reduce, filter, and preprocess data into a form useful to the decision-maker.

12.4.1.8 User Work Load

Avoid increasing the user's work load, when possible. Prepare users for changes and possible increases in work effort when necessary and point out the aid's abilities to increase effectiveness.

12.4.1.9 Information Presentation

Reducing complexity is a major reason for using decision aids. Therefore, some guidelines are necessary on the amount of information to present to the user. In general, the system should provide information required to perform the tasks allocated to the user; however, it should only present information relevant to the task being performed. The system should provide no more information than is essential and should avoid repeating already available information. Present information using a level of abstraction, resolution, or detail appropriate to the immediate task.

12.4.1.10 Time Requirements

When time is limited, systems should anticipate the user's needs and provide a greater degree of autonomous decision-making.

12.4.1.13 Types of Procedures

Provide procedures appropriate to the user's level of expertise. Designers should recognize that experts may use mental imagery; novices depend more on rule-based procedures.

12.4.2 Decision-Aid Interface

12.4.2.1 Intelligent Interfaces

The user-machine interface should support an intelligent dialogue between the user and the decision aid. For example, it should adapt to the user; understand the user's goals, needs, and abilities; interpret poorly formulated queries; correct user errors; and overcome user limitations. The interface also should reflect the tasks to be performed and should be tailored to the resources available.

12.4.2.2 Errors

The system should help prevent the user from making errors. When errors are made, it should provide automatic error recovery.

12.4.2.3 Human Factors Issues

User-interface design guidelines mentioned elsewhere in this document also apply.

12.4.2.4 Display Formats

The decision aid should allow users to customize formats to their own needs. However, it is preferable to minimize the user's requirements to make such changes. To do this, the application should associate and group data in a meaningful way, and displays should match the task.

12.4.2.5 On-Line Help

Detailed information on HELP applications is available in Section 14 of these *Guidelines*.

12.4.3 Explanations

The decision aid should be capable of providing domain-specific explanations to answer user questions. Decision aids must be capable of guiding the user through the decision process, as well as providing procedural help on use of the system.

12.4.3.1 User's Point of View

When the system provides explanations, ensure they are easy for the user to understand. Explanations should use terms familiar to the user, incorporating the user's concept of the problem and maintaining consistency with the immediate task. Intuitive explanations or analogies are helpful for topics that are likely to be too difficult for the user to understand.

12.4.3.2 Keep it Short

Length of explanation is important. Provide a short explanation initially, with the ability to provide more detail at the user's request. Consider how much to tell the user. Weigh trade-offs in what the user can learn about the decision aid and what the decision aid can/should explain to the user.

12.4.3.3 Easy Access to Explanations

Assist the user in locating key elements of the decision model, as related to a specific decision task.

12.4.3.4 Document Algorithms

Be able to explain the current decision model or method and be prepared to justify the use of component factors. Document the decision aid's algorithms, and make them available for user inspection.

12.4.4 Training

12.4.4.1 Backup

Provide backup systems and appropriate training in the performance of any user tasks replaced by decision aids. When decision aids are available, provide regular training to the user in all skills required to maintain proficiency on backup systems. This training will be necessary if decision aids become unavailable. Training may be preferable to using decision aids for handling infrequent critical events occurring in dynamic environments.

12.4.4.2 Teach Limitations

Users must be trained to recognize inappropriate uses of the aid and to recognize errors. Provide readily accessible lists of limitations; include information concerning limitations and errors in embedded training. Users should learn not to categorically accept a decision aid's capabilities.

12.4.5 Decision Graphics and Displays

12.4.5.1 User-Oriented

Prepare graphics, textual reports, and input screens in formats familiar to the user. This will facilitate rapid and accurate information-processing. However, the user should be able to control formats or to select from alternate preprogrammed formats.

12.4.5.2 Graphic Design

Graphics are another important part of the user interface. Graphics help assist the user in visualizing information. However, guard against inaccurate graphics, as they can have a strong negative impact.

12.4.5.3 Graphic Coverage

Provide historical displays of comparative cases, to include time-sequenced presentations.

12.4.5.4 Other Formats

Use spatial rather than textual formats when the task involves extensive spatial processing, in particular when task performance time is limited. Use tables rather than graphs when reading specific data points.

12.5 ORGANIZATIONAL FACTORS

12.5.1 Information Requirements

Decision aids must be flexible in meeting the different *information requirements at different echelons.*

12.5.1.1 Detail

Different levels of command require different levels of abstraction. Decision aids must accommodate different levels of detail and time constraints at each echelon.

12.5.1.2 Location

Command ana control decision aids should be distributed (i.e., they should support multiple, cooperating decision-makers at different locations sharing a common database).

12.5.1.3 Entire Process

Where practical, decision aids should be designed to support the entire command and control process, rather than developed to support isolated phases of the process.

12.5.2 Entire Organization

Decision aid designs should consider impacts on the entire organization, particularly where organizational goals may supersede those of subordinate decision aid users.

12.5.3 Complementary

Decision aids should complement existing tasks and information-distribution systems.

12.6 FLEXIBILITY

12.6.1 Change-Over Time

Decision aids should be designed as adaptive systems (i.e., they must accommodate growth and evolve over time to meet changing conditions, doctrine, etc.).

12.6.1.1 Policies for Change

Establish policies for implementing changes, as well as the mechanisms for those changes.

12.6.1.2 Changing Preferences

Adjust to changing situations and user preferences (different circumstances and users may require different methods).

12.6.2 Maintainability

Decision aids should be maintainable by the user. Rules, data, and decision logic should reflect current needs.

12.6.3 Type of Support

Allow the user to tailor the type of support provided by the decision aid in the presence of changing conditions.

This Page Intentionally Left Blank

REFERENCES

Paragraph	Reference
12.1.1.1	Zachary (1988) pp. 997-1030; McKeown et al. (1991); Ehrhart (1990); Andriole and Adelman (1990)
12.1.1.2	Andriole and Adelman (1990); Zachary (1988) pp. 997-1030; Walrath (1989)
12.1.1.3	Zachary (1988) pp. 997-1030; Walrath (1989); Andriole and Adelman (1990); McKeown et al. (1991); Ehrhart (1990)
12.1.1.4	McKeown et al. (1991)
12.1.2.1	Zachary (1988) pp. 997-1030; Lysaght et al. (1988)
12.1.2.2	McKeown et al. (1991); Zachary (1988) pp. 997-1030; Lysaght et al. (1988)
12.1.2.3	McKeown et al. (1991); Lysaght et al. (1988)
12.1.3	McKeown et al. (1991); Lysaght et al. (1988)
12.1.4.1	Holtzman (1989); Lysaght et al. (1988); McKeown et al. (1991); Walrath (1989)
12.1.4.2	Walrath (1989); McKeown et al. (1991); Holtzman (1989)
12.1.4.3	McKeown et al. (1991); Walrath (1989); Lysaght et al. (1988); Klien and MacGregor (1988)

REFERENCES (Cont.)

Paragraph	Reference
12.1.4.4	Holtzman (1989); Klien and MacGregor (1988); McKeown et al. (1991); Walrath (1989)
12.1.5.1	Holtzman (1989)
12.1.5.2	Holtzman (1989)
12.1.5.3	McKeown et al. (1991)
12.1.5.4	McKeown et al. (1991)
12.1.6.1	Walrath (1989)
12.1.6.2	Walrath (1989)
12.1.6.3	Walrath (1989)
12.1.6.4	Walrath (1989); Gordon (1988) pp. 55-59
12.2.1.1	Gordon (1988) pp. 55-59; Thierauf (1988); Main and Paulson (1988)
12.2.1.2	Thierauf (1988)
12.2.1.3	Main and Paulson (1988)
12.2.1.4	Gordon (1988) pp. 55-59
12.2.2	Thierauf (1988)
12.2.2.1	Thierauf (1988); Holtzman (1989); Schwartz (1983) pp. 13-17; Andriole and Adelman (1990); LeMay (1988) pp. 227-229; Finke and Lloyd (1988) pp. 170-193
12.2.2.2	Urban (1990); Andriole and Adelman (1990); Thierauf (1988); McCann (1988); Main and Paulson (1988)
12.2.3.1	Urban (1990); Andriole and Adelman (1990); Thierauf (1988); Walrath (1989)

REFERENCES (Cont.)

Paragraph	Reference
12.2.3.2	Thierauf (1988); Tannenbaum (1990) pp. 54-59; Bidgoli (1989) pp. 27-34; Andriole and Adelman (1990); Urban (1990); Zachary (1988) pp. 997-1030
12.2.3.3	O'Keefe (1989) pp. 217-226; LeMay (1988) pp. 227-229
12.2.4.1	Minasi (1990) pp. 13-15; O'Keefe (1989) pp. 217-226
12.2.4.2	Holtzman (1989); Schmitz et al. (1990) pp. 29-38; Pew (1988) pp. 931-940; Holtzman (1989)
12.2.4.3	Holtzman (1989); Bidgoli (1989) pp. 27-34; Thierauf (1988); Pew (1988) pp. 931-940; Finke and Lloyd (1988) pp. 170-193
12.2.4.4	Andriole and Adelman (1990)
12.2.4.5	Finke and Lloyd (1988) pp. 170-193
12.2.4.6	Finke and Lloyd (1988) pp. 170-193; Minasi (1990) pp. 13-15; Tannenbaum (1990) pp. 54-59
12.3.1.1	Lysaght et al. (1988); McKeown et al. (1991); Schmitz et al. (1990) pp. 29-38; Pew (1988) pp. 931-940; Ma et al. (1989) pp. 996-1012

REFERENCES (Cont.)

Paragraph	Reference
12.3.1.2	Holtzman (1989); McKeown et al. (1991); Gordon (1988) pp. 55-59; Riedel (1988); McCann (1988); Zachary (1988) pp. 997-1030; Thierauf (1988); Walrath (1989); Ehrhart (1990); Pew (1988) pp. 931-940
12.3.1.3	McKeown et al. (1991); Osborn and Zickefoose (1990) pp. 28-35; Lysaght et al. (1988)
12.3.1.4	Holtzman (1989); Riedel (1988)
12.3.1.6	Holtzman (1989); Ma et al. (1989) pp. 996-1012; Pew (1988) pp. 931-940
12.3.1.7	Thierauf (1988); Schmitz et al. (1990) pp. 29-38; O'Keefe (1989) pp. 217-226; LeMay (1988) pp. 227-229; McCann (1988)
12.3.1.8	Gordon (1988) pp. 55-59; Riedel (1988)
12.3.1.9	Gordon (1988) pp. 55-59; Pew (1988) pp. 931-940; LeMay (1988) pp. 227-229
12.3.1.10	Pew (1988) pp. 931-940
12.3.1.11	O'Keefe (1989) pp. 217-226; Ma et al. (1989) pp. 996-1012
12.3.1.12	Schmitz et al. (1990) pp. 29-38; Finke and Lloyd (1988) pp. 170-193; Thierauf (1988); O'Keefe (1989) pp. 217-226; Lysaght et al. (1988)
12.3.1.13	Walrath (1989)

REFERENCES (Cont.)

Paragraph	Reference
12.3.2.1	Gordon (1988) pp. 55-59; Riedel (1988); McCann (1988); Bidgoli (1989) pp. 27-34
12.3.2.2	Gordon (1988) pp. 55-59; McCann (1988); Urban (1990); Zachary (1988) pp. 997-1030; Minasi (1990) pp. 13-15
12.3.2.3	Pew (1988) pp. 931-940; Finke and Lloyd (1988) pp. 170-193; Urban (1990)
12.3.2.4	Riedel (1988); Holtzman (1989); Pew (1988) pp. 931-940; Finke and Lloyd (1988) pp. 170-193; Main and Paulson (1988); McCann (1988); Thierauf (1988); Urban (1990)
12.3.2.5	Finke and Lloyd (1988) pp. 170-193; Urban (1990)
12.3.3.1	Lysaght et al. (1988)
12.3.3.2	McCann (1988); Ehrhart (1990); Holtzman (1989)
12.3.3.4	Lysaght et al. (1988)
12.3.4.1	Pew (1988) pp. 931-940; Ehrhart (1990); Schwartz (1983) pp. 13-17; Riedel (1988)
12.3.4.2	Holtzman (1989); Gordon (1988) pp. 55-59; Pew (1988) pp. 931-940
12.3.4.3	Holtzman (1989); Ehrhart (1990); Pew (1988) pp. 931-940; Tannenbaum (1990) pp. 54-59

REFERENCES (Cont.)

Paragraph	Reference
12.3.4.4	Holtzman (1989); Ehrhart (1990); Finke and Lloyd (1988) pp. 170-193; Riedel (1988)
12.3.5.1	Riedel (1988); Crolotte et al. (1980) pp. 1216-1220
12.3.5.2	Riedel (1988)
12.3.6.1	Thierauf (1988); Pew (1988) pp. 931-940; Ehrhart (1990)
12.3.6.2	Hoadley (1990) pp. 120-127; Jarvenpaa and Dickson (1988) pp. 764-775; Urban (1990)
12.3.6.3	Jarvenpaa and Dickson (1988) pp. 764-775; Thierauf (1988); Ehrhart (1990); Finke and Lloyd (1988) pp. 170-193; Bidgoli (1989) pp. 27-34; Zachary (1988) pp. 997-1030
12.3.6.4	Andriole and Adelman (1990)
12.3.6.5	Barnett (1990) pp. 1552-1556; Jarvenpaa and Dickson (1988) pp. 764-775
12.4.1	McCann (1988)
12.4.1.1	McCann (1988)
12.4.1.2	McCann (1988)12.4.1.3 McCann (1988)
12.4.2	Weingaertner and Levis (1988) pp.195-201
12.4.3	Weingaertner and Levis (1988) pp. 195-201; Riedel (1988)

REFERENCES (Cont.)

Paragraph	Reference
12.5.1	Thierauf (1988); Schmitz et al. (1990) pp. 29-38; Gordon (1988) pp. 55-59; McCann (1988)
12.5.1.1	McCann (1988)
12.5.1.2	Holtzman (1989)
12.5.2	Bidgoli (1989) pp. 27-34
12.5.3	Urban (1990); Gordon (1988) pp. 55-59
12.6	Mittal (1985) pp. 32-36
12.6.1	Pew (1988) pp. 931-940
12.6.1.1	Osborn and Zickefoose (1990) pp. 28-35; Mittal (1985) pp. 32-36; Schwartz (1983) pp. 13-17; Riedel (1988); McCann (1988); Bidgoli (1989) pp. 27-34; Zachary (1988) pp. 997-1030
12.6.1.2	Pew (1988) pp. 931-940
12.6.1.3	Zachary (1988) pp. 997-1030
12.6.1.4	Zachary (1988) pp. 997-1030
12.6.1.5	Riedel (1988)
12.6.2	Zachary (1988) pp. 997-1030
12.6.2.1	Mittal (1985) pp. 32-36
12.6.2.2	Zachary (1988) pp. 997-1030
12.6.2.3	Tannenbaum (1990) pp. 54-59; Riedel (1988); O'Keefe (1989) pp. 217-226; Ma et al. (1989) pp. 996-1012
12.6.2.4	Finke and Lloyd (1988) pp. 170-193; Walrath (1989)
12.6.3	Barnett (1990) pp. 1552-1556

REFERENCES (Cont.)

Paragraph	Reference
12.6.4	Ehrhart (1990)
12.6.4.1	LeMay (1988) pp. 227-229
12.6.4.2	Pew (1988) pp. 931-940
12.6.4.3	Schwartz (1983) pp. 13-17
12.6.5	Schwartz (1983) pp. 13-17
12.6.5.1	Gordon (1988) pp. 55-59; Schwartz (1983) pp. 13-17
12.6.5.2	Tannenbaum (1990) pp. 54-59

13.0 QUERY LANGUAGE

This section initially reviews database terminology, types of database queries, and the methods used to store data in databases. Then, the section provides general guidance on user-oriented database design. The remainder provides specific guidance on query screen designs, user requirements, user-friendliness, database searching, and design requirements for novice and expert user interfaces.

A database consists of interrelated data that are searchable by a computer. The computer software that facilitates processing information into organized or summarized groups is called a database management system (DBMS). Retrieving information from a database involves identifying a set of items that match or are similar to the user's query or statement of information need. The term "data access" refers to the process of locating and retrieving requested sets of data. By contrast, the term "data presentation" refers to the process of displaying that data to the user in an appropriate fashion. Data access is the query, and data presentation is the result.

The software that makes up the user interface usually consists of applications programs, report program generators, and query languages. The applications programs allow the end users to enter, retrieve, and update the data in the database. Report generator utility programs help users specify the content and format of reports. Query languages are used to meet requests for information or to provide a means to browse through the database.

Databases must usually be searched in a series of steps. The computer-readable message containing the search terms and logical operators for combining them must be derived from the search query or queries submitted by the user. The search terms are then matched against terms in the database file, either indirectly by searching the index or by searching records directly. The computer responds with counts of retrieved items and should allow the user to sample the items by displaying them on the screen. The user can then make iterative adjustments, either to broaden or narrow the scope of the query.

TYPES OF QUERIES

Users most often communicate with databases by means of command-driven (i.e., query languages), form- and menu-driven, natural-language, and icon-based interfaces.

Command languages provide flexibility and relieve the experienced user of the requirement to traverse an entire menu structure to select a command. Users of this type of interface must be familiar with the command language, the steps required for solving problems, and the computer's syntax for accomplishing each step of the process. Standard Query Language (SQL) and Query-by-Example (QBE) are commonly used languages that perform similar functions.

SQL is a textual language that is becoming a relational database standard. SQL includes table definition, database update, view definition, and privilege-granting, in addition to query facilities. SQL is often embedded in programs written in other languages, where it generates query results that can be processed by programs written in the host language.

QBE is a table-oriented version of the SQL relational database language and is often supported where SQL is used. QBE provides a pictorial representation of database tables. Symbols placed in the proper table columns specify query selection conditions, grouping, data display, and database updates. Although QBE's tabular format offers advantages to users, it requires user sophistication for effective use.

Another screen-oriented method provides a relatively easy technique for developing queries. Query by Forms (QBF) presents the user with data-entry forms that also can be used as templates when developing queries. When accessing data, the user can select one or more of the data-entry fields and enter values, ranges of values, or logical conditions, which are then automatically translated to database queries. Using familiar forms for data entry and search tasks facilitates the user's performance in creating straightforward queries.

Menus provide user-friendly interfaces to command languages, such as SQL. They are designed in a hierarchial or tree structure, which allows the user to proceed step by step through the menu structure to the desired level of detail. Some menu systems allow the user to go directly to a specified level by keyboard command or selecting items from a multi-level menu map. Menu-based query aids offer several advantages. They lead the user through the problem-solving process by indicating which options are available at each point. They are comparatively easy for a novice to use, particularly when the user is not familiar with the query command structure (low syntactic knowledge) or is uncertain how to proceed in solving a particular problem (low semantic knowledge). Menu systems also have disadvantages. Users are forced to make selections from the choices offered by the system and are, therefore, subject to any constraints that might be present. If a user makes an incorrect choice at any level, it can be time-consuming and frustrating to retrace the steps in the menu structure.

Natural language interfaces allow users to formulate queries in their native language (e.g., English, Spanish, etc.). These interfaces use a knowledge of syntax (grammar) and semantics (meaning) to interpret queries and translate them into the query language used by the database system. This approach frees the user from learning the usual conventions and rules of query language. Although natural language interfaces offer great potential, they may require considerable user effort in setting up the underlying dictionaries.

Users may directly query icons, maps, schematics and other visual depictions of physical objects by using a pointing device to select the picture or its features in some sequence. Pointing devices (e.g., a mouse, touch-sensitive screen, or trackball) are often used in combination with menus and text-entry screens to formulate queries. Direct interaction with visual representations of physical objects and icons can facilitate human performance.

DATABASE DESIGN

Ease of use and overall performance of a database system depend on its file structure (the manner in which the records are organized in the file or database) and search processes. The details are chosen by the designer or programmer of the system, often with more concern for the programming aspects of a particular model than for the human performance constraints imposed by that model. The optimum form of information representation will be a function of the task being performed. Unfortunately, current research offers little guidance on how to proceed in database retrieval situations.

Database designs typically use hierarchial, network, relational, or object-oriented models. The hierarchial model represents data in tree structures, and networks represent data as interconnected structures of records linked in one-to-one or one-to-many relationships. Relational databases organize data in tables. Because of its power and ease of use, the relational representation is the prevalent model today and is likely to be the database model of choice in the near future. Object-oriented (sometimes called extended relational) database systems are considered to be part of the next generation of database systems. An object-oriented system represents real-world entities as "objects" that have attributes and defined relationships with other objects.

It is important to recognize that each of these database models can influence the format in which information is presented and the way in which the user can add to, retrieve, or change the information contained in the database. In the end, database models determine the modes of user-database interaction, the format in which the data are presented to the user, and the ease with which a user can acquire information from the database.

This Page Intentionally Left Blank

13.1 GENERAL RECOMMENDATIONS

13.1.1 Ease of Use

A query language or procedure must be easy to learn and use. In fact, ease of use and user-friendliness often determine whether the database is used. A program will not be used if it is intimidating, is too difficult, or requires too much effort.

13.1.2 Interactive Queries

On-line query is preferable to batch or off-line modes because it provides the user the opportunity to interact with the system.

13.1.3 User Assistance

An application should assist the user in creating complex queries and in narrowing down the search in a step-by-step fashion.

13.1.4 Error Detection

The user should be alerted to syntax errors in queries and, if possible, to semantic faults (semantic integrity).

13.1.5 Minimum Training

An effective user interface should not require extensive training to be used easily.

13.1.6 User-Oriented Designs

The system interface should be designed in cooperation with the end users to ensure their satisfaction with the final product. User involvement is most effective when users participate in both development and implementation of the system interface.

13.1.7 Multiple Search Options

Consider the nature of the searches to be performed before choosing an interface format. When more than one type of query is possible, one solution is to choose the interface format that provides the best average performance. Alternatively, provide multiple query and display formats, so the user can change formats as desired or when the nature of the search task changes.

13.1.8 Appropriate Displays

The forms of information display that facilitate quick responses are not necessarily the same forms that produce accurate responses. The three basic forms of information display are spatial, verbal, and tabular formats. Pictures (spatial) are superior to words (verbal) in recall and recognition tasks and often lead to quicker completion times on procedural tasks. However, words lead to greater accuracy in performance.

13.1.9 Individual Preferences

Individual preference plays an important role in the effectiveness of any query application. Users perform better and provide a higher proportion of correct answers when the format of the database matches the format they prefer. Observation shows that, although experience with an application can lead to changes in preference, only pre-existing preferences for display formats influenced user performance.

13.1.10 Displaying Results

Data can be displayed numerically or graphically. Graphical displays include the bar graph, plot, pie chart and other computer-drawn pictures. Because graphical presentations provide less accuracy than numerical presentations, the most important consideration is the transfer of meaning to the user. See Figure 13.1.

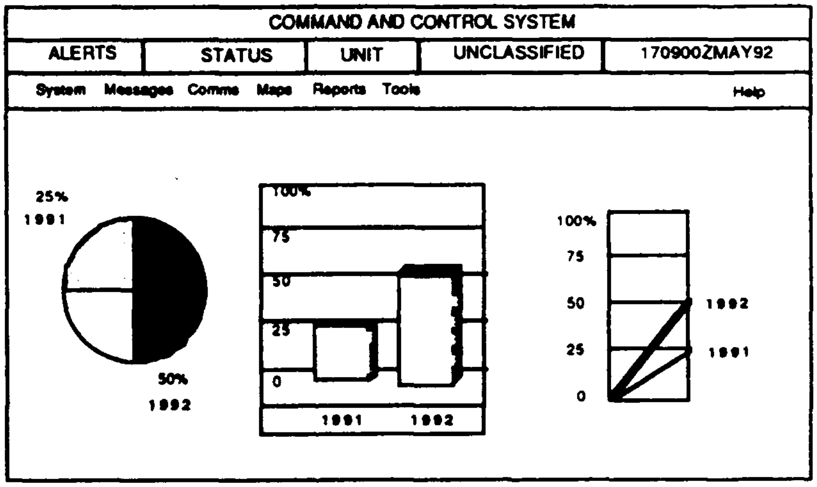


FIGURE 13.1: Examples of Pie Chart, Bar Graph, and Line Graph

13.2 QUERY SCREEN DESIGN

Query screens display the results of a query request or the contents of computer files. The objective in query screen design is to aid the user in quickly and easily locating data or information. Query screen development should optimize human scanning, as scanning is easier when eye movements are minimized, required eye movement direction is obvious, and a consistent pattern is followed.

13.2.1 Screen Design Principles

13.2.1.1 Relevant Information Only

Include on a query screen only information that is relevant to that screen. Forcing a user to wade through volumes of data is time-consuming, costly, and error-prone. If information will never or very seldom be used, do not display it. An item may be relevant one time a screen is displayed but may be irrelevant another. Limit a transaction or screen to whatever is necessary to perform actions, make decisions, or answer questions.

13.2.1.2 Group Information Logically

The interface display should group information in a logical or orderly manner. Locate the most frequently requested information in the upper left corner.

13.2.1.3 Prioritize Information

Locate the most frequently requested information on the initial screens for multiscreen transactions.

13.2.1.4 Balance Screen Design Perceptually

Ensure the screen is not overloaded, and use spaces and lines to balance the screen perceptually.

13.2.1.5 Consistency

Use consistent terminology, commands, formats, and general appearance throughout the interface. Ensure learning can be transferred between modules of the program.

Current technology presents query output mainly in tabular format. Emerging object-oriented technology will provide different ways to present such information as visualization.

13.2.2 Query Screen Organization

Organize the query screen in a logical, orderly, and meaningful manner. When information is structured consistently with a person's organizational view of a topic, that person comprehends more information. Finding information on a query screen can be accelerated by many factors, including the following.

13.2.2.1 Upper Left-Hand Corner

The interface should locate the most frequently sought information on a screen in the upper left-hand corner. If there are multiple screen transactions, locate the most frequently sought information on the earliest transaction screens.

13.2.2.2 Columnization

To aid the user in locating a particular item, provide easily scanned and identifiable data fields. Accomplish this through *columnization* with a top-to-bottom, left-to-right orientation, which permits the eye to move easily left to right across the top of the columns to the proper column before beginning the vertical scan.

13.2.2.3 Top-to-Bottom Scanning

Top-to-bottom scanning will minimize eye movements through the screen and enable human perceptual powers to be used to the fullest.

13.2.3 Captions (Labels)

13.2.3.1 Structure and Size

Captions should be complete and written in a clearly understandable language. Display captions in upper case, although lower case may be used for long, descriptive captions. Do not use reverse video or highlighting for labels.

13.2.3.2 Formatting

For single fields, locate the caption to the left of the entry fields. Separate the caption from the entry field using a unique symbol and one blank space (a colon ":" is recommended). With multiple occurrence fields, locate the caption one line above and centered over the column of data fields.

13.2.4 Data Fields

13.2.4.1 Emphasize Data Fields Visually

The application interface should provide visual emphasis to the data fields.

13.2.4.2 Complete Data

The interface should display directly usable information, as well as fully spell out codes and compressed information. The data displayed should include natural splits or predefined breaks.

13.2.4.3 Data Display

The interface should display data strings of five or more characters (numbers or alphanumeric), with no natural breaks, in groups of three or four characters with a blank between each group. Data strings should be left justified, and numeric data should be right justified. For all types of data, identical data should be consistent despite their origin. See Figure 13.2.

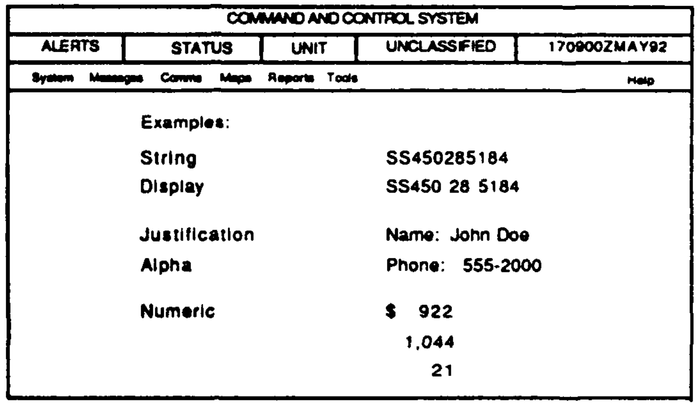


FIGURE 13.2: Data Layout and Justification

13.2.5 Data Organization

13.2.5.1 Order Data Visually

Organize data in accepted and recognizable order, with vertically aligned captions and data fields in columns.

13.2.5.2 Field Justification

The application should be consistent in justifying data displays. (See Figure10.6.)

13.2.5.3 Readability

To promote readability, the interface design should leave at least one space between the longest caption and the data field column, with at least one space between each heading. Section headings should be on-line above related screen fields with captions indented a minimum of five spaces from the beginning of the heading and fully spelled out in upper case.

13.2.5.4 Multiscreen Transactions

When presenting multiscreen transactions, place a screen identifier or page number in the upper right-hand corner of the display.

13.2.5.5 Error and Status Messages

Locate error and status messages consistently in a separate area of the screen. Emphasize these messages by using a contrasting display feature (e.g., reverse video, highlighting, or preceding series of unique symbols, such as asterisks).

13.2.5.6 Multiple Formats

Different search tasks may require different forms of information display. Therefore, the interface should provide, as necessary, more than one display format for user selection and a review format where certain fields of the retrieval records can be reviewed without retrieving the entire record.

13.3 USER REQUIREMENTS

13.3.1 Search Enhancements

13.3.1.1 Query Optimizers

Query optimizers are software procedures that automatically enhance the ability of the database application to execute queries. Use query optimizers to increase the effectiveness of the program, but they should be invisible to the user.

13.3.1.2 Rank Order Search

Permit the user to rank search terms according to importance. Then, use this ranking in a formula for automatically ranking records in the retrieval set according to relevance.

13.3.1.3 Additional Search Terms

A retrieval set should provide additional search terms. For example, use a memo field to list the additional search terms related to a particular field.

13.3.1.4 Avoid Repeat Processing

The application should allow redisplay of results of the previous search without requiring reprocessing.

13.3.2 Automatic Functions

13.3.2.1 Spelling Variants

Provide automatic recognition of spelling variants (e.g., color versus colour).

13.3.2.2 Acronyms

Provide automatic recognition of acronyms.

13.3.2.3 Romanization

Provide automatic recognition of variations in romanization (e.g., Peking versus Beijing).

13.3.2.4 Inverted Form

Provide automatic inclusion of the inverted form (e.g., Newborn Infant to Infant, Newborn).

13.3.2.5 Punctuation

The application should automatically remove punctuation from search terms when matching them against search key values.

13.3.3 Word Stemming

13.3.3.1 Rules for Root Forms

The application should use a set of rules for reducing words to their root forms by stripping them of their suffixes (e.g., reduce, reduction, reducing).

13.3.3.2 Search Root Form

The application should automatically search the index for all words containing a given root (e.g., the word "form" is the root of formation, formable, inform, and information).

13.3.3.3 Rules for Exceptions

Provide rules for exceptions based on the language of the discipline or specialty area.

13.3.3.4 Allow Truncation

Allow truncation. The application should automatically search for all words or phrases that begin with the same character stem (e.g., term for terms, termination, and terminated).

13.3.4 Erasing

13.3.4.1 Immediate Deletion

Allow immediate deletion of individual characters or deletion of the entire line of input (provided it has not been processed by the computer).

13.3.4.2 Deliberate Interrupts

Permit deliberate interruption of computer messages or displays without disconnection (break or interrupt key).

13.3.5 User Satisfaction

User satisfaction with the system can be enhanced by including those factors described in the following paragraphs.

13.3.5.1 Timeliness

Provide results in a timely manner. One factor of timeliness is the elapsed time from when the command is sent until a response is displayed (response time). Another is the time required for characters or graphics to appear on the screen or hard-copy device (display rate).

13.3.5.2 Appearance

The appearance, print format, and organization of output should be natural to the user. User-generated report formats aid in matching the appearance of the output to what the user expects.

13.3.5.3 Level of User Effort

Minimize the level of effort required by the user, including the limitations or qualifications that the application places on search output.

13.3.5.4 Search Capability

Provide maximum capability to the search system while maintaining maximum retrieval effectiveness.

13.3.5.5 Usefulness

The application should assist the user in formulating searches to ensure maximum usefulness of the search results.

13.4 USER-FRIENDLINESS

13.4.1 Commands

13.4.1.1 Mnemonic

Mnemonics help avoid the need for remembering syntax (i.e., as sequences or specifications in output instructions).

13.4.1.2 Brief

Commands should be in an easy-to-learn, user-oriented system language.

13.4.1.3 Unambiguous

Commands should be unambiguous. The meaning should be clear to the user.

13.4.1.4 Keystrokes

Entering data should not be physically awkward for the user, and keystrokes should be limited to those absolutely necessary. Provide the capability to define Ctrl key, Alt key, or function key combinations (i.e., Ctrl/Alt/Del to reboot the system) in place of keystroke combinations.

13.4.1.5 Error Correction

A command that would result in deleting stored information should have a complementary command that reverses the action. If deletion is irreversible, provide the user with the opportunity to reconsider the action. The application should check for meaningless commands against a list of authorized commands, after which the application should allow the user to enter a revised command rather than automatically abort the procedure.

13.4.1.6 Dialog Management

Provide the user with abort or escape facilities for controlling the dialog flow.

13.4.2 Computer Messages

Messages should be clear, simple, and concise. Present the user with the briefest message that can be properly interpreted. Directive messages should be specific and in the context of the current working environment. Messages should warn the user of irreversible action.

13.4.3 Error Messages

Deal with mistakes in a positive, helpful manner. Users will thus gain confidence in the system and feel less intimidated or fearful of damaging it or the data. The error message should appear when the user enters a command that is misspelled, improperly formatted, or cannot be processed because it is inappropriate to the situation. The message should provide instructions for revising the erroneous command.

13.4.4 Documentation

Full system documentation should be available in manual form.

13.4.5 Tailor the Interface

Tailor the interface to suit the needs of users.

13.4.5.1 Tailor Frequent Queries

Tailor frequently used queries. In cases where the value of only one or two parameters changes, provide the user with default values for those parameters.

For example, a query might request the names of all Army officers with over ten years service who graduated from an academy in the top 10 percent of their class and who serve in the infantry.

This query contains elements that could be requested several different times, using slightly different conditions each time.

13.4.5.2 Macro Definition

Macro definition procedures (user-defined commands) are an important feature for expert users who prefer to define their own commands and personalize their environment by encapsulating frequently used query sequences in a new command. Macros greatly simplify user interactions with the application as well as save time. The application should allow the user to store these macros as files or define function key combinations to perform the function.

13.4.6 Accelerators

The interface should provide accelerators to save keystrokes. For example, special keys can be dedicated to commonly used functions. The application should permit direct commands as alternatives to menu options.

13.4.7 Backup

The application should shield the user from system failure. Provide backup facilities both internally by the software application program and externally by the operating system.

13.4.8 Restore

The application should provide a restore utility to facilitate recovery of damaged or destroyed data from backup copies.

13.4.9 Interrupt

The system should provide the capability to interrupt work with the application software, then come back later to resume work at the same point

13.5 SEARCHING

13.5.1 Commands

The following types of database utility and search commands should be available to the user.

13.5.1.1 Database Selection

The application should provide a database SELECT command

13.5.1.2 CREATE/ERASE

The application should have commands to create and erase sets.

13.5.1.3 Combine Sets

Applications should allow users to combine two or more sets to create new sets.

13.5.1.4 Reports/Output

The application should provide the capability for users to specify report formats. The user should be able to name the report, identify the relations from which the report data will be derived, determine the report layout, and define the lines and headings or captions of the report. The user should be able to save the created formulating query and report format for later use.

13.5.1.5 Restrict Retrieval

The user must be able to restrict the output of retrieval sets.

13.5.1.6 Save Retrieval Set

All users should be able to save the results of their searches easily.

13.5.1.7 Search History

The application should provide the user a list of previous search commands upon request. The number of saved commands could be set by the user or could be a prespecified number.

13.5.2 Control Functions

The application should provide control functions to aid the user in dealing with the system. These functions should include signaling about the system's current state or performing an action based on the state.

13.5.2.1 MARK

The input parameter for the MARK command should be the current field value, and the application should note the marked value for future reference. For example, fields or records could be marked for deletion.

13.5.2.2 DESCRIBE

DESCRIBE should use as its parameter the current field value. Provide the user with a detailed explanation or description.

13.5.2.3 DROP

The parameter for the DROP command should be the current field value. The current field should be dropped from the structure.

13.5.2.4 STATUS

The application should provide the user status information upon request. This should include the completion and success or failure of the last search operation executed.

13.5.3 Editing Commands

Editing commands are necessary during query formulation. The application should provide a text-editing box to be used for typing search queries. The following functions should be available. See Figure 13.3.

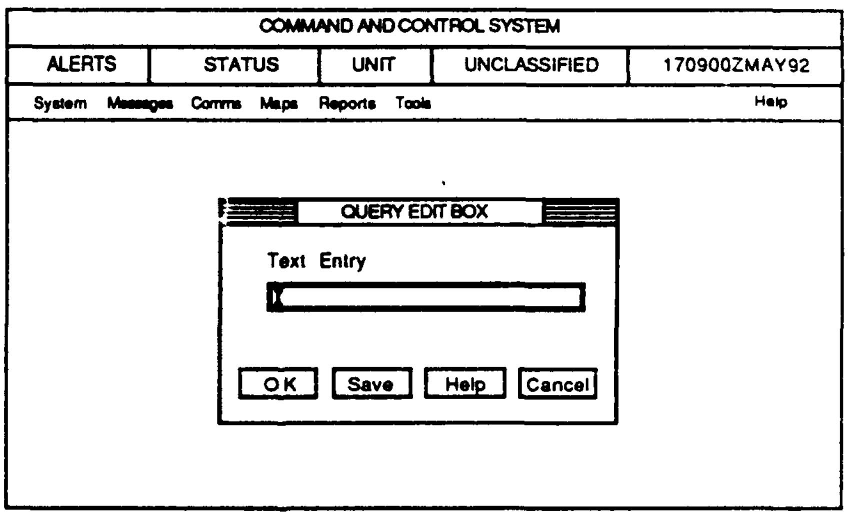


FIGURE 13.3: Sample Text Editing Box

13.5.3.1 CUT

CUT should allow the user to remove the selected text and place it in a clipboard.

13.5.3.2 COPY

COPY should allow the user to duplicate selected portions of text and place them in a clipboard.

13.5.3.3 PASTE

PASTE should allow the user to place text from the clipboard into the current text.

13.5.3.4 CLEAR

CLEAR should remove all characters currently in the text-editing box.

13.5.3.5 SEARCH

SEARCH should allow the user to locate a word or group of characters in the text-editing box.

13.5.3.6 REPLACE

When used in conjunction with SEARCH, REPLACE allows the user to replace a word or set of characters with another word or set of characters.

13.5.3.7 Spell Check

Spell Check should check the words in the text-editing box against a dictionary of recognized words. This function also should check textual commands to assure correct spelling and syntax.

13.5.4 Query Formulation Commands

Major tasks performed by queries include extracting, manipulating, and performing calculations on tabular data, including the creation of tabular results and new tables. Query applications should be able to build functions as needed for developing application programs, as well as update and maintain tables.

13.5.4.1 SELECT

SELECT should identify fields from tables and functions to appear in the query results.

13.5.4.2 COMPILE

COMPILE should generate an executable function and check for correctness.

13.5.4.3 RUN or DO QUERY

RUN or DO QUERY commands cause execution of the query. The application should monitor the execution with prompts for input and error recovery.

13.5.4.4 SHOW

The SHOW command should allow various presentations of a tabular result and could be used to present a preview of the results of a query or report.

13.5.4.5 MODIFY

MODIFY should allow the user to make changes to the query definition of an already existing query or report. The new query could be saved to a file or as a report, if desired.

13.5.4.6 SAVE

SAVE should allow repeated use or modification of a query. Store the queries in a file with a unique extension, such as ".QRY."

13.5.5 User-Friendly Searching

13.5.5.1 Abbreviations

Abbreviations should be significantly shorter than the original word or mnemonic. Truncation is the preferred form of abbreviation. The application should allow both the abbreviation and the full term.

13.5.5.2 Automatic Search Term Completion

The application should automatically complete a search term (opposite of truncation) as soon as it recognizes that the portion of the term entered is unique in the index of search terms.

13.5.5.3 Search Time Feedback

Because even simple queries can overload the computer, the computer should inform the user of the problem and prompt user input to terminate the query or continue.

13.5.6 Features

13.5.6.1 BROWSE

Provide an interactive program that allows the user to navigate through the database. The BROWSE function is especially helpful when queries would be too lengthy to run interactively.

13.5.6.2 Report Formats

Provide facilities to format the results of queries as reports.

13.5.6.3 Search Index

The application should provide the ability to view the list of words and phrases available for searching and term variations, including a link to a database thesaurus to suggest search terms.

13.5.6.4 Parsing

Parsing is the process of deciding how the field will be entered into the search index. Parsing decisions have a direct impact on how a database can be searched and provide flexibility in searching, regardless of how fields were parsed.

13.5.6.5 Proximity Searching

Proximity searching provides the ability to search words in a positional relationship from word index fields, such as titles or abstracts. The words should be either in a specific order or independent of order. For example, the words "query" and "formation" could be searched in the same field.

13.5.6.6 Search Formation Logic

The application should provide the use of Boolean logic, including the use of the logical operators AND, OR, and NOT. It should prompt the user for sets consisting of search terms and combine (intersect) the sets. The search can then be completed as a combined (union) set. The application also should allow interactive editing of queries.

13.5.6.7 Set Building and Step-Wise Refinement

The application should provide set building as a means of performing the search in a series of steps, then view the records that answer a query as a set defined by the query.

13.5.6.8 Range Searching

Include range searching in the application. This type of search should be based on an ordered sequence using FROM and TO.

13.5.6.9 Comparison Searching

The application should allow the user to specify the fields to search, because limiting a search to particular fields may speed the search.

13.5.6.10 Controlled Vocabulary

Using a controlled vocabulary of natural language terms helps novice users formulate queries.

13.5.6.11 Keyword Search Terms

The application should facilitate selecting search terms from key words in records. Then, the interface can display these terms and prompt user selection. For example, the application could rank additional search terms by frequency of appearance in a retrieval set and provide them in ranked order.

13.5.6.12 Search on Field Values

The system should be able to search on specific data field values input by the user. The application should provide a list of possible field values from which users select.

13.5.6.13 Search Sorted Values

The application should be able to order the field values in a reasonable way, such as alphabetically or from greatest number to the smallest.

13.5.6.14 Crossfile Searching

The application should provide a crossfile search, which will obtain the number of references in all potential databases for the search terms or search profile.

13.6 MULTIPLE LEVELS

User-friendly features and requirements differ for the novice and experienced user. Because the novice may soon become an experienced user, the human-computer interface needs to change to suit the evolving needs of the user and the demands of users who have different levels of expertise.

13.6.1 Accommodate Novice and Experienced Users

Multiple levels of interaction are necessary to accommodate the varying levels of experience.

13.6.1.1 Levels of Interaction

Users should be able to change levels at any time during a session.

13.6.1.2 Tutorial

A tutorial mode should be available when possible.

13.6.1.3 HELP

Offer context-sensitive HELP on request at all levels.

13.6.2 Novice Users

The level of computer knowledge required of a novice user should be minimal. The application should be easy to use, provide familiar terminology, and allow the user to begin work with little training. To be used effectively, an application should not depend on a complex command language. However, this ease of use may require a loss of power and flexibility.

13.6.2.1 Reduced Capabilities

A novice interface may contain only a subset of the search capabilities. This system may be a scaled-down version of a more comprehensive program. In addition, novice interfaces may require fewer searchable fields, so the system may not attain the same specificity or variety of search techniques.

13.6.2.2 Computer Prompting

a. The computer software should prompt the novice user to select from a list of options.

b. The interface should provide an explanation of the options presented.

13.6.2.3 Simplify Commands

The novice interface should have a simplified command structure using fewer and more easily understood commands.

13.6.2.4 Mnemonics

Mnemonic selections are preferred over numbered selections.

13.6.2.5 Intelligent Interface

The system design should strive for "intelligent interfaces between naive users and search systems." Two main components of an intelligent front-end are forms and graphics. Menus and data forms can control the flow of the application, and graphics can be used to provide a visual readout of the data.

13.6.3 Experienced User

Experienced users can accommodate comprehensive versions of query applications.

13.6.3.1 Less Detail

The application designed for the expert user can reduce computer overhead by providing less detailed on-line information.

13.6.3.2 Multiple Commands

The application should allow the experienced user to enter multiple commands to speed the dialog.

REFERENCE LIST

Paragraph	References
13.0	Hansen and Hansen (1992); Flynn (1987) p. 221; Harter (1986) pp. 124-127; Frost (1984) p. 8; Humphrey and Melloni (1986) pp. 41-50; Kelley (1984); Katzeff (1986); Martin (1983) pp. 427-483; Hershman Kelly and Miller (1979); Schur (1988); Shneiderman in Vassiliou (1984); Boehm-Davis (1989); Ogden and Brooks (1983)
13.1	Martin (1983) pp. 427-483
13.1.1	Tenopir and Lundeen (1988) pp. 43-45
13.1.2	Frost (1984) pp. 187; Martin (1983) pp. 143, 452
13.1.3	Martin (1983) pp. 452
13.1.4	Grill (1990) pp. 78
13.1.5	Cuff (1980)
13.1.7	Boehm-Davis (1989)
13.1.8	Boehm-Davis (1989)
13.1.9	Boehm-Davis (1989)
13.1.10	Flynn (1987) pp. 401-441
13.2	Galitz (1989) pp. 155-165; Flynn (1987) pp. 401-441
13.2.1.5	Tenopir and Lundeen (1988) p. 45
13.2.5.6	Boehm-Davis (1989)
13.3	Humphrey and Melloni (1986) pp. 200-203
13.3.1.1	Chapnick (1989)

REFERENCE LIST

Paragraph	References
13.3.3.4	Humphrey and Melloni (1986) pp. 143-144, 200; Tenopir and Lundeen (1988) pp. 34; Ehrenreich (1982)
13.3.5	Harter (1986) pp. 154
13.3.5.1	Shneiderman in Vassiliou (1984) p. 5
13.4	Tenopir and Lundeen (1988) pp. 44-45; Humphrey and Melloni (1986) pp. 197-199
13.4.1.4	Flynn (1987) pp. 520-521
13.4.1.5	Flynn (1987) pp. 514-515
13.4.1.6	Benbasat and Wand (1984); Carlson and Metz (1980)
13.4.4	Tenopir and Lundeen (1988) pp. 155-167
13.4.5	Frost (1984) pp. 241-242; Feldman and Rogers (1982)
13.5	Harter (1986) pp. 76-94
13.5.1	Harter (1986) p. 28
13.5.2	Lochovsky and Tsichritzis in Vassiliou (1984) p. 131
13.5.4	Schauer in Blaser and Zoeppritz (1983) p. 33
13.5.5	Humphrey and Melloni (1986) p. 200; Tenopir and Lundeen (1988) p. 34; Ehrenreich (1982)
13.5.5.3	Grill (1990) p. 78
13.5.6	Tenopir and Lundeen (1988) pp. 31-39
13.5.6.1	Frost (1984) pp. 196-197
13.5.6.3	Sormunen in Wormell (1987)

REFERENCE LIST

Paragraph	References
13.5.6.10	Humphrey and Melloni (1986) p. 200; Harter (1986) pp. 41-58; Ogden and Brooks (1983)
13.5.6.11	Humphrey and Melloni (1986) p. 202
13.5.6.14	Sormunen in Wormell (1987)
13.6	Humphrey and Melloni (1986) pp. 199-200; Benbasat and Wand (1984); Tenopir and Lundeen (1988) p. 45; Shneiderman in Vassiliou (1984) pp. 4, 7; Harter (1986) Schur (1988); Kelley (1984); Flynn (1987) pp. 504-506; Cuff (1980); Sormunen in Wormell (1987)

14.0 ON-LINE HELP

On-line help (HELP) provides procedural aids, the ability to recover from errors, and advice without requiring the user to exit the application. Ideally, HELP is always available. A well designed system offers context-sensitive HELP.

Two elements are critical to HELP: the user interface and the content; both are equally important (Kearsley 1988). HELP should be easy to use and provide readily understandable user guidance. HELP must not add problems or make the user's situation more confusing. The HELP interface design will contribute to how often HELP is used because the more difficult the interface is to use or access, the higher the probability HELP will not be used. No help application will be useful if it is difficult to obtain, hard to use, or not easy to return to the application program.

Computer users want to accomplish a particular task quickly and with the least effort possible. When users encounter a problem, they want a solution that involves minimal interruption of the task at hand. If information is not immediately available, users often guess, repeat a previous sequence, or ignore what is not understood. These responses usually lead to further problems.

Along with individual differences, users have varying degrees of computer experience. The following three types of user group may require different types or levels of HELP.

- Novices (users who have little experience with computers) will need help with basic concepts and operations. Novices usually want to see only necessary information.

- Experts (experienced computer users) want to know about limitations, shortcuts, complex operations, and anything else that will allow them to do their work more efficiently.

- Because casual users (who may be either novices or experts) only occasionally use a computer or current application, they need help remembering aspects of the application they previously learned.

General guidelines:

- Make HELP easy for users to access

- Make HELP available throughout the application

- Make access to HELP uniform

- Make HELP easy to understand

- Make it easy to return to the application

14.1 TYPES OF HELP

HELP should reflect the user's requirements without significant impact on application response time. Of the following three types of HELP, the advice and active forms are preferred. Embedded training is often included in HELP. The recommendation is that embedded training not be combined with HELP.

14.1.1 Advice

When users query HELP, they find an interactive, context-sensitive source of information that indicates what entry to make at the current location in the application, the required keystroke, or which steps to be taken to complete the task.

14.1.2 Active

When the HELP application software senses an inappropriate entry, it interrupts to ask users what they are attempting and if they are sure they want to complete the operation they initiated. HELP then suggests the correct form or keystroke.

14.1.3 Passive

Users query HELP when they need assistance. The information may be in the form of on-line system documentation, such as a user's guide or a list of functions performed by combinations of keypresses.

14.2 GENERAL DESIGN

14.2.1 Minimize Keystrokes

Provide single keystroke access to and exit from HELP.

14.2.2 Provide Memory Aids

Assume users cannot remember everything required to run the application; provide memory aids.

14.2.3 Include Basic Information

Include basic information you would expect only novices to seek.

14.2.4 Expand Upon the Manual

Provide clearer explanations of information in the manual, using subsequent screens as needed. Do not simply repeat phrases from the manual the user has read but may not understand.

14.2.5 Choose On-line Portions of the Manual Selectively

Be selective when placing information from the user's manual on line; do not place the entire manual on line. This would be more difficult to navigate through and read than the hard-copy version and would waste system memory.

14.2.6 Include Obvious Information

Include all pertinent information, even that which may appear obvious to the developer.

14.2.7 Avoid Jargon

Avoid using jargon. A friendly and effective interface is the most important component of a HELP system. It is frustrating to a naive computer user to type "HELP," then receive a bit of cryptic jargon in reply. Where use of jargon is unavoidable, ensure it is the jargon of all the users and not of the designer or programmer.

14.2.8 Do Not Overload the User

Do not expect the user to read about more than three HELP displays at a time or to remember more than about five points.

14.2.9 Do Not Use HELP to Teach

Do not use HELP to teach novices how to operate the system. Provide step-by-step instructions to remind occasional users how to perform the most common tasks. Remember that most users perform the same few tasks over and over, in the simplest possible way.

14.3 ACCESSIBILITY OF HELP

14.3.1 Universal Access

Provide access to HELP from every screen.

14.3.2 Availability

Remind users that HELP is easily available by displaying the command or function key used to get HELP.

14.3.3 Display HELP Status

Display a message indicating the status of HELP availability if HELP is not available at all times or places in the program.

14.3.4 Single Action to Invoke

Make it possible for users to get HELP using only a single keypress or mouse-click.

14.4 PROVIDE HELP ON HELP

14.4.1 Alphabetical Index of Functions

Make an alphabetical index of HELP functions available to the user.

14.4.2 Alphabetical Index of Commands

Provide an alphabetical index with explanations of all commands used by the application software, showing the argument options.

14.4.3 Show How to Use

Show users how to use the HELP function. Never assume it is obvious, even to expert users.

14.4.4 Present Alternatives

Show how to get HELP from anywhere in the system. Users may know only one route, so detail the alternatives, including how quick and easy it is to use the options. Define the different meanings of the HELP display and explain their functions.

14.4.5 Navigating Through HELP

Show how to navigate within HELP. Explain how to scroll or page through a topic and how to jump to related topics.

14.4.6 Provide HELP on Screens and Windows

Describe the current window, including its function and tasks the user can perform.

14.4.7 Provide Instructions to Novices

To assist novice and casual users, put instructions for using HELP on every HELP display.

14.4.8 Instruct on When to Use

Provide users with complete instructions on when to use the information supplied by HELP.

14.5 APPLICATION INFORMATION

Provide a list of application capabilities. Show the application components, options, and structure to help the user understand the application and use it more effectively. Experienced users as well as novices underutilize many applications because they do not recognize the full range of capabilities.

14.5.1 Provide Shortcuts

Use HELP to point out shortcuts and unused features.

14.5.2 HELP on Error Messages

Make available successively more detailed explanations of a displayed error message.

14.5.3 HELP on Prompts and Definitions

Make available successively more detailed explanations of a displayed question or prompt and definitions of specified terms.

14.5.4 Show Correct Input

Provide examples of correct input or valid commands.

14.5.5 Show Command Format

Provide a description of the format of a specified command and a list of allowable commands.

14.5.6 Provide User-Centered HELP

Make the HELP user-centered; base it on the user's task, not on application characteristics. Descriptions of application characteristics are more appropriate for a hard-copy user's manual.

14.6 PROVIDE HELP IN CONTEXT

Context-sensitive HELP may be the most important kind of HELP for users. Context-sensitive HELP should describe the nature of a specific control (check button, radio button, slider bar) and how people use that control.

14.6.1 Provide Specific HELP

Ensure the HELP is specific to each level of user interaction (e.g., for context-sensitive HELP in a field specifying printer baud rate). See Figure 14.1.

"Baud rate is the speed in bits per second at which your printer can accept data. Acceptable speeds are 1200, 2400, and 9600. Enter the speed in bits per second."

Figure 14.1: Printer Baud Rate

14.6.2 Show Correct Alternatives

List correct alternatives if the user enters an incorrect command.

14.6.3 Provide HELP Within Application

Provide HELP within the application so users do not have to abandon their place in the application to seek HELP. Users should not have to close files, exit the application, and log off to invoke a HELP utility.

14.6.4 Use Split Screen or Window

Allow users to see the application screen that relates to the HELP request by means of a split screen or window. A separate HELP screen that completely replaces the application screen is undesirable because it prevents the user from simultaneously observing the problem and the HELP screen.

14.6.5 Resize and Reposition HELP Windows

Provide the user the capability to resize and reposition windows to see the HELP information and the problem at the same time.

14.6.6 Identify Special Keys

Where applications have special uses for keys, especially function keys, display the meanings assigned by the application.

14.7 USER CONTROL OF THE HELP SYSTEM

The more control users have over the help system, the more useful they will find it.

14.7.1 User-Initiated

Allow users to initiate a HELP request and select the desired HELP topic.

14.7.2 User-Selected Levels

Allow users to select a level of HELP if multiple levels are available.

14.7.3 Annotate Messages

Allow users to annotate existing HELP messages.

14.7.4 Describe Key Functions

Within HELP, provide the capability of pressing any key to obtain a list of features whose names begin with that letter. When the user then selects a feature from the list by highlighting or clicking, provide an explanation of the feature.

14.8 PROVIDE CONSISTENT HELP FORMAT

14.8.1 Consistent Screens

Provide consistent HELP aids from screen to screen, both with indicators that HELP is available (e.g., "F1 = HELP") and specific location on the screen.

14.8.2 Progressive Detail

When providing progressively more detailed explanations, ensure the process of moving from level to level is consistent from screen to screen.

14.9 SELF-EXPLANATORY AND CONCISE DISPLAYS

14.9.1 Match Titles to Contents

The content of a HELP window should be reflected in its title (e.g., the title of a HELP window for the entry field "Trans" could be "Help for Trans").

14.9.2 Match Names

Ensure the name on the HELP display matches the panel from which help was requested (e.g., when working on an accident report, the help display may read, "HELP: ACCIDENT REPORT").

14.9.3 Ensure Relevancy to User

Tailor the display to the current information requirements of the user, so only relevant data are displayed.

14.9.4 Provide Clear Messages

Make error and HELP messages clear, concise, and appropriate to the experience and training users have had in using the system.

14.9.5 Use Task-Oriented Wording

Adopt task-oriented wording for labels, prompts, and user guidance messages, incorporating whatever special terms and technical jargon may be normally employed in the user's tasks.

14.9.6 Increase Understandability

To increase understandability of HELP, apply the following principles.

14.9.6.1 Short Sentences

Use short sentences when writing HELP messages.

14.9.6.2 Active Voice

Use the active voice in all HELP messages.

14.9.6.3 Examples

Provide as many examples as possible for each HELP screen.

14.9.6.4 Tables

Place HELP information in tables, where applicable.

14.9.6.5 Answer First

Put the answer before the explanation when presenting HELP information in the dialog.

14.9.6.6 Answer Likely Questions

Answer the most likely HELP questions immediately.

14.9.6.7 Limit Scroll Requirements

Minimize the user requirement to scroll or page through displays.

14.10 MAKE RETURN TO APPLICATION EASY

14.10.1 Single Keystroke

The user should be able to return to the application with only a single keypress or mouse-click.

14.10.2 Exit HELP Easily

When a single keystroke exit is not possible, the user should be able to return to the application easily, without calling up a menu, then choosing an item from it.

14.11 KEEP HELP CURRENT

14.11.1 Provide Up-to-Date HELP

Plan and build HELP concurrently with developing applications so HELP information reflects the current version of the software. Provide updates to HELP with subsequent software releases.

14.11.2 Tailor HELP to the User

Collect data on user target population to tailor HELP to the training and experience of the users.

14.12 PROVIDE USER OPTIONS

14.12.1 Bookmarking

Bookmarking is a means of tagging items of interest for easy referral at a later time.

Provide "bookmarking" so users can tag specific HELP messages for easy referral later. This can be especially useful in a large help system consisting of many topics and screens. Bookmarking allows users to customize the help system to their own needs and filter out information of no interest; it can speed up the HELP process and return the user to work with fewer interruptions.

14.12.1.1 Bookmark Selection

To tag a HELP message, allow the user to select a bookmark option while viewing the message.

14.12.1.2 User Options

Ensure the user has an option to see all or just the bookmarked messages.

14.12.2 Print for Later Reference

Ensure a print option is available while HELP messages are being displayed. Users often want to print out HELP information to study it further.

14.13 SYSTEM-INITIATED MESSAGES

Provide system-initiated messages when an error has been detected or when there is other evidence the user is having a problem (e.g., missing parameters, duplicating erroneous commands, long lapses in response, out-of-range responses, etc.).

14.13.1 Positive Tone

Messages should have a positive tone, indicating what must be corrected. Focus on correcting the problem, not the action that caused it.

14.13.2 User Control

Present system-initiated messages to users as advice or suggestions. If users perceive system-initiated HELP messages as an interruption, they may prefer to initiate HELP requests.

14.13.3 Error Control

Provide system-initiated HELP messages for systems where incorrect user actions could result in serious consequences. This is especially true for destructive actions, such as deletions (e.g., MS-DOS command: "Del *.*"), file replacements, exiting the application without saving data, or renaming a file.

14.13.4 Document Errors

Error messages should always state the error detected, the input field containing the error, and the corrective action.

14.13.5 User Understanding of Message

Messages should clearly indicate whether they are meant to inform of error, indicate status, prompt for action, or provide feedback.

14.13.6 User Options

Give users the option to turn off system-generated messages or to specify the level or type of message to be given (e.g., advisory, caution, warning).

14.13.7 Avoid Jargon

Error messages should be specific and address the problem in user terms, avoiding vague terms such as "syntax error" or obscure error code numbers.

REFERENCE LIST

Paragraph	References
14.1	Kearsley (1988) p. 9
14.4.4	Horton (1990) p. 264
14.4.7	Brown (1988) p. 167, para 9.27
14.4.8	Hurd (1983) Cited in Horton 1990, p. 264
14.5	Walker (1987) Cited in Horton 1990
14.5.2	Relles and Price (1981) Cited in Horton 1990
14.5.3	Relles and Price (1981) Cited in Horton 1990
14.5.4	Relles and Price (1981) Cited in Horton 1990
14.5.5	Relles and Price (1981) Cited in Horton 1990
14.5.6	Relles and Price (1981) Cited in Horton 1990
14.6	OSF (1990) p. 8-2, para 8.1.1
14.6.1	OSF (1990) p. 8-2, para 8.1.1
14.6.6	OSF (1990) p. 8-3, para 8.1.3
14.7	Kearsley (1988) p. 79
14.7.1	Kearsley (1988) p. 79
14.7.2	Kearsley (1988) p. 79
14.7.3	Kearsley (1988) p. 79
14.9.1	DoD (1991) p. 7-3, para 7.3
14.9.2	Otte (1982) p. 273, para 3.7

REFERENCE LIST (Cont.)

Paragraph	References
14.9.3	Smith and Mosier (1986) p. 298, para 4.0.5
14.9.5	Smith and Mosier (1986) p. 302, para 4.0.17
14.9.6	Kearsley (1988) p. 68
14.9.6.7	Fenchel (1981) Cited in Horton (1990)
14.10.1	Kearsley (1988) p. 76
14.10.2	Unix System Laboratories (1991) p. 4-27, para 3
14.12.1	Kearsley (1988) p. 25
14.13.1	Unix System Laboratories (1991) p. 4-27, para 3
14.13.7	Nickerson (1986); Smith and Mosier (1986) p. 302, para 4.0.17
14.13.8	Dumas (1988) Cited in Horton (1990) p. 264; Rockley (1987) Cited in Horton (1990) p. 264
14.14	Kearsley (1988) p. 19
14.14.1	DoD (1991) p. 6-2, para 6.1
14.14.5	Thomas and Carroll (1981)

15.0 EMBEDDED TRAINING

The interface of the optimally designed computer program should be designed and tested such that no user assistance is needed. However, because of differences between humans and computers, the variety of task demands, and the ever-present human tendency to make errors, assistance is needed. People differ in computer experience, patience level, learning style, reasoning ability and style, and numerous other characteristics. At the same time, sophisticated computer systems and software programs are often highly complex but still retain the requirement to be highly usable without requiring extensive training or technical expertise. Assistance programs offer one of the primary methods used by designers to achieve a high degree of usability.

User assistance is commonly offered through on-line help, documentation, and on-line training. The distinction between on-line help and on-line training is often blurred. For the purposes of these *Guidelines*, on-line help refers to assistance for a specific problem, function, command, or term. On-line training programs focus on process; they offer instruction.

On-line training programs may exist completely embedded within the application software, separately as an application, or as a combination of both. The on-line training program also may be executed by some form of supplemental component (e.g., strap-on [video disk player] or plug-in [floppy disk]). Though many guidelines apply to both embedded and supplemental training, the interface guidelines presented in this section pertain specifically to embedded training.

The guidelines also apply to a range of embedded training formats and capabilities including:

- fixed format provides the same information regardless of what the user has done

- context-sensitive format depends on what users are currently trying to do or on the context in which they are working

- prompting intervenes or prompts automatically if a user proceeds incorrectly

- dialog allows users to obtain assistance through natural language interaction

- adaptability keeps track of a user's operation and provides appropriate help or training based on the user's operation, for example, intelligent tutoring systems (Kearsley 1986).

The embedded training guidelines included in this section are derived from the results of empirical research, reported computer training experience, and experts' recommendations. Guidance for embedded training interface design appears under a variety of types of on-line training: computer-assisted instruction (CAI) and intelligent computer-assisted instruction (ICAI), computer-based training (CBT), intelligent tutoring systems (ITS), embedded training (ET), coaching, electronic performance support systems (EPSS), and guided discovery, among others.

On-line training strives to support learning how to use an application. However, it conflicts with the user's primary task, because consulting a training program interrupts work in progress. This conflict may cause new users to skip training altogether or to select immediate task help without furthering their overall understanding of the system (Grice 1989; Hackos 1991; Horton 1990). Two crucial factors in determining whether or not users accept and use embedded training are a well designed, intuitive interface and the opportunity to practice. Each of the embedded training guidelines addressed in this section assumes a basic set of objectives for assisting users: consistency, efficient use of capabilities, minimal memory load on users, minimal learning time, and flexible support of different users.

The goal of embedded training interface design is to ensure users can obtain answers to their questions with maximum efficiency, maximum accuracy, and minimum additional memory requirements. An embedded training program should answer the following types of questions:

- Goal-oriented: What types of things can I do with this program?
- Descriptive: What is this? What does this do?
- Procedural: How do I do this?
- Interpretive: Why did that happen? What does this mean?
- Navigational: Where am I?
- Choice: What can I do now?
- History: What have I done? (Baecker and Small 1990; Gery 1991; Laurel 1990).

The manner in which assistance is provided affects the ability of users to learn and transfer that learning to other situations. Research in instruction and on-line documentation has identified basic concepts and practices that support learning and transfer. Central among these concepts relating directly to embedded training are:

- opportunity to practice
- readability
- user control - perceived and actual

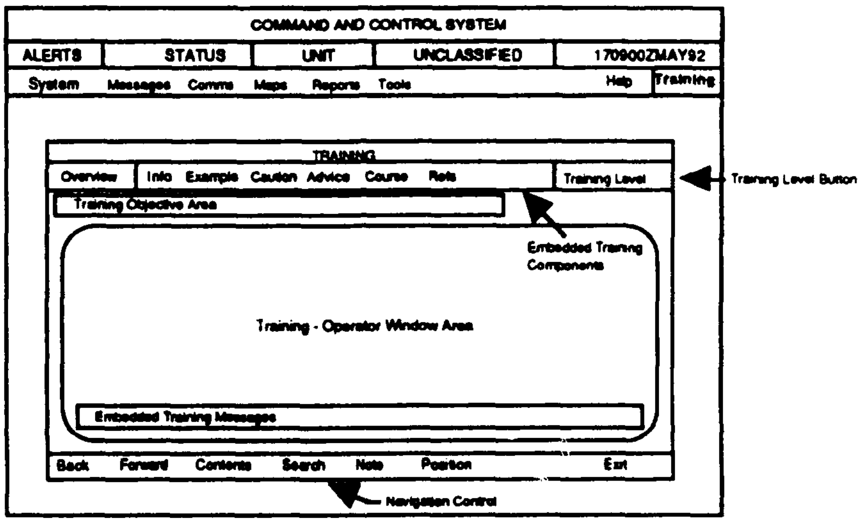
- learning mode - visual (graphics and text)

- advance organizers.

Specific guidelines related to these concepts appear in embedded training components and instructional presentation guidelines (see Paragraphs 15.4 and 15.6).

Much of the relevant research and development work from which these guidelines were developed comes from individual demonstration and limited distribution systems. Significantly less empirical research has explored the behavioral issues pertaining to on-line training or advice-giving systems. On-line training experts call for additional behavioral research, for example, effects of feedback timing, preferences for and effectiveness of different on-line training components, suitability of media presentation (animation, text, sound), and the effects of system-initiated intervention. The current research focus has moved from building systems capable of detecting all possible errors and misconceptions to building an empathetic partner that chooses among several forms of interaction based on the content of the task and needs of the user.

This section of *Guidelines* includes general guidance for embedded training, followed by guidelines pertaining to more specific features. This section also contains a series of figures illustrating embedded training. Each figure is based on a basic screen prototype (see Figure 15.1). To illustrate a particular guideline clearly, a number of the figures show only a portion of the basic screen display.



<u>FIGURE 15.1</u>: Simplified Prototype Embedded Training Screen

15.1 GENERAL

A strong embedded training interface provides users with an understanding of the training program and the linkage between the application program and the training program. The guidelines in this section address user orientation and the linkage between application and embedded training programs.

15.1.1 Initial Use Overview

Provide first-time users of embedded training an overview of the embedded training program. This orientation should convey what the embedded training achieves by combining text and graphics (animated or static).

15.1.2 Positive User Attitude

To build positive user attitudes and increase use of the embedded training, the interface should maintain a positive tone by not evaluating the user's performance when practicing and experimenting, ensuring system messages do not blame the user, and avoiding the implication that the computer is human. See Figure 15.2.

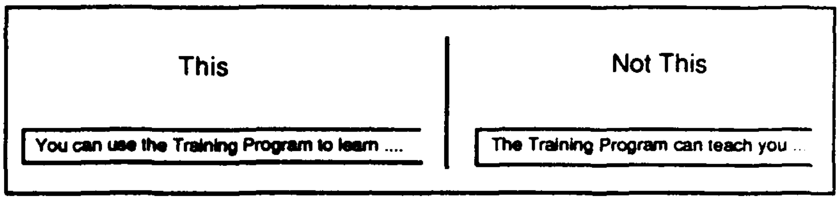


FIGURE 15.2: Examples of Implying Human Characteristics

15.1.3 Personalization

Personalized messages interrupt and often annoy users.

Avoid personalization (i.e., "You did a good job, Sam") and personal recognitions (including even simple statements, such as "Excellent!").

15.1.4 Availability of Embedded Training

The embedded training program should be available to users at all points during use of the application, except where it would interfere with time-critical operations.

15.1.5 Accuracy of Embedded Training

The effectiveness of the embedded training is related directly to the accuracy of the embedded training information.

Ensure the embedded training is accurate, reflects the most current form of the application, and is updated in response to changes in the application. When changes occur in application procedures or critical operations, ensure users are notified changes have occurred. In addition, consider providing users with an option to see new and revised information (i.e., by selecting "News").

15.1.6 Moment of User Need

Whenever possible, provide training support at the moment the user needs it.

15.1.7 Embedded Training Browsing

Allow users to work with the embedded training independent of the application to accommodate user browsing.

15.1.8 Return to the Application From Embedded Training

Ensure users can return to the application from any point within the embedded training with a single action (e.g., keystroke, command, point and click) without shutting down either system.

15.1.9 Restore Application Screen

When users exit the training program, restore the application screen to the state that existed prior to the request.

15.1.10 Restore Embedded Training

When training is interrupted (e.g., system failure, tactical operation requirements) or users exit before completion, offer them the opportunity to return to the position in the embedded training that existed before the interruption.

15.1.11 Protection from Hazardous or Destructive Actions

Prohibit users from accidentally activating hazardous events (e.g., minefield activation) and destructive control actions (e.g., accidental erasure or memory dump) during the embedded training.

15.1.12 Application Screen Protection

Ensure embedded training commands do not alter or destroy application screen data.

15.1.13 Noninterference During Critical Operation

Prohibit system-initiated embedded training interruption of the primary application during a critical operation.

15.1.14 Notification of Critical Operation

Ensure the user is notified of incoming critical application information (e.g., tactical operation input).

15.1.15 Multiple Stations

If the application system has multiple stations, ensure that stations using the embedded training have no effect on the stations performing an operational task.

15.1.16 Context Sensitivity

Make the training context-sensitive; that is, wherever possible, the training should depend on where the user is in the application or on the general nature of the content of the application.

15.1.17 Consistent Application Interface

Provide the greatest possible consistency between the application interface and the embedded training interface to ensure a smooth transition between platforms and to minimize the user's learning requirements (e.g., terminology, displays, commands).

15.1.18 Inconsistent Interface Assistance

Provide assistance if the embedded training interface is substantially different from the application systems operations or when the embedded training interface has complex features that might need to be explained.

15.2 ADAPTATION TO USERS

Users vary in many ways, including computer experience, domain experience (program content - e.g., command and control), learning style, preferred work style, and immediate task demands. Adapting the embedded training interface to the user's characteristics and preferences will encourage use of embedded training and, consequently, should increase user efficiency with the application. See Figure 15.3.

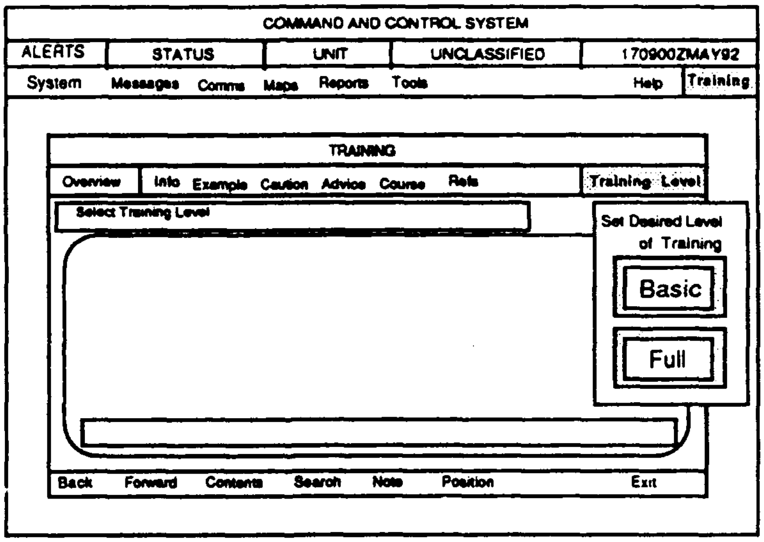


FIGURE 15.3: User Selection of Training Level

15.2.1 User Control Over Level of Difficulty

Accommodate the differences in computer experience by allowing users to select the level and type of assistance. The ability to select is important because users may be novices in some areas, casual operators in others, and experts in still others.

15.2.1.1 Limited Embedded Training Features Options

Reducing the complexity of the training interface for beginners simplifies the demands of learning.

Allow novice users and/or first-time users to select a *restricted capability interface* that blocks features and allows only basic feature operation (e.g., in word processing - creating, editing, and printing).

15.2.1.2 Essential Information for Novices

Provide novices only the necessary information, but allow them access to all of the capabilities by direct request.

15.2.1.3 Expanded Capabilities for Experts

Offer experts assistance in using the system more efficiently (e.g., shortcuts, limitations, complex operations).

15.2.2 Learning Structure

Allow the user to select a type of learning structure. This accommodates the user's individual needs for information and practice. Learning structure types may be:

a. discovery - undirected exploration or browsing

b. guided/supported discovery - directed exploration

c. structured - menu identifies options explicitly and provides implicit cues.

15.3 EMBEDDED TRAINING COMPONENTS

An embedded training system can integrate several resource components to support users while they perform their jobs. Embedded training programs may provide an information database (infobase), common errors, examples and scenarios, interactive advice, internal cross-referencing, expert system-initiated training, and formal courseware.

15.3.1 Multiple Components

In addition to the immediate context-sensitive assistance, offer users multi-component training that is easy to specify and access (e.g., scenarios, examples, information databases, off-line references, common problems, and/or coaching).

15.3.2 Information Database Component

Provide users an interactive information database containing both conceptual and task-oriented information.

15.3.3 Reference Component

Provide a *reference component* that *includes all* on-line resources, as well as system- and job-related, off-line resources.

15.3.4 Examples Component

Offer users the opportunity to practice using common examples in an exploratory or guided mode, which would allow users to work through the steps required to perform a specific task.

15.3.4.1 Experimentation

Encourage user experimentation (i.e., "what would happen if...") by making it easy for them to exit an application, practice, then return to the unaltered application position.

15.3.4.2 Application Protection

If users explore a problem within the application, protect the application system with an UNDO command requirement.

15.3.4.3 Distinction of Application

Clearly distinguish between the exercise and the application to minimize possible confusion arising from switching back and forth between operation modes (e.g., highlight or shadow the practice session).

15.3.4.4 Opportunity to Practice

Avoid demonstrations and exercise summaries if they do not provide opportunities to practice the procedure or function.

15.3.5 Advisor or Coaching Component

Provide an embedded training component that advises or coaches users in solving problems. This may make users aware of enhanced system operation. This also may be used in response to user request, system recognition of suboptimal user performance, or complex tasks. A system can coach users through tasks by presenting a series of questions and recommending a course of action based on the responses. See Figures 15.4 a-c.

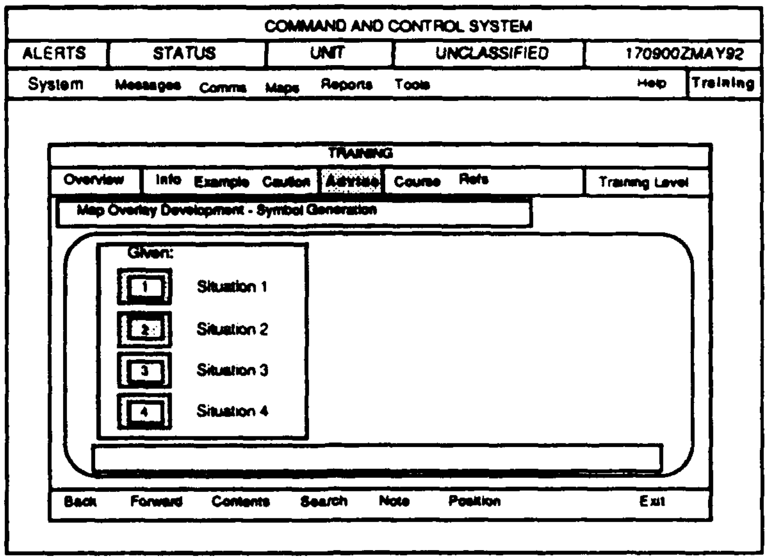


FIGURE 15.4a

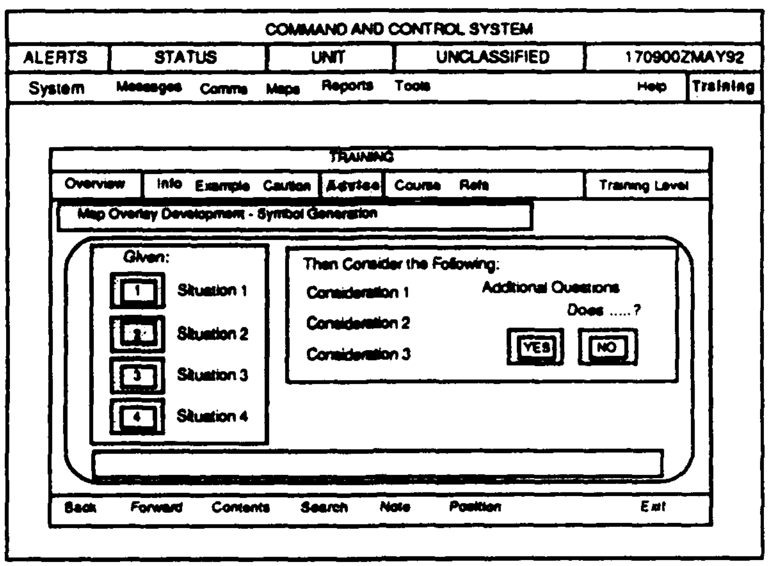


FIGURE 15.4b

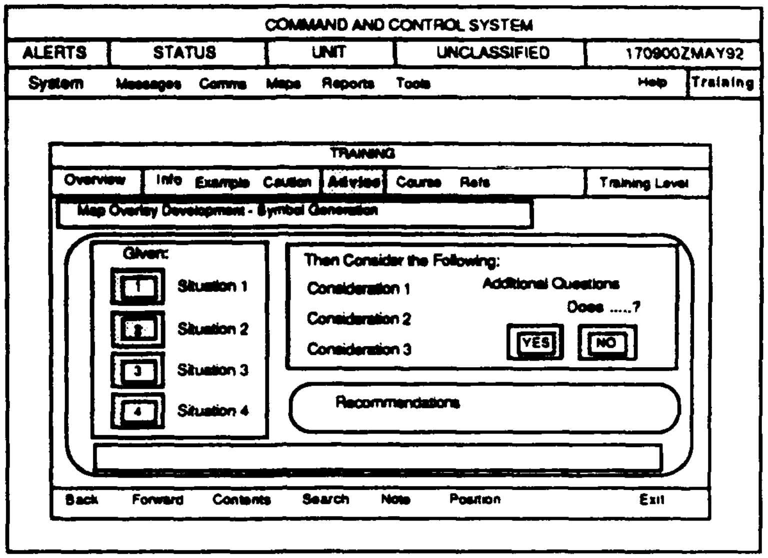


FIGURE 15.4c

FIGURE 15.4: Example of How an Advisor Can Guide Users

15.3.6 Common Errors Component

Provide users with a context-similar, embedded training component that shows common user errors or "Cautions" associated with a given approach or task procedure. See Figure 15.5.

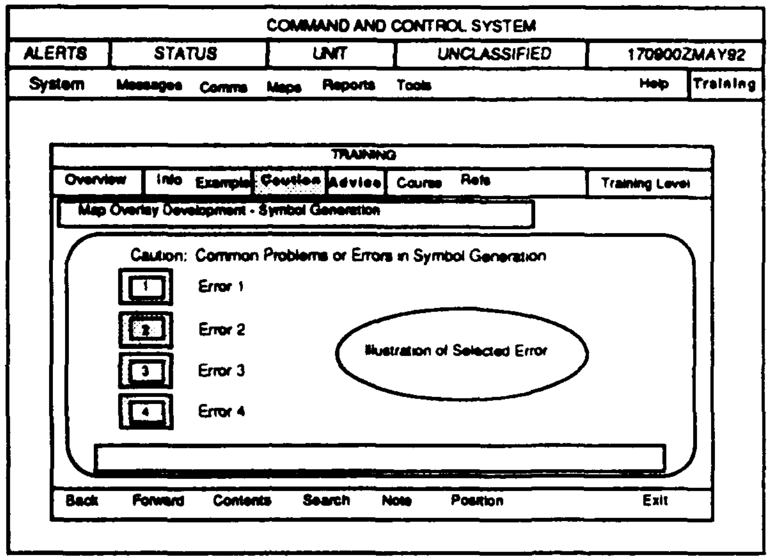


FIGURE 15.5: Example of "Cautions" that Identify Common Errors

15.3.7 Record Keeping

Records of user interaction with the embedded training can be helpful to users, supervisors, and system designers. Using embedded training records requires careful planning to avoid threatening users.

Users are more likely to experiment and practice if they feel their errors will not be seen by others.

15.3.7.1 User Records

Allow the user to record the path through a process and/or the training modules completed successfully or unsuccessfully. This will aid in later reference or experimentation.

15.3.7.2 Protect User Records

If records of user training sessions are stored, ensure the privacy of users is protected by storing their records as anonymous files.

15.3.7.3 Records That Evaluate User Performance

If the exercise or courseware module will be used for evaluation purposes, give users prior notification. Explicitly state the criteria for evaluation.

15.4 INSTRUCTIONAL STRUCTURE

Both the size of instructional unit (granularity) and control of instructional sequence affect the efficiency and attitude of the user.

15.4.1 Granularity

Structure the embedded training components into "single learning episodes," small enough and homogeneous enough to be learned as single units. This enables users to select the particular section or subtopic within a component for which they desire assistance.

15.4.2 System-Controlled Sequences

For novice users and for embedded training that deals with critical or hazardous procedures, the system should direct user movement through the procedures.

15.4.3 Sequence Control for Experienced Users

Provide experienced users with the flexibility to move through the steps of a procedure sequentially or to move directly to any specific step or resource point.

15.5 INSTRUCTIONAL PRESENTATION

Graphic media aid in visualizing significant patterns, whereas natural language text conveys the meaning and significance of the visualization.

The manner in which assistance is provided affects the ability of users to learn from the instructional experience and to transfer that learning to other situations. The following statements outline interface guidelines for instructional presentation.

15.5.1 Combined Media Presentation

Present the embedded training using a combination of media, graphics, and natural language, where appropriate. See Figure 15.6.

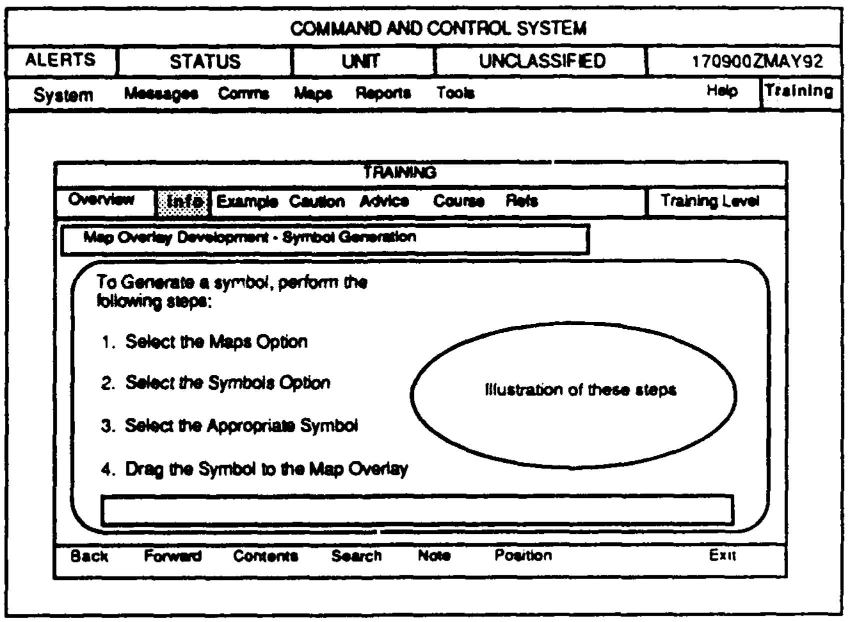


FIGURE 15.6: Combined Graphic and Natural Language Presentation

15.5.2 Graphics for Method-Based Knowledge

Offer users flowchart diagrams that provide an overview conveying method-based knowledge, consisting of a series of procedural steps and decisions.

15.5.3 Reading Requirements

Users prefer to read text in print and often will not read text on a screen that exceeds even a few sentences in length.

Keep reading requirements to a minimum.

15.5.4 Advance Organization

Provide cues and overviews that orient users unfamiliar with the embedded training content and/or process through brief descriptions of scenarios and exercises or courseware outlines if the component will be used for knowledge training.

15.5.4.1 Stated Exercise Objectives

Stated objectives are an important feature for novice users.

Provide users a brief statement of the exercise objective. The statement should refer to the primary purpose of the embedded training request.

15.5.4.2 Module Labeling

Clearly identify each embedded training module, stating objective, content, and, where appropriate, the number of subsections and estimated completion time.

Users prefer reading and studying material in print format rather than in screen displays.

15.5.4.3 Purpose of Assistance Reminder

Remind users of the purpose of their request for assistance. See Figure 15.7.

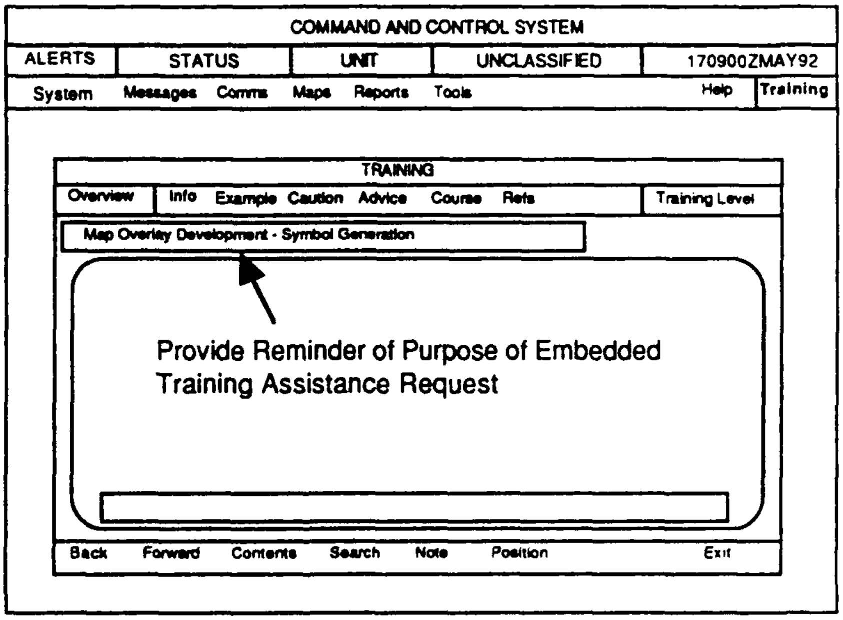


FIGURE 15.7: Example of Assistance Request Reminder

15.5.5 Printing

Provide users the ability to print embedded training content - ranging from screen displays to courseware, information to study further, or for future reference - and/or to print the displayed training material.

15.5.6 Fidelity

Adjust the level of training content and presentation fidelity to match the training:

a. low fidelity for initial training, simple data, and easy process

b. high fidelity for unusual processes, hazardous events, or difficult processes.

15.5.7 Simplicity

Give users simple answers to simple questions. If the answer is long or complex, offer a summary and options to request additional guidance.

15.5.8 Verification

Allow users to verify or confirm selected options, solutions, and commands. This allows them to evaluate the completeness of a process or task and the accuracy of their approach without having to sort through extraneous material.

15.6 ACCESSING TRAINING

15.6.1 Displayed Embedded Training Availability

Display the command, icon, or function key used to access training throughout the application to remind the user of training availability.

15.6.2 Access Via Training Icons

Allow users to access directly the embedded training by selecting an embedded training icon and moving to the point of user need. For example, a user could activate embedded training and select a "Cautions" component icon, move to the point where assistance is needed, click, and receive additional information without exiting the application. See Figure 15.8.

15.6.3 Structured Menu

When using a structured menu to access the embedded training, allow users to add to or change existing embedded training messages (e.g., add terms to the menu using an "ADD" function). If users are allowed to customize menus, the original menu must be protected (e.g., log-on files for individual users).

15.6 Embedded Training - Accessing Training

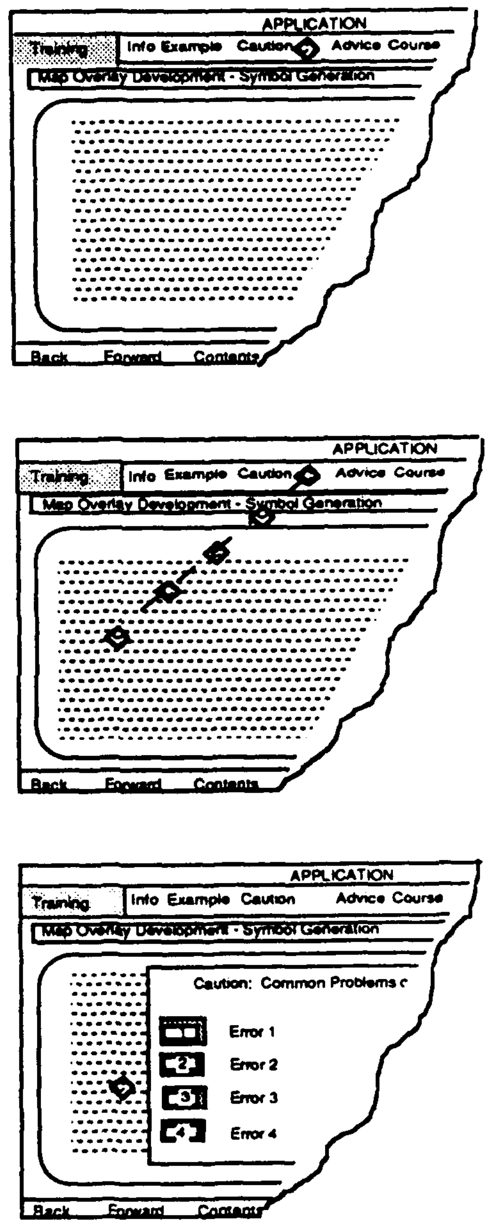


Figure 15.8: Using an Icon to Access Embedded Training Directly

15.7 SCREEN DISPLAY

15.7.1 Complete Display

The content of each screen should stand on its own; users should not have to refer to a previous screen within a module to recall essential information. For example, if users need to enter identical information on a series of screens, the system should automatically enter the appropriate information, repeat the information on each screen, or prompt users to record the information.

15.7.2 Graphics

Select uncomplicated graphics that portray the functional objective clearly without nonessential visual detail.

15.7.3 Window Placement

Display the assistance in windows that do not completely obscure the application's "critical" navigation buttons, operational icons, or the status message line or window.

15.7.4 User Window Control

Allow users to resize and reposition overlapping windows. This will allow users to see the portions of the application with which they are most concerned.

15.8 TECHNICAL COMMUNICATION/WRITING STYLE

Phrase embedded training topics, messages, and menu options in the active voice. Phrase task-related terms to refer to the learning task (e.g., "Creating and modifying fields" instead of "Fields").

15.9 MOBILITY/NAVIGATION

15.9.1 Mobility Within the Embedded Training

Allow users to move among embedded training components freely:

a. without returning to the top of a central hierarchy

b. without exiting the current embedded training component

c. without having to proceed through a preset path

d. without having to step through introductory material.

15.9.2 Embedded Training Navigation Button Display

Display embedded training navigation "buttons" in each embedded training screen for controlling movement between and within modules. See Figure 15.1.

15.10 ERROR FEEDBACK

15.10.1 Immediate Feedback

Provide feedback to users in a timely manner, adjusted to the users' expertise.

15.10.1.1 Multiple-Step Procedures

When practice requires multiple steps, provide users immediate feedback to avoid a series of incorrect actions.

15.10.1.2 Novices

For novices and for uncomplicated problems, offer immediate feedback that includes a suggested next best step.

15.10.2 Context-Similarity Feedback

Provide feedback to users in a form similar to the application, product, or outcome (e.g., an error in equipment setup may be illustrated by correctly configured equipment rather than by a checklist, menu, or even natural language message).

15.10.3 Error Identification

Provide specific feedback that identifies errors rather than assigns a score.

15.10.4 Tone of Error Message

Provide error messages that are constructive and neutral in tone; avoid messages that suggest a judgement of the user's behavior. See the examples in Figure 15.9.

This	Not This
NEUTRAL:	JUDGEMENTAL:
The System cannot....	Invalid Number: Entry must be 4 digits

FIGURE 15.9: Examples of Tone in Error Message

15.10.5 System-Initiated Error Feedback

15.10.5.1 System-Initiated Query Response Time

When a response to a system-initiated query or recommended action is required, the system should provide users the opportunity to stop and think (i.e., consider other options, recall past experiences, weigh problem solutions) without premature system interruption. For example, the system should avoid over-prompting and unwanted problem resolution. The system could offer users the ability to place the system initiation on hold or cancel an upcoming intervention.

15.10.5.2 Preventative Feedback

Provide novice users with prompts identifying probable next-step errors.

15.10.5.3 Control Blocking

Blocking access to system functionality can be very frustrating and can be a result of a misdiagnosed error or correct, but uncommon, approach.

Use control blocking sparingly (e.g., to protect a system from accidental hazardous actuation and system destruction).

15.10.5.4 Control Blocking for Novices

Allow novice users to select an error-blocking option that limits errors.

15.10.5.5 User Feedback Control

Give users the option to block temporarily the system-initiated error feedback or instruction.

15.10.5.6 User Notification of Automatic Correction

Automatic system error correction may cause user confusion.

If the system automatically corrects some errors (e.g., replacing an out-of-bounds parameter), ensure users are notified of the corrections (e.g., by a message and highlighting of the corrected information). See Figure 15.10.

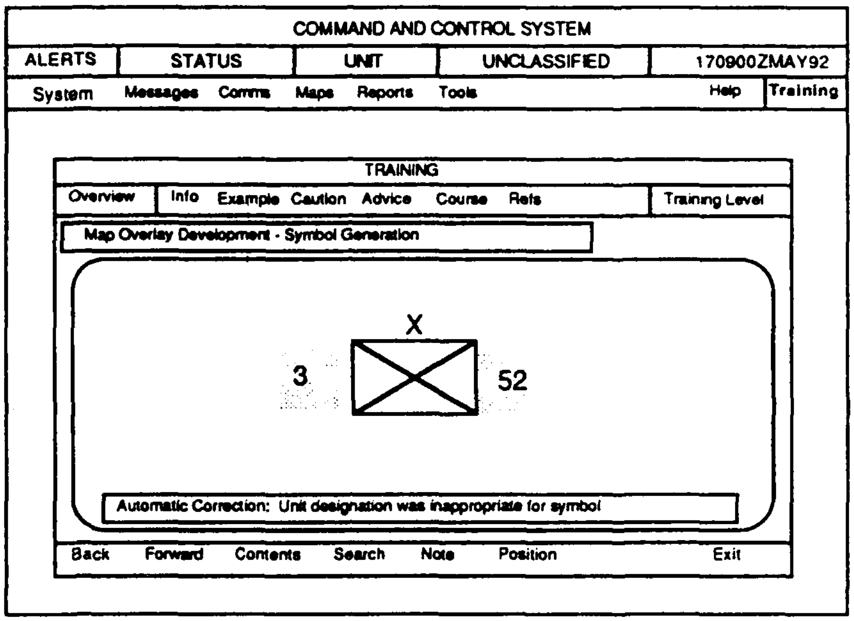


FIGURE 15.10: Example of Automatic Correction Notification and Identification

15.10.5.7 Automatic Correction

Allow experienced users to select automatic system error correction without requiring their confirmation. This option assumes that mundane errors made by experts are a result of minor actions, such as misstriking a key, command, or icon.

15.11 ABILITY TO MODIFY

15.11.1 Additions to the Embedded Training

Provide users the capability to add to, but not modify, the original training system. If the embedded training allows individual user modifications, the original application and embedded training should be protected (e.g., separate log-on files for each user).

15.11.2 Multi-User Systems

On multi-user systems, permit the individual user to store and reference for individual use.

15.11.3 Annotation

Permit users to annotate a copy of the training program (i.e., examples, pitfalls, process notes, references etc.).

15.11.4 Annotation Search

Provide users the ability to search an annotation log.

15.11.5 Icons Used to Designate Annotations

Use icons to designate the position of an annotation in the embedded training program (e.g., a user's example, caution, or additional reference). See Figure 15.11.

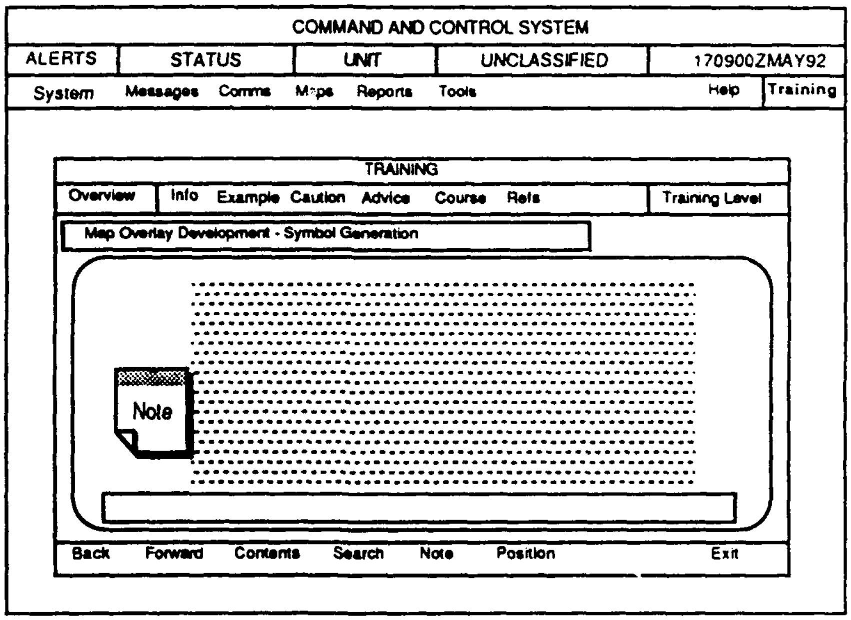


FIGURE 15.11: Example of Identification Annotation Position

This Page Intentionally Left Blank

REFERENCE LIST

Paragraph	Reference
15.1.1	Nicol (1990) p. 115
15.1.2	Kearsley (1988) p. 27; *Shneiderman* (1987) pp. 371-372
15.1.3	Kearsley (1988) p. 27; Carroll and Mazur (1986) p. 38
15.1.4	Shneiderman (1987) p. 374
15.1.5	Kearsley (1988) p. 99
15.1.6	Gery (1991) p. 59
15.1.7	Kearsley (1988) p. 22
15.1.8	Wexelblat (1989) p. 74
15.1.9	Helander (1990) p. 603
15.1.9.1	Helander (1990) p. 603
15.1.10	Roth et al. (1988) p. 915.
15.1.11	Getler (1991) pp.14-22
15.1.12	Roth et al. (1988) p. 98; Carroll and McKendree (1987) p. 24
15.1.13	Avery (1992) Personal Communication
15.1.14	Knerr (1992) Personal Communication
15.1.15	Kearsley (1988) p. 77
15.1.16	Fernandes and Maracle (1991) p. 9; Ripley (1989) p. 811-822
15.1.17	Roth et al. (1988) p.98
15.2.1	Kearsley (1988) p. 66, 81; Wexelblat (1989) p. 76

REFERENCE LIST (Cont.)

Paragraph	Reference
15.4.2	Fernandes and Maracle(1991) p. 9
15.4.3	Fernandes and Maracle (1991) p. 9; Seybold's Office Computing Report (4.1989) p.6
15.5.1	Badler and Webber (1991) p.71; (1990) p. 638
15.5.2	Helander (1990) p. 358
15.5.3	Nicol (1990) p. 115; Roth et al. (1988) p. 118
15.5.3.1	Carol and Mazur (1986) p. 38
15.5.3.2	Roth et al. (1988) p. 118; MIL-STD 1379D (12.5.1990) p. C-5
15.5.3.3	Roth et al. (1988) p. 118
15.5.4	Carol and Mazur (1986) p. 37; Walker (1987) pp. 238-243
15.5.5	Kearsley (1988) p. 24
15.5.6	Roth et al. (1988) p. 30
15.5.7	Wexelblat (1989) p. 74
15.5.8	O'Malley et al. (1983) p.6; Carroll (1982) pp. 49-58
15.6.1	Kearsley (1988) p. 67
15.6.2	Seybold's Office Computing Report (4.1989) p. 5
15.6.3	Kearsley (1988) p. 79
15.7.1	Fernandes and Maracle (1991) p. 9
15.7.2	Brooks et al. (1990) p. 1387
15.7.3	Fernandes and Maracle (1991) p. 915.7.4Kearsley (1988) p. 76

REFERENCE LIST (Cont.)

Paragraph	Reference
15.8	Sellen and Nicol (1990) p. 145; Helander (1990) p. 360
15.9.1	Fernandes and Maracle (1991) p.10
15.9.2	Fernandes and Maracle (1991) p. 9
15.10.1	Roth et al. (1988) p. 36
15.10.1.1	Roth et al. (1988) p. 36
15.10.1.2	Wenger (1987) pp. 296-297
15.10.2	Roth et al. (1988) p.36
15.10.3	Roth et al. (1988) p.36
15.10.4	MIL-STD-1472D (1981) p.274; Shneiderman (1987) p.317; Smith and Mosier (1986) p.391
15.10.5.1	Wexelblat (1989) p.76
15.10.5.2	Wexelblat (1989) p.76
15.10.5.3	Carroll and McKendree (1987) p. 24
15.10.5.4	Carroll and McKendree (1987) p. 24
15.10.5.5	Kearsley (1988) pp. 20, 80
15.10.5.6	Carol and Mazur (1986) p. 40
15.10.5.7	Carroll and McKendree (1987) p. 24
15.11.1	Kearsley (1988) p.23
15.11.2	Kearsley (1988) p.23
15.11.3	Wenger (1987) p. 319
15.11.4	Wenger (1987) p. 319
15.11.5	Gery (1991) p.63

16.0 SPECIAL DISPLAYS

The cathode ray tube (CRT) is the principal display technology used in computer-based systems. Success of the CRT can be attributed to its ability to deliver full color imagery at high luminance and resolution inexpensively. However, tasks of the modern military and emerging alternative display technologies have permitted development of computer-based systems using display technology other than the traditional CRT. Examples include the liquid crystal display (LCD) used in portable computers and projection technology used to brief military personnel in command centers. As a result, the designer has more alternatives when selecting a display for a military system. The purpose of Section 16 is to present guidelines relevant to the specification, selection, use, or design of displays other than the CRT.

Although human-computer interface (HCI) guideline documents frequently assume the display technology used is a CRT, tactical systems increasingly use alternative technologies. Tactical system designers using such display technologies face difficulty in determining how to design the user interface.

Two major problems associated with special displays:

- Relatively little human factors or human performance research has been devoted to special displays, compared to research for CRT-based displays. Research needs are as varied as the special display technologies themselves and include character design issues (Benson and Farrell 1988), effects of display failure (Lloyd et al. 1991; Dye and Snyder 1991), display polarity (Decker et al. 1991), and the development of image quality metrics (Reger et al. 1989; Biberman and Tsou 1991). Rapid advances in display technology, coupled with difficulties in performing behavioral and image quality research with emerging technologies, have compounded this problem. Consequently, the Department of Defense faces a challenge in establishing display requirements and image quality assessment (Biberman and Tsou 1991).

- CRT guidelines are not necessarily directly applicable to other display technology. Many CRT guidelines and standards applied to video display terminals have been derived from research performed with printed text (Benson and Farrell 1988). Though appropriate for CRT-generated displays because images in both medias are analog, these guidelines may not generalize to some of the newer display alternatives where images are formed by patterns of discrete cell-based pixels. As a result, CRT guidelines should be carefully applied to different display technologies. Information display factors (e.g., viewing distance, display orientation, and constraints on character and line spacing) should be considered before applying guidelines across display technologies. The effect of a display technology on legibility, readability, and usefulness of the information to be presented should also be carefully considered and tested prior to implementation.

Section 16 has been divided into subsections, each of which describes current technology, cites advantages and limitations, and presents available guidelines. Many reports upon which this section is based were published in various conference proceedings rather than in refereed, scientific journals and, therefore, may have had less extensive peer review and professional scrutiny. The *Guidelines* user should consult the references for further information on display technologies and human performance considerations. The following subsections deal with five types of special display technology:

- flat-panel displays
- large-screen displays
- stereographic and 3D displays
- glare reduction techniques
- touch interface devices (TIDs).

This Page Intentionally Left Blank.

16.1 FLAT-PANEL TECHNOLOGY

A flat-panel display is flat and light and does not require a lot of power. "Flat" means being thin in form, as well as having a flat display surface. An ideal flat-panel display has the following characteristics: thin form, low volume (cubic size), even surface, high resolution, high contrast, sunlight readable, color, low power, and light weight. Recent advances in flat-panel display technologies have made them realistic alternatives to CRTs for displaying information at computer workstations. Advances have been made in many areas: addressability, contrast, luminance, and color production. Continued research in flat-panel displays has resulted in introducing high information content products that challenge the CRT in specialized applications.

Although a long list of flat-panel display types can be made, liquid crystal displays (LCD), electroluminescent displays, and gas plasma displays are the only flat-panel technologies currently mature enough and economical enough to be used in the military. A major characteristic of each of these display technologies, as distinguished from CRT technology, is that images are formed by turning discrete, non-overlapping, rectangular, cell-based pixels on and off. This discrete, pixel-based structure provides part of the reason that measures of image quality used to evaluate CRT resolution cannot be effectively used to predict image quality and human performance with flat-panel displays.

Factors affecting human performance that differ from the guidance given for CRT's include character-to-character spacing, interline spacing, character and symbol design, the effect of ambient illumination, image polarity, and failure mode. An overriding guideline when specifying flat-panel display technology relative to CRT technology is to apply more stringent image quality criteria when selecting flat-panel technology.

16.1.1 Character Size

Character size is an important variable affecting performance error rates. Height and width of the character and the size of the pixel matrix have important effects on human performance. Exercise special care when determining the character size to use on a flat-panel display.

16.1.1.1 Vertical Orientation

To improve text search and sorting task performance, use a 9 x 13 pixel matrix or larger.

16.1.1.2 Nonvertical Orientation

When displaying dot matrix symbols in nonvertical orientations, use at least an 8 x 11 pixel matrix and preferably a 15 x 21 matrix size.

16.1.1.3 Stroke Width

Character stroke width (SW) should be in the range defined by: (character height ÷ 12) + 0.5 ≤ SW ≤ (character height ÷ 6). See Table 16.1 for guidance.

Table 16.1: Required Pixels for Stroke Width

Pixels in Upper Case Character Height	Minimum Stroke Pixel Count	Maximum Stroke Pixel Count
7 to 8	1	1
9 to 12	1	2
13 to 14	2	2
15 to 20	2	3
21 to 23	2	4

16.1.1.4 Height:Width Relationship

Character height to width should be in the range defined by: (character height x 0.5) ≤ character width ≤ (character height x 0.9). See Table 16.2 for guidance.

Table 16.2: Required Pixels for Width Design

Pixels in Upper Case Character Height	Minimum Width Pixel Count	Suggested Minimum Width Pixel Count	Maximum Width Pixel Count
7	4	5	5
8	4	6	7
9	5	6	8
10	5	7	9
11	6	8	10
12	6	9	11
13	6	9	12
14	7	10	13
15 or 16	8	11	14

16.1.2 Luminance Nonuniformity

Display luminance should be uniform across the surface of the display. Maximum luminance nonuniformity levels should be consistent with the values specified in Table 16.3.

Table 16.3: Maximum Luminance Nonuniformity

Test Object Separation At the design viewing distance	$\frac{L_{higher}}{L_{lower}}$ Maximum
$>7^{\circ}$	1,7
5 to 7 $^{\circ}$	1,6
4 to 5 $^{\circ}$	1,5
2 to 4 $^{\circ}$	1,4
$\leq 2^{\circ}$	1,3

16.1.3 Image Formation Time

Image formation time (IFT) is the time required to render a new image. Four classes of IFTs (see Table 16.4) have been defined, each relating to information update requirements for the application. IFTs for all systems should be consistent with Classes III and IV.

Table 16.4: Image Formation Time Classes

Class	Image formation time in milliseconds	Significance
I	$120 < t$	Satisfactory for displays which update an entire page of information at once. Noticeable during key entry. Applications using scrolling, animation and pointing devices are significantly degraded.
II	$55 < t \leq 120$	Satisfactory for displays which update an entire page of information at once. Not noticeable during key entry. Applications using scrolling, animation and pointing devices are somewhat degraded.
III	$10 < t \leq 55$	Satisfactory for most applications. Motion artifacts can be distracting but are usually acceptable.
IV	$3 < t \leq 10$	Motion artifacts become less noticeable at formation times approaching 3 milliseconds.

16.1.4 Display Failures

The three most common failures on matrix-addressable displays are cell failures involving individual elements, vertical line failures, and horizontal line failures. Displays can fail actively or passively and leave pixels or lines permanently on or off, respectively.

16.1.4.1 Cell versus Line Failures

Because cell failures often lead to greater performance problems, select displays that minimize the likelihood of cell failure.

16.1.4.2 Active versus Passive Failures

To minimize the performance impact of cell failures, select displays and set display polarity so these failures are likely to match the display background.

16.1.4.3 Reading Performance

When display element failure is an expected problem, increase the redundancy in the text to minimize the impact on reading performance associated with display element failures.

16.1.4.4 Symbol Recognition/ Identification

Recognition and identification performance with cartographic display is subject to significant decline with as little as 1% pixel failure. Select and maintain displays to ensure a pixel failure incidence below this level.

16.1.4.5 Character Size

Use characters with a pixel matrix larger than 7 x 9 pixels in order to reduce the negative effect of "on" failures.

16.1.5 Polarity

A display with white (or light) characters on a black (or dark) background is said to have "negative contrast" or to be a "positive (image) display." Conversely, dark characters on a light background is said to have "positive contrast" or to be a "negative (image) display." If character stroke width, modulation, and luminance values are nearly equal for both polarities, select a positive contrast/negative image display for better reading speed, search time, and search error-rate performance.

16.2 LIQUID CRYSTAL DISPLAYS (LCDs)

LCDs are perhaps the most developed and popular of flat-panel display technologies. Rather than emit light, as do active flat-panel technologies, an LCD controls or modifies the passage of externally generated light. An LCD is typically made of transparent plate electrodes that sandwich a liquid crystal substance. Voltages applied to these electrodes cause realignment of the liquid crystal material, changing its optical properties and allowing light to propagate through the material. By selectively applying voltage to the electrodes, individual display elements can be made light or dark to create the desired image on the LCD.

The LCD is available in a large variety of formats for both commercial and military applications. Display size and resolution range from small, character-based displays (e.g., those found in watches) to computer displays with up to 640 x 480 resolution. LCDs are monochrome or color and with backlighting will operate across a wide range of ambient illuminances. LCDs are especially suited for information display in environments where ambient illuminances are high.

Advantages of the LCD include excellent contrast, long life, rugged design, low voltage, and low power consumption (except when backlighted). LCD technology is limited by slow speed, limited color capability, temperature range, and manufacturing problems for larger panels with higher resolution.

16.2.1 Ambient illumination

Provide adequate levels of ambient illumination, because reading performance improves as ambient illumination increases over the range 20-1500 lux.

16.2.1.1 Usage

Consider LCDs for effective display in high ambient illumination situations.

16.2.1.2 Low Ambient Illumination

In low light situations, provide the ability to adjust the viewing angle and the amount of backlight to enhance the legibility of presented information.

16.2.2 Polarity

For legibility of transmissive or backlighted LCDs, use dark characters on a light background (positive contrast/negative image displays). For reflective LCDs, use light characters on a dark background for better performance.

A footLambert (fL) is a unit used to measure light reflected from a surface. An ideal surface that reflects all light striking it and diffuses it with perfect uniformity has a luminance of one fL when illuminated by a one footcandle source.

Candela (cd) per *square meter* (cd/m^2) or nit (normalized intensity) is a metric unit for reflected light. One cd or nit is equal to 0.29 footLambert (fL) or one fL is equal to 3.4 nit.

16.2.3 Level of Backlighting

Minimize or eliminate use of backlighting because display reading errors increase as the level of LCD backlighting increases over the range 0-122 candela $(cd)/m^2$.

16.2.4 Backlighting and Angle of View

Carefully consider the potential impact of user performance decrements before using a backlighted LCD that is to be viewed off-axis. There is an adverse impact of backlighting on user performance when the display is viewed at an angle.

16.3 GAS PLASMA DISPLAYS

Plasma panels or gas discharge displays are widely used flat-panel technology in the information and computer industry because of their inherently high-contrast and high-resolution capabilities. Images are formed by ionizing a gas, usually neon, trapped between a set of horizontal and vertical electrodes. When an electrical field created by the electrodes is increased rapidly, the gas begins to discharge, resulting in a glow that forms an image. The image can be maintained by sustaining the electrical field, or erased by dropping the voltage below some threshold value. The high contrast exhibited by plasma panels is a result of almost no light output in the "off" state (the electrical field is below threshold) and high luminance in the "on" state. Full-color plasma can be made by depositing phosphors on the glass display surface. The plasma gas discharge in this case excites the phosphor, in much the same manner the electron beam does in a CRT, and color images are produced.

Plasma panels can be found as either monochrome or full-color displays in a number of sizes and configurations. A major advantage of plasma technology is that very bright, high-resolution panels are available. Panels that measure 2048 x 2048 pixels at 100 pixels/inch are available, as well as those that can be viewed in direct sunlight.

Panels with luminances of 150-600 cd/m^2 have been produced, with typical large-area, high-resolution display luminances being 30-50 cd/m^2. Full-color DC plasma panels are not yet able to achieve the luminance output nor the display life normally associated with plasma technology.

Features of plasma technology generally include uniformity, high resolution, large size, long life, ruggedness, and the absence of flicker. Applying plasma technology in the computer and information industry is limited by high voltage and power requirements, complexity of the drive circuitry, low luminous efficiency, need to develop more fully a color capability, and lack of developers of the technology. The literature review conducted for *Guidelines* determined that specific interface performance information was unavailable for plasma displays. In the absence of specific guidance, the designer should use the most conservative approach to interface design.

16.3.1 User Concurrence

Verify the use of gas plasma displays by user personnel as a viable alternative.

16.3.2 Testing Prototypes

Designers and developers of command and control systems should consider field testing prototypes before committing to gas plasma technology.

16.4 ELECTROLUMINESCENCE (EL) DISPLAYS

Electroluminescence (EL) displays consist of a layer of polycrystalline phosphor powder or evaporated film phosphors sandwiched between sets of vertical and nearly transparent horizontal electrodes. When an electric field is applied across the polycrystalline phosphor, it is stimulated and light is emitted. The display resolution and the shape of the pixel are defined by the arrangement of the electrodes. EL displays are usually classified as one of four types: either AC or DC thin-film displays, or AC or DC powder displays.

AC thin-film displays and DC powder displays are the most advanced of the EL technologies, and discussions of the strengths and weaknesses of EL technologies will be limited to these types. AC thin-film EL displays have very good luminous efficiency, high contrast, good resolution, and long life. As with some of the plasma technology, these displays require high voltages and complex drive electronics, are expensive, and need research and development to deliver full-color performance. Currently, ELs that display up to 864 lines by 1024 pixels are available for alphanumeric and graphics applications.

16.4 Special Displays - Electroluminescence (EL) Displays

DC powder ELs have a good appearance, simple structure, good luminous efficiency, and the ability to produce grey scales. However, they require high voltages and complex drive circuitry, and have limited luminance output coupled with high reflectance, which may lead to contrast problems. Also, they are expensive. Full-color displays have been produced using DC powder technology, but this too needs more development. Currently, resolutions compatible with graphics interfaces for personal computers are available (480 lines by 640 pixels) for use in applications that require alphanumerics and moderate graphics.

The greatest problems with EL display technology are that developers are few, and investment in research and development is small.

Interface performance information on EL displays is not available in the current literature. In using EL displays, follow the guidelines recommended for LCDs, and use the most conservative level of those recommendations.

16.4.1 User Concurrence

The use of EL displays should be verified by the user personnel as a viable alternative.

16.4.2 Demonstration of Concept

The use of EL displays should be field-prototyped before incorporating into a new system.

16.5 GLARE-REDUCING TECHNIQUES

Glare, as observed on the face of an electronic display, is composed of two components: 1) Diffuse glare or veiling glare, caused by the general illuminance in the environment, can be characterized as a field effect and has little or no modulation. The effect of diffuse glare is to reduce the effective contrast of the display. 2) Modulated or specular glare is the first surface reflection off the faceplate of the display and results from some object or objects in the area surrounding the display. The effect of this type of glare is the appearance of unwanted images on the display surface, making the displayed information more difficult to see and interpret.

The most effective control of glare is to design appropriate workspace illumination so neither diffuse nor specular glare is produced. This is the only method of glare control that will not compromise the resolution and contrast of the display. Because it is not always possible to control the sources of illumination properly, glare reduction techniques have been developed to minimize the unwanted effects of glare.

Many kinds of glare control techniques are used in the electronic display market. Some are screen meshes placed over the display surface, chemical or mechanical etches of the faceplate of the display, anti-reflective coatings, and bonded quarterwave filters. Each has advantages and disadvantages in terms of ability to control diffuse or specular glare, and in terms of effect on display resolution, flexibility, maintenance, and cost. For example, a bonded quarter wave filter only minimally degrades display resolution but is very expensive, whereas a mesh overlay is very inexpensive but has a major effect on the resolution of the display.

The effectiveness of the glare reduction technique is a function of its ability to suppress each component of glare, while minimizing degradation to the display's resolution and contrast. The desired effect is to match as closely as possible the display performance under optimum conditions. While both contrast and resolution are degraded in an absolute sense, the effective image quality in the operational environment and the acceptance of the display system should improve.

Selecting the glare control alternative most effective for a particular display depends on the information to be displayed, task required of the operator, and environment in which the display will be used. In a command and control facility, use careful analysis and testing to determine the type of glare reduction measures that should be taken.

16.5.1 Reflected Glare

When possible, avoid reflected glare by altering the angular relationship among the observer, display, and glare source. For example, provide the ability to adjust height, viewing angle, and/or contrast.

16.5.2 Filter Selection

When possible, leave selection of the specific glare-reduction technique to individual users.

16.5.3 First-Surface Specular Reflections

Because many types of display consist of multiple plates of glass, each of these acts as a specular reflector. All flat-panel displays should incorporate a first-surface treatment to diminish first-surface specular reflections.

16.5.4 Etched Filters

Etches with gloss values of 45 or less should not be used on monochrome CRTs, and etched filters should not be used at all with high-resolution displays.

16.5.5 Projection Displays

With projection displays, minimize glare potential by positioning projection equipment so the light source is not readily visible to viewers.

16.6 LARGE-SCREEN DISPLAYS

The command and control environment not only imposes requirements for information display at individual workstations, but also on work areas where many persons must observe and use the information presented on a display. Large-screen displays first appeared in the command and control environment as the presentation of mission information on transparent plexiglass overlays. Today, large-screen technology found in the command center is computer-driven, with the ability to present graphic and video information.

Large-screen display of surveillance, weather, and intelligence information to operations personnel and as a briefing aid to the principal decision makers is typical in the command and control environment. Unfortunately, because of minimal brightness, poorer contrast, and lower resolution when compared to higher resolution desktop displays, most implementations of large-screen technology have been disappointing to the users. Current display technology offers the military a number of choices when implementing a large-screen display. Figure 16.1 illustrates the types of technology available for fielding large-screen displays. The military currently uses both direct view (e.g., high luminance CRTs or large plasma displays) or projection (e.g., light valves or projection CRTs) large-screen displays in its command and control facilities.

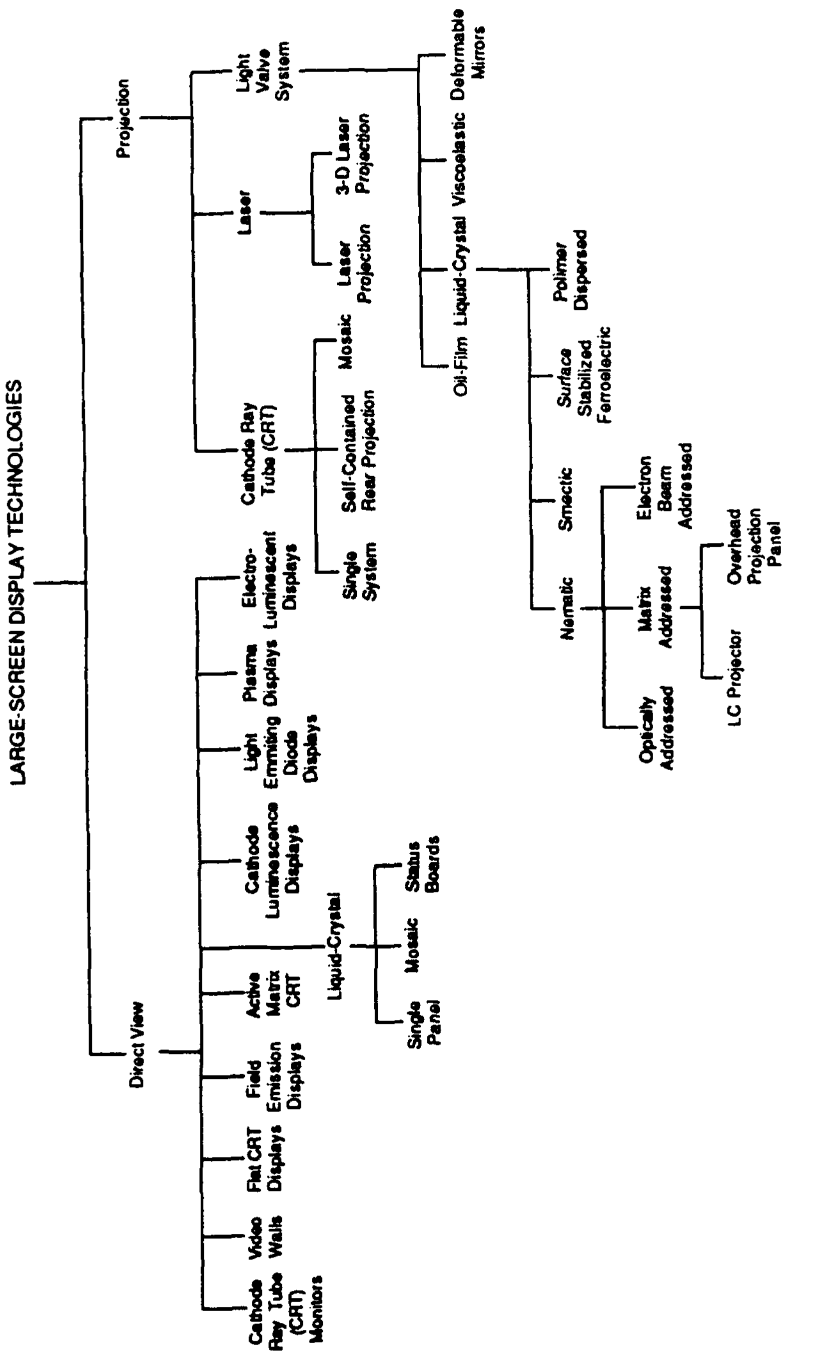


FIGURE 16.1: Large-Screen Display Technology Classification

Requirements for large-screen display selection in the command and control facility vary relative to use and size of the room in which the display will be used. Large-screen displays found in small briefing rooms (approximately 600 square feet) have a screen size of about 50 square feet, horizontal resolution of 300-1000 pixels, and a luminance output of 300-500 lumens. The briefing room must be nearly dark (<2 fc) when the large-screen display is in use.

The other type of facility where large-screen displays may be found is in the command and control center. The command center may be a two-story structure larger than 2400 square feet, where the information is presented to 10-100 personnel. Large-screen displays in this environment have a screen size of about 100 square feet, screen luminance of 10 fL, horizontal resolution of 800-1000 pixels, and a luminance output in the range of 1000 lumens. Room illuminance in the command center is adjustable from about 5-15 fc, but a more normal office illuminance of 75 fc is often requested. Full-color capability is required of large-screen displays in both facilities. Large-screen displays that are larger, brighter, and have more resolution are desired for the command and control environment.

Selecting or designing a large-screen display, especially a projection display, may be more complex than for other workstations. The effects of ambient illuminance, observer location, and type of data to be displayed are critical in implementing large-screen display technology. Presentation requirements for data not only relate to one's visual acuity for symbol size and contrast when dealing with projection technology but also to screen size, screen format, symbol luminance, and screen gain. For example, as screen gain increases, the ability to view the screen off the center line decreases. However, a certain amount of screen gain may be necessary to present an image of the necessary contrast, given the expected ambient illuminance in the room. In addition, symbols, graphics, and text should be designed to compensate for the degraded viewing conditions that may exist in the command and control center environment due to a number of factors. Implementing large-screen displays into the command and control environment should always take into account the environmental factors, as well as the information display requirements.

16.6.1 Character Dimensions

Because information is often viewed off the center axis, use character sizes between 10 and 20 minutes of visual arc and with a minimum of a 10 x 14 dot matrix format.

16.6.2 Stroke Width

The ratio of character stroke width to character height should be 1:6 to 1:10. Characters with double stroke widths should be used in situations requiring off-axis, longer distance, and/or viewing under difficult lighting conditions.

16.6.3 Luminance

Typical office ambient level is greater than 75 fc, whereas typical command and control centers are 5-15 fc.

Modulated output luminance, spatially averaged over the full screen, should be 300-400 lumens for small conference rooms and command posts and 750-2000 lumens for a control center, assuming 20-40 footcandles (fc) ambient lighting in each case.

16.6.4 Size versus Luminance

To ensure legibility, small characters (5 min. arc) require contrast ratios of 15-20:1, whereas large characters (>20 min. arc) require contrast ratios of 1.5-5:1.

16.6.5 Aspect Ratio

Aspect is the ratio of the horizontal to vertical dimensions of a character or image. Character aspect ratio should be approximately 1.33:1.48 (width:height ratio).

16.6.6 Modulation Depth

A display should deliver at least 15% visual contrast when measured as modulation depth [$(L_{max}-L_{min})/L_{max}$], when an alternating pixel pattern is displayed at normal luminance levels.

16.6.6.1 Contrast

The contrast ratio between the reflected luminance of the screen with a projected light source and the reflected luminance of the screen without a projected light source should be approximately 500:1.

16.6.6.2 Polarity

Negative contrast (black characters on a white background) should be used.

16.6.7 Displayed Data Characteristics

16.6.7.1 Quantity of Data Displayed

Avoid displaying too much data. As with standard displays, consider data type, amount, and appropriate sequence of presentation in designing large-screen display screens.

16.6.7.2 Color-Coding Requirements

If displaying color-coded targets, use only a neutral color such as grey for the background.

16.6.8 Projection Equipment

16.6.8.1 Position of Projection Equipment

Minimize glare potential by positioning projection equipment so it is not readily visible to viewers.

16.6.8.2 Optical Distortion

To minimize optical distortions, ensure image source equipment and the projection screen are fully parallel. Electronic or optical distortion-compensating devices may be used to compensate for any remaining distortion and to assure clarity of displayed information.

16.6.8.3 Field of Projection

Consider using rear projection or other direct view large-screen displays when increased contrast demands are encountered and/or when there is a need to position personnel in the field of projection.

16.7 STEREOSCOPIC/3D DISPLAYS

The display of three-dimensional (3D) images and graphics is an emerging technology that may benefit future military applications. Examples of where three-dimensional technology may be used are in battlefield and theater of operations analysis, photo-interpretation, teleoperation, air space control, and training and simulation exercises. The goal of introducing three-dimensional displays is to improve user performance and increase naturalness of the inter-action. Currently, most three-dimensional technology is experimental and, as such, is not suitable for an operational command and control environment, although a few stereographic and true three- dimensional electronic displays can be commercially purchased.

Because traditional display technology is a two-dimensional media, it has not been able to take full advantage of the human visual system to interpret complex spatial data. Binocular depth information, such as vergence or horizontal disparity, normally in the scene, are not available in the traditional electronic display. Compensation for this has been accomplished by using monocular cues, such as interposition, shading, and perspective. However, improvements in naturalness of the display and potential for gains in human performance with computing systems have stimulated the development of systems that make use of the stereoscopic capabilities of the human visual system.

Three-dimensional display technology is classified as stereoscopic and autostereoscopic. The major criterion distinguishing stereoscopic displays from autostereoscopic displays is that the latter requires no special viewing aids to see the 3D image. There is also a difference in the amount of information necessary to create the 3D image.

Stereoscopic displays create the 3D image by requiring an observer to wear a pair of glasses that provides separate images to the left and right eye. When the alternate fields are presented to the eye sequentially at the appropriate rate, the illusion of depth is created. The temporal phase difference that accounts for the stereopsis creating the 3D illusion has been implemented normally by requiring the viewer to wear a pair of glasses containing either shuttered lenses, polarized lenses, or red and green lenses. By synchronizing the image presentation to the operation of the glasses, images corresponding to the left and right scenes are presented to the viewer and the illusion of depth is created.

Autostereoscopic displays, by contrast, can be viewed directly. These displays generally use a multiplanar approach to add depth to a two-dimensional image. Examples of this type of 3D display are BBN's SpaceGraph 3D Display System (uses a flexible mirror to provide the z axis), Tectronix liquid crystal shutter 3D display (uses LCD technology together with a CRT display to create a 3D effect), and Texas Instruments' Omniview (uses a rotating multi-planar surface to produce the z axis). Holography has also produced three-dimensional images, but none to date has been created in real time.

While innovative technology to provide 3D images is becoming available, no clear guidance outlines where stereoscopic displays might best affect task performance or subjective image quality. Additionally, no database derived from applied vision or human factors research currently exists for developing application guidelines for this new technology. Consequently, the system designer must be cautious in applying 3D display technology in the military environment. Current technology often limits field of view, number of observers, and kind of data that can be presented. It also may exacerbate visual deficiencies that normally have little effect on task performance. Guidance presented here is by no means complete. Many questions remain unanswered, both in terms of human visual response to artificially generated depth from electronic displays and the ways best to enhance performance using this technology.

16.7.1 Purposeful

Presenting 3D information must be purposeful to benefit the user. That is, 3D displays should be associated with the type of work to be performed and required for task completion.

16.7.2 System Performance

Presentation of 3D or depth information should not slow information updates, degrade other aspects of system performance, or degrade image quality.

16.7.3 Interocular Crosstalk

Interocular crosstalk or bleed-through occurs in stereoscopic displays when images intended for the left eye are seen by the right eye and vice versa. Because this compromises the observer's ability to fuse the image and perceive it as a three-dimensional object, there should be zero interocular crosstalk between the two images.

16.7.4 Color Coding

Avoid saturated primary colors, as these colors may evoke depth perceptions that may be inconsistent with stereopsis, affecting the perception of depth. Designers should use secondary colors rather than saturated primary colors in coding stereoscopic images.

16.7.5 Symbols

When displaying symbols, disparity should range from 0 to 20 minutes of arc in both crossed and uncrossed directions.

16.7.6 Dynamic Depth Displays

When using dynamic depth displays, the temporal modulation of stereopsis should be approximately 1 Hz to ensure the most accurate perception of stereo-motion.

16.7.7 Depth-Coded Objects

Spatially separate depth-coded objects in stereoscopic images to eliminate disparity averaging, crowding, or repulsion.

16.7.8 Size Scaling

Image size should be scaled according to the disparity of the image. When accurate size perception is critical to task performance, size scaling should be done for an individual observer.

16.7.9 Luminance

Because brightness is also a depth cue (bright objects are viewed as nearer), luminance should be co-modulated with stereopsis.

16.8 TOUCH INTERACTIVE DEVICES

Display parallax is the apparent displacement of an object displayed on a curved CRT screen and viewed through a flat TID.

A touch interactive device (TID) is an input device that permits a user to interact with the system by pointing to objects on the display. The TID is addressed in this section, because some implementations of touch technology can severely degrade the quality of the displayed image. Degradation in image quality using TIDs may result from decreases in display luminance, reductions in display resolution due to visibility of conductors or the device material, increased susceptibility to glare, and dirt on the display surface resulting from touching the display surface. Display parallax, caused by separation between the touch surface and the touch targets, may also contribute to problems with implementing TIDs.

There are six basic types of touch-screen display technologies, each having an impact on display parallax, transmissivity of light, and glare. Each is briefly discussed below. The designer needs to be aware of the advantages and disadvantages of each type of TID when selecting hardware and designing interfaces using TIDs.

a. Fixed-wire TIDs place wires, either in parallel or in grid fashion, in front of the display. Finger contact with the wire(s) signifies the x,y coordinate of the user's response. This technology is associated with minimal parallax, 70-80% transmissivity, and a medium to high degree of TID glare.

b. Capacitive TIDs consist of a transparent conductive film on a glass overlay. Touching this surface changes the small electrical signal passing through the surface, and this signal is converted into the corresponding x,y coordinate. This technology is associated with minimal parallax, 85% transmissivity, and a medium degree of TID glare.

c. Resistive membrane TIDs are "sandwich" devices in which a touch results in the contact of two conductive layers. Specific current and voltage levels are associated with individual x,y coordinates. This technology is associated with minimal parallax, 50-60% transmissivity, and a high degree of TID glare.

d. Infrared (IR) or light-emitting diode TIDs use IR transmitters along two perpendicular sides of the display frame and photocell receptors along the opposite sides of the frame. A user touch breaks the resulting matrix of light beams, and the appropriate x,y coordinates of the touch are thus determined. This technology is associated with no parallax problems in seeing the display (although a noticeable degree of parallax exists between the plane of the IR grid and the screen surface for touch responses), 100% transmissivity, and no TID-related glare.

e. Surface acoustic wave TIDs operate in similar fashion to IR TIDs, except that the matrix overlay is one of ultrasonic sound beams rather than IR beams. Another approach, called the reflective array, uses a piezoelectric transmitter and a series of reflectors and receivers. Touch x,y coordinates are determined by differential timings in reception of the acoustic waves. At least some devices require glass overlay screens. This technology is associated with minimal parallax, 92% transmissivity, and a medium degree of TID glare.

f. Pressure-sensitive devices use strain gauges mounted between the display screen and an overlay. Output voltages of these strain gauges are encoded into the appropriate x,y coordinates. This technology is associated with minimal parallax and zero TID glare. Figures for transmissivity are not applicable because the overlay is built into the display screen.

16.8.1 Parallax

TID/display parallax should be minimized because it has been shown to lead consistently to poorer entry time and touch count performance.

16.8.2 Specular Glare

Specular glare should be minimized for applications using TIDs.

REFERENCE LIST

Paragraph	References
16.0	Benson and Farrell (1988); Lloyd et al. (1991); Dye and Snyder (1991); Decker et al. (1991); Reger et al. (1989); Biberman and Tsou (1991)
16.1	Tannas (1985) page 11; Goode (1991); Beaton and Knox (1987)
16.1.1	Reger et al. (1989); Petrun et al. (1985)
16.1.1.1	Laycock (1985)
16.1.1.2	DoD (1989a) par. 5.2.6.8.3
16.1.1.3	ISO (1991) Part 3 Addendum
16.1.1.4	ISO (1991) Part 3 Addendum
16.1.2	ISO (1991) Part 3 Addendum
16.1.3	ISO (1991) Part 3 Addendum
16.1.4	Lloyd et al. (1991)
16.1.4.1	Dye and Snyder (1991); Lloyd et al. (1991)
16.1.4.2	Lloyd et al. (1991); Laycock (1985)
16.1.4.3	Lloyd et al. (1991)
16.1.4.4	Dye and Snyder (1991); Lloyd et al. (1991)
16.1.4.5	Lloyd et al. (1991)
16.1.4.6	Dye and Snyder (1991); Lloyd et al. (1991); Snyder (1988); McNeese and Katz (1986)

REFERENCE LIST (Cont.)

Paragraph	References
16.2	Snyder (1980); Goode (1991)
16.2.1	Petrun et al. (1985); Payne (1983)
16.2.1.1	Payne (1983); Muracka et al. (1989)
16.2.1.2	Biberman and Tsou (1991), Reger et al. (1989); Cristensen et al. (1985)
16.2.2	Kubota et al. (1988), Kubota Murushige Takabayashi and Kobayashi (1986)
16.2.3	Payne (1983)
16.2.4	Payne (1983) Muracka et al. (1989)
16.3	Tannas (1985) Chapter 10; Goode (1991); Snyder (1980)
16.4	Snyder (1980); Goode (1991)
16.5.1	Reger et al. (1989); Petrun et al.(1985)
16.5.2	Morse (1985)
16.5.3	ISO (1991) Part 3
16.5.4	Hunter (1988)
16.5.5	Woodson et al. (1992)
16.6	Breen (1987); Breen (1989); Hurley and Hoffberg (1991); Carbone and MacIver (1987)
16.6.1	McNeese and Katz (1986); Shurtleff et al. (1982)
16.6.2	Woodson et al. (1992)
16.6.3	Blaha (1990); Breen (1989)
16.6.4	Shurtleff et al. (1982)
16.6.5	Woodson et al. (1992)
16.6.6	Blaha (1990)

REFERENCE LIST (Cont.)

Paragraph	References
16.6.6.2	McNeese and Katz (1986)
16.6.7.1	Woodson et al. (1992)
16.6.7.2	Woodson et al. (1992)
16.6.8.1	Woodson et al. (1992)
16.6.8.2	Woodson et al. (1992); Vance (1987)
16.6.8.3	Woodson et al. (1992)
16.7	Veron et al. (1990); Yeh and Silverstein (1989); Reinhart (1990); Beaton (1990); Snyder (1984); Williams (1990)
16.7.1	Beaton (1990)
16.7.2	Beaton (1990)
16.7.3	Yeh and Silverstein (1991)
16.7.4	Yeh and Silverstein (1991)
16.7.5	Yeh and Silverstein (1989); Yeh and Silverstein (1991)
16.7.6	Yeh and Silverstein (1991)
16.7.6	Yeh and Silverstein (1991)
16.7.7	Yeh and Silverstein (1991)
16.7.8	Yeh and Silverstein (1991)
16.7.9	Yeh and Silverstein (1991)
16.8	Dominessy (1989)
16.8.1	Baggen et al. (1988)
16.8.2	Baggen et al. (1988)

This Page Intentionally Left Blank.

APPENDIX A - REFERENCES

Abramson, S. R., L. H. Mason, and H. L. Snyder. 1983. "The Effects of Display Errors and Font Styles Upon Operator Performance with a Plasma Panel." In Proceedings of the Human Factors Society - 27th Annual Meeting, pp. 28-32. HFS, Santa Monica, California.

ACM - see Association for Computing Machinery, Inc.

Ambron, S., and K. Hooper. 1990. Learning with Interactive Multimedia: Developing and Using Multimedia Tools in Education. Microsoft Press, Redmond, Washington.

Andrews, J. R., W. E. Haas, and M. D. Rainsdon. 1989. "Holographic Displays for the Man-Machine Interface." Optical Engineering, 28(6):643-649.

Andriole, S. J., and L. A. Adelman. 1990. "Prospects for Cognitive Systems Engineering." Presentation, Center for Multidisciplinary Information Systems Engineering, Drexel University, Philadelphia.

Antin, J. 1988. "An Empirical Comparison of Menu Selection, Command Entry and Combined Modes of Computer Control." Behavior and Information Technology 7(2) 173:173-182.

Armstrong, H. G. 1988. "Colored Light-Emitting Diodes: Use of Red or Green in High Ambient Illumination." In Engineering Data Compendium: Human Perception and Performance, eds. K. R. Boff and J. E. Lincoln, Vol. 3:2260-2261. Harry G. Armstrong Aerospace Medical Research Laboratory, Wright-Patterson Air Force Base, Ohio.

Aspillaga, M. 1991. " Screen Design: Location of Information and Its Effects on Learning." Journal of Computer-Based Instruction 18(3):89-92.

Association for Computing Machinery, Inc. (ACM). 1989. Human Factors in Computing Systems: CHI '89 Conference Proceedings, Austin, Texas. ACM Press, New York.

Association for Computing Machinery, Inc. (ACM). 1990. Resources in Human-Computer Interaction. ACM Press, New York.

Auerbach Publishers, Inc., ed. 1981. Practical Data Base Management. Reston Publishing Company, Inc., Reston, Virginia.

Avery, L. W., R. V. Badalamente, S. E. Bowser, P. A. O'Mara, S. E. Reynolds. 1990. Human Factors Design Guidelines for the Army Tactical Command and Control System (ATCCS) Soldier-Machine Interface. Pacific Northwest Laboratory (PNL) for the U.S. Army Tactical Command and Control System (ATCCS) Experimentation Site (AES), Fort Lewis, Washington.

Badler, N., and B. Webber. 1991. "Task Communication Through Natural Language and Graphics." AI Magazine 11(5):71-73.

Baeker, R. 1980. "Towards an Effective Characterization of Graphical Interaction." IFIP Workshop on Methodology of Interaction, pp. 127-147.

Baggen, E. A., H. L. Snyder, and M. R. Miller. 1988. "A Human Factors Evaluation of Current Touch-Entry Technologies." In SID International Symposium Digest of Technical Papers, pp. 259-262. SID, Playa Del Rey, Kingsbeach, California.

Bailey, R. W. 1982. Human Performance Engineering: A Guide for System Designers. Prentice Hall, Englewood Cliffs, New Jersey.

Bailey, R. W. 1989. Human Performance Engineering: Using Human Factors/Ergonomics to Achieve Computer system Usability. Prentice Hall, Englewood Cliffs, New Jersey.

Baker, C. 1988 . Text-Editing Performance as a Function of Screen Size: A Pilot Study. 612716.H7000700011, U.S. Army Human Engineering Laboratory, Aberdeen, Maryland.

Barnett, B. J. 1990. "Aiding Type and Format Compatibility for Decision Aid Interface Design." In Proceedings of the Human Factors Society 34th Annual Meeting, pp. 1552-1556. HFS, Santa Monica, California.

Barten, P. G. J. 1991. "Resolution of Liquid-Crystal Displays." In Society for Information Display 91 Digest, pp. 772-775. SID, New York.

Beaton, R. J. 1990. "Human Factors Engineering of 3D Stereoscopic Display Systems." In Stereographics, 17th International Conference on Computer Graphics and Interactive Techniques, Association for Computing Machinery, Special Interest Group on Graphics, Dallas, Texas.

Beaton, R. J., and S. T. Knox. 1987. "Flat Panel Image Quality." In Society for Information Display 87 Digest, pp. 115-118. SID, New York.

Benbasat, I., and Y. Wand. 1984. "A Structured Approach to Designing Human-Computer Dialogues." International Journal of Man-Machine Studies 21(2):105-126.

Benson, B. L., and J. E. Farrell. 1988. "The Effect of Character Height-to-Width Ratio on CRT Display Legibility." In Society for Information Display 88 Digest, pp. 340-343. SID, New York.

Biberman, L. M., and B. Tsou. 1991. Image Display Technology and Problems with Emphasis on Airborne Systems. Technical Report No. AD-B1157 161, Defense Technical Information Center, Alexandria, Virginia.

Bidgoli, H. 1989. "DSS Products Evaluation: An Integrated Framework." Journal of Systems Management 40(11):27-28

Billingsley, P. A. 1988. "Taking Panes: Issues in the Design of Windowing Systems." In Handbook of Human-Computer Interaction, ed. M. Helander, pp. 413-434. Elsevier Science Publishers B.V., Amsterdam.

Blaha, R. J. 1992. Emerging Large Screen Display Technology. Technical Report No. M92B0000010, MITRE Bedford, Massachusetts.

Blaha, R. J., C. D. Crotty, and D. F. Martin. 1992. Evaluation of Large-Screen Display Systems in Workstation Development Environment. Working Paper No. 92B0000081, The MITRE Corp., Bedford, Massachusetts.

Blaha, R. J. 1990. Large-Screen Display Technology Assessment for Military Applications. Technical Report No. M90-16, MITRE Bedford, Massachusetts.

Blaser, A., and M. Zoeppriz, eds. 1983. Enduser Systems and Their Human Factors. Springer-Verlag, Berlin.

Blattner, M. M., D. A. Sumikawa, and R. M. Greenberg. 1989. "Earcons and Icons: Their Structure and Common Design Principles." Human-Computer Interaction 4:11-44.

Bobko, D. J., P. Bobko, and M. A. Davis. 1986. "Effect of Visual Display Scale on Duration Estimates." Human Factors Society, 28(2):153-158. HFS, Santa Monica, California.

Boehm-Davis, D. A., R. W. Holt, M. Koll, G. Yastrop, and R. Peters. 1989. "Effects of Different Data Base Formats on Information Retrieval." Human Factors 31(5):579-592.

Bosman, D. 1991. "Simulation of the Perception of a Computer-Generated Display Image." Proceedings of Society for Information Display, 32(1):3-12. SID, New York.

Bowser, S. E. 1991. "Review of Army Tactical Command and Control System Soldier-Machine Interface Functional Issues." Pacific Northwest Laboratory (PNL) Task Order 91-04, Subtask 2b for the U.S. Army Tactical Command and Control System (ATCCS) Experimentation Site (AES), Fort Lewis, Washington.

Breen, P. T. 1987. "Functional Requirements for C3I Large Screen Displays." In Large Screen Projection Displays, 760:2-8. SPIE Press, Bellingham, Washington.

Breen, P. T. 1989. Functional Requirements for C3I Large Screen Displays. Technical Report No. M89-01, MITRE Bedford, Massachusetts.

Breen, P. T., and H. C. Masterman. 1990. Modulation Depth Metrics for the Large Screen Displays in NASA's Mission Control Centers. Working Paper No. 28736, The MITRE Corp. Bedford, Massachusetts.

Breen, P. T., P. E. Miller-Jacobs, and H. H. Miller-Jacobs. 1987. "Color Displays Applied to Command, Control, and Communications (C3) Systems." In Color and the Computer, ed. J. H. Durrett, pp.171-187. Academic Press, Inc., Boston.

Brooks, J. D., R. D. Gilson, and H. R. Myler. 1990. "Display Design Guide for Bivisual Media." Proceedings of the Human Factors Society 34th Annual Meeting 1990. HFS, Santa Monica, California.

Brosda, V., and G. Vossen. 1988. "Update and Retrieval in a Relational Database Through a Universal Schema." ACM Transactions on Database Systems 13(4):449-485.

Brosley, M., and B. Shneiderman. 1978. "Two Experimental Comparisons of Relational and Hierarchial Database Models." International Journal of Man-Machine Studies 10:625-637.

Brown, C. M. 1989. Human-Computer Interface Design Guidelines. Ablex Publishing Corporation, Norwood, New Jersey.

Brown, C. M., D. B. Brown, H. V. Burkleo, J. E. Mangelsdorf, R. A. Olsen, and R. D. Perkins. 1983. Human Factors Engineering Standard for Information Processing Systems. LMSC-D-877141 (Revision of C410), Lockheed Missiles and Space Company, Inc., Los Angeles.

Bullinger, H. J., K. P. Fähnrich, and J. Ziegler. 1987. "Software-Ergonomics: History, State-of-the-Art, and Important Trends." In Cognitive Engineering in the Design of Human-Computer Interaction and Expert Systems, ed. G. Salvendy, pp. 307-316. Elsevier Science Publishers B.V., Amsterdam.

Campbell, J. A., and S. P. Ross. 1987. "Issues in Computer-Assisted Interpretation of Graphs and Quantitative Information." In Cognitive Engineering in the Design of Human-Computer Interaction and Expert Systems, ed. G. Salvendy, pp. 473-480. Elsevier Science Publishers B.V., Amsterdam.

Carbone, R. M., and D. MacIver. 1987. "A Survey of Large-Screen Displays of C3I Applications." In Large Screen Projection Displays, 760:6-10. SPIE Press, Bellingham, Washington.

Carey, J. M., ed. 1988. Human Factors in Management Information Systems. Ablex Publishing Corporation, Norwood, New Jersey.

Carlson, E. D., and W. Metz. 1980. "Integrating Dialog Management and Data Base Management." Information Processing: Proceedings of the IFIP Congress, Vol. 80, pp. 463-468. North-Holland Publishing Co., Amsterdam.

Carroll, J. M., and S. A. Mazur. 1986. "LisaLearning." Computer 19(11):35-49.

Carroll, J. M., and J. McKendree. 1987. "Interface Design Issues for Advice-Giving Expert Systems." Communications of the ACM 30(1):14-31.

Chao, B. P. 1987. "Prototyping a Dialogue Interface: A Case Study." In Cognitive Engineering in the Design of Human-Computer Interaction and Expert System, ed. G. Salvendy, pp. 357-363. Elsevier Science Publishers B.V., Amsterdam.

Chao, B. P. 1986. Design Guidelines for Human-Computer Dialogues. SAND86-0259, Sandia National Laboratories, Albuquerque, New Mexico.

Chapnick, P. 1989. "Ushering in a New Era." AI Expert, Feb. 1989, p. 7-8, Miller Freeman, Inc.

Christensen, C., E. A. Baggen, and H. L. Snyder. 1985. "Performance Measures and Subjective Evaluations for Two Color Displays." In Proceedings of the Human Factors Society - 29th Annual Meeting, pp. 1075-1078. Santa Monica, California.

Clarke, A. A. 1986. "A Three-Level Human-Computer Interface Model." International Journal of Man-Machine Studies 24:503-517.

Cowen, M. 1991. A Comparison of Four Types of Feedback During Computer-Based Training (CBT). NPRDC-TR-92-2, Navy Personnel Research and Development Center, San Diego, California.

Crolotte, A., J. Saleh, and A. Freedy. 1980. "Human Decision Processes in Command and Control of Marine Amphibious Brigade Operations, IEEE, pp. 1216-1220.

Cuff, R. N. 1980. "On Casual Users." International Journal of Man-Machine Studies 12(2):163-187.

Davies, S. P., A. J. Lambert, and J. M. Findlay. 1989. "The Effects of the Availability of Menu Information During Command Learning in a Word Processing Application." Behavior and Information Technology 8:135-144.

Decker, J. J., C. J. Dye, C. J. C. Lloyd, and H. L. Snyder. 1991 The Effects of Display Failures and Symbol Rotation on Visual search and Recognition Performance. Technical Memorandum 4-91. U.S. Army Human Engineering Laboratory, Aberdeen Proving Ground, Maryland.

Decker, J. J., P. L. Kelly, K. Kurokawa, and H. L. Snyder. 1991. The Effect of Character Size, Modulation, Polarity, and Font on Reading and Search Performance in Matrix-Addressable Displays. Technical Memorandum 6-91. U.S. Army Human Engineering Laboratory, Aberdeen Proving Ground, Maryland.

DoD (see U.S. Department of Defense)

Doll, T. J. 1991. "Electro-Optic/Infrared Technology and the Human-Machine Interface." In Proceedings of the Human Factors Society - 35th Annual Meeting, pp. 1500-1501. HFS, Santa Monica, California.

Dominessy, M. E. 1989. A Literature Review and Assessment of Touch Interactive Devices. Technical Memorandum 11-89. U.S. Army Human Engineering Laboratory, Aberdeen Proving Ground, Maryland.

Dumas, J. S. 1988. Designing User Interfaces for Software. Prentice Hall, Englewood Cliffs, New Jersey.

Durrett, H. J., ed. 1987. Color and the Computer. Academic Press, Inc., Boston.

Dye, C. J., and H. L. Snyder. 1991. The Effects of Display Failure, Polarity, and Clutter on Visual Search for Symbols on Cartographic Images. Technical Memorandum 9-91, U.S. Army Human Engineering Laboratory, Aberdeen Proving Ground, Maryland.

Ehrenreich, S. L. 1981. "Query Languages: Design Recommendations Derived from the Human Factors Literature." Human Factors 23(6):709-725.

Ehrhart, L. S. 1990. "New Approaches to Human-Computer Interaction Research and Design for Decision Aiding Systems." In Proceedings of the 5th IEEE International Symposium on Intelligent Control. IEEE Computer Society Press, Los Alamitas, California.

Elkerton, J., R. C. Williges, J. A. Pittman, and J. Roach. 1982. "Strategies of Interactive File Search" in Proceedings of the Human Factors Society. HFS, Santa Monica, California.

Farooq, M. U., and W. D. Dominick. 1988. "A Survey of Formal Tools and Models for Developing User Interfaces." International Journal of Man-Machine Studies 29:479-496.

Feldman, M. B., and G. T. Rogers. 1982. "Toward the Design and Development of Style-Independent Interactive Systems." In Human Factors in Computer Systems: Proceedings, March 15-17, 1982, pp. 111-116. ACM, New York.

Fenchel, R. 1981. "An Integrated Approach to User Assistance." ACM SIGSOC Bulletin 13 2:98-104.

Fernandes, K., and K. E. Maracle. 1991. Design Standards for the Command and Control Information Navigation and Training System (CINTS). Technical Note 1660, Naval Ocean Systems Center, San Diego, California.

Finke, D. J., and M. M. Lloyd. 1988. "Guidelines and Tools for Human-Decision Aid Interface Design and Evaluation." In Proceedings of the Fifth Annual Workshop on Command and Control Decision Aiding, pp. 170-193, September 1988.

Flynn, R. R. 1987. An Introduction to Information Science. Marcel Dekker, Inc., New York.

Frost, R. A. 1984. Database Management Systems. McGraw-Hill, New York.

Funk, H. L. 1989. "Information Display - An Overview and Trends." In Proceedings of the Institution of Electrical and Electronics Engineers, pp. 2-1 - 2-7. IEEE, Washington, D.C.

Gaines, B. R., and M. L. G. Shaw. 1986. "Foundation of Dialog Engineering: The Development of Human-Computer Interaction." Part II. International Journal of Man-Machine Studies 24:101-123.

Gaines, B. R., P. V. Facey. 1975. "Some Experience in Interactive System Development and Application." In Proceedings of the IEEE 63(6):894-911.

Galitz, W. O. 1984. Humanizing Office Automation. QED Information Sciences, Inc., Wellesley, Massachusetts.

Galitz, W. O. 1989. Handbook of Screen Format Design. QED Information Sciences, Inc., Wellesley, Massachusetts.

Garner, K. H. 1990. "20 Rules for Arranging Text on a Screen." CBT Directions:13-16.

Gery, G. 1991. Electronic Performance Support Systems. Weingarten Publications, Boston, Massachusetts.

Getler, R. 1991. "The Case for Concurrent Authoring." CBT Directions:14-22.

Gilmore, W. E., D. I. Gertman, and H. S. Blackman. 1989. "The User-Computer Interface in Process Control." In A Human Factors Engineering Handbook, pp. 21-33. EG&G Idaho, Inc., Idaho Falls, Idaho.

Gittins, D. 1986. "Icon-Based Human-Computer Interaction." International Journal of Man-Machine Studies 24:519-543.

Glasser, J., and A. Rolland. 1989. "Visual Performance Evaluation for LCD Displays: Appropriate Methods for Measuring Luminance and Contrast." Proceedings of Society for Information Display, 1077:9-20. SID, New York.

Gold, R. S., and A. G. Ledebuhr. 1985. "Full Color Liquid Crystal Light Valve Projector." In Advanced in Display Technology V, 526:51-58. SPIE Press, Bellingham, Washington.

Goode, W. F. 1991. "Status of Electronic Displays." Seminar for the Society of Information Display, SID, New York.

Gordon, S. E. 1988. "The Human Factor in Expert Systems," AI Expert, pp. 55-59.

Grabinger, R. S., and D. Amedeo. 1988. "CRT Text Layout: Perceptions of Viewers." Computers in Human Behavior 4:189-205.

Greitzer, F. 1987. Comments on draft NASA Guidelines Document: User/Computer Interaction Section. Memo on 1, 22, 1987.

Grether, W. F., and C. A. Baker. 1972. Human Engineering Guide to Equipment Design, eds., H. P. Van Cott and R. C. Kinkade. GPO, Washington, DC.

Gri!!, E. 1990. Relational Databases: A Methodical Guide for Practical Design and implementation. Ellis Harwood, New York.

Gunderson, J., G. Gruetzmacher, and N. Swanson. 1991. "Legibility of Seven Segment Numeric LED Displays: Comparisons of Two Fonts at Various Distances." In Proceedings of the Human Factors Society - 35th Annual Meeting, pp. 491-495. HFS, Santa Monica, California.

Hamel, C. J., and S. L. Clark. 1986. CAI Evaluation for the Design of Computer-Aided Instruction. Technical Report NTSC TR86-002 (DTIC No. AD-A172383). Naval Training Systems Center, Orlando, Florida.

Hannigan, S., and V. Herring. 1987. "Human Factors in Office Product Design - European Practice." In Proceedings of the Second International Conference on Human-Computer Interaction, Conference Title "Cognitive Engineering in the Design of Human-Computer Interaction and Expert Systems," ed. G. Salvendy, pp. 226-232. Elsevier Science Publishers B.V., Amsterdam.

Hansen, G. W., and J. V. Hansen. 1992. Database Management and Design. Prentice Hall, Englewood Cliffs, New Jersey.

Harrell, T. H. 1987. "Designing User-Computer Dialogues: Basic Principles and Guidelines." In Computerized Psychological Testing: Current Issues and Future Directions, Honaker, L. M., Chair. Symposium conducted at the American Psychological Association, New York.

Harter, S. P. 1986. On-line Information Retrieval. Academic Press, Orlando, Florida.

Hawkridge, D. G. 1988. Computers in Company Training. Croom Helm, London.

Heines, J. M. 1984. Screen Design Strategies for Computer-Assisted Instruction. Digital Press, Bedford, Massachusetts.

Helander, M., ed. 1988. Handbook of Human-Computer Interaction. Elsevier Science Publishers B.V., Amsterdam.

HFS - See Human Factors Society

Hildreth, C. R. 1982. Online Public Access Catalogs: The User Interface. Online Computer Library Center, Inc. Dublin, Ohio.

Hoadley, E. D. 1990. "Investigating the Effects of Color." Communications of the ACM 33(2):120-127.

Holtzman, S. 1989. Intelligent Decision Systems. Addison-Wesley, Reading, Massachusetts.

Horton, W. K. 1990. Designing and Writing Online Documentation; Help Files to Hypertext. John Wiley and Sons, New York.

Human Factors Society, Inc. (HFS). 1988. American National Standard for Human Factors Engineering of Visual Display Terminal Workstations. ANSI/HFS 100-1988, HFS, Santa Monica, California.

Humphrey, S. M., and B. J. Melloni. 1986. Databases: A Primer for Retrieving Information by Computer. Prentice Hall, Englewood Cliffs, New Jersey.

Hunter, M. W. 1988. "CRT Anti-glare Treatments, Image Quality, and Human Performance." Ph.D. Dissertation, Virginia Polytechnic Institute and State University, Blacksburg, Virginia.

Hurd, J. C. 1983. "Writing Online Help." In Proceedings of 30th International Technical Communication Conference, pp. 151-154. Society for Technical Communication, Washington, D.C.

Hurley, A. F., and L. I. Hoffberg. 1991. Evaluation of Six Large-Screen Displays. Technical Report No. 29399, MITRE Bedford, Massachusetts.

Hussein, B. 1989. "DSS Products Evaluation: An Integrated Framework." Journal of Systems Management, pp. 27-34, November 1989.

Hyland, R. M., and P. R. Boivin. 1987. Performance Characteristics of Infrared Touch Screens for Typical Workstation Applications. Technical Report No. NSU 8093, Defense Technical Information Center, Alexandria, Virginia.

IBM. 1984. Human Factors Workstations with Visual Displays, Third Edition. San Jose, California.

Inmon, W. H. 1981. Effective Data Base Design. Prentice Hall, Englewood Cliffs, New Jersey.

International Standards Organization (ISO). 1991. "Part 3 Flat Panel Addendum, Additional Notes for Reviewers." In "ISO 9241, Working Draft of ISO Committee 159 SC 4 Working Group 2." ISO 9241, ISO, Washington, D.C.

ISO (see International Standards Organization)

Jarvenpaa, S. L., and G. W. Dickson. 1988. "Graphics and Managerial Decision-Making: Research Based Guidelines." Communications of the ACM 31(6):764-775.

Katzeff, C. 1986. "Dealing With a Database Query Language in a New Situation." International Journal of Man-Machine Studies 5(1):1-17.

Katzeff, C. 1988. "The Effect of Different Conceptual Models Upon Reasoning in a Database Query Writing Task." International Journal of Man-Machine Studies 29(1):37-62.

Kawamura, T., H. Kawamura, K. Kobara, A. Saitoh, and Y. Endo. 1991. "Anti-reflection Coating for Inner Surface of CRT Faceplate." In Society for Information Display 91 Digest, pp. 49-52. SID, New York.

Kearsley, G. 1988. Online Help Systems: Design and Implementation. Ablex Publishing, Norwood, New Jersey.

Kelley, J. F. 1984. "An Iterative Design Methodology for User-Friendly Natural Language Office Information Applications." ACM Transactions on Office Information Systems 2(1):26-41.

Kelster, R. S., and G. R. Galloway. 1983. "Making Software User Friendly: An Assessment of Data Entry Performance." Proceedings of the Human Factors Society. HFS, Santa Monica, California.

Kieras, D. E., and S. Boviar. 1984. "The Role of a Mental Model in Learning to Operate a Device." Cognitive Science 8:255-273.

Klien, G. A., and D. MacGregor. 1988. "Knowledge Elicitation of Recognition-Primed Decision-Making." Technical Report 799. U.S. Army Research Institute Field Office, Fort Leavenworth, Kansas.

Korth, H. F., A. Silbershatz. 1986. Database System Concepts. McGraw-Hill, New York.

Krause, J. 1980. "Natural Language Access to Information Systems: An Evaluation Study of Its Acceptance by End Users." Information Systems 5(4):297-318.

Kubota, H., T. Marushige, T. Takabayashi, and S. Kobayashi. 1986. "LCD Legibility." In Society for Information Display 88 Digest, pp. 157-160. SID, New York.

Kubota, H., T. Marushige, C. M. Gomes, and S. Kobayashi. 1988. "Legibility of Multiplexed Dot-Matrix LCDS: The Effect of Surface Reflection." In Proceedings of Society for Information Display (SID), 29(3):213-216. SID, New York.

Lansdale, M. W. 1988. "On the Memorability of Icons in an Information Retrieval Task." Behavior and Information Technology 7:131-151.

Lanzetta, T. M., and N. D. Lubart. 1988. "Comparing the Readability of Display Technologies: Paper, CRT, and LCD." In Society for Information Display 88 Digest, pp. 336-339. SID, New York.

Larson, H. T. 1989. "Large Automated Flat Situation Display for the Commander." Maintaining Start-of-the Art in ACCS:3-44. Army Science Board Summer Study.

Laurel, B., ed. 1990. The Art of Human Computer Interface Design. Addison-Wesley, Reading, Massachusetts.

Laverson, A., K. Norman, and B. Shneiderman. 1987. "An Evaluation of Jump-Ahead Techniques in Menu Selection." Behavior and Information Technology 6(2):97-108.

Laycock, J. 1985. "The Legibility of Passive Displays". Proceedings of Society for Information Display, 26(2):89-93. SID, New York.

LeMay, M. 1988. "Why Some Decision Aids Work and Others Do Not." Proceedings of the 1988 IEEE International Conference on Systems, Man, and Cybernetics, pp. 227-229.

Lewis, H. V., and J. J. Fallesen. 1989. Human Factors Guidelines for Command and Control Systems: Battlefield and Decision Graphics Guidelines. Research Project 89-01. U.S. Army Research Institute for the Behavioral and Social Sciences, Alexandria, Virginia.

Lickteig, C. W. 1989. Design Guidelines and Functional Specifications for Simulation of the Battlefield Management Systems (BMS) User Interface. U.S. Army Research Institute for the Behavioral and Social Sciences, Alexandria, Virginia.

Lloyd, C. J. C., J. J. Decker, and H. L. Snyder. 1991. The Effects of Line and Cell Failures on Reading and Search Performance Using Matrix-Addressable Displays. Technical Memorandum 7-91. U.S. Army Human Engineering Laboratory, Aberdeen Proving Ground, Maryland.

Lloyd, C. J. C., J. J. Decker, K. Kurokawa, and H. L. Snyder. 1988. "Effects of Line and Cell Failures on Reading and Search Performance Using Matrix-Addressable Displays." In Society for Information Display 88 Digest, pp.344-346. SID, New York.

Lochovsky, F. H, and D. C. Tsichritzis. 1984. "Querying External Databases." In Human Factors and Interactive Computer Systems, ed. Y. Vassiliou. Ablex Publishing Corporation, Norwood, New Jersey.

Lysaght, R. J., R. Harris, and W. Kelly. 1988. "Artificial Intelligence for Command and Control." Technical Report 2122. U.S. Army Communications and Electronics Command, Fort Monmouth, New Jersey.

Ma, P., F. H. Murphy, and E. A. Stohr. 1989. "A Graphics Interface for Linear Programming," Communications of the ACM 32:996-1012.

MacGregor, J. M., and E.S. Lee. 1988. "A Feature Matching Approach to the Retrieval of Graphical Information." Behavior and Information Technology 7(4):457-465.

Magnavox, Inc. 1985. Appendix B - Application of Data Processing Networking Techniques for Army Command and Control Systems (ACCS) Task 1 Subtask 4. Report No. 11-89 on Contract No. DAAK11-84-D-0006, Magnavox, Inc., Fort Wayne, Indiana.

Main, R. and D. Paulson. 1988. Guidelines for the Development of Military Training Decision Aids. NPRDC TR 88-16. Navy Personnel Research and Development Center, San Diego, California.

Mallary, T. C. 1985. "Design of the Human-Computer Interface for a Computer Aided Design Tool for the Normalization of Relations." Master's Thesis, Air Force Institute of Technology, Wright Air Force Base, Ohio.

Martin, J. 1983. Managing the Data-Base Environment. Prentice Hall, Englewood Cliffs, New Jersey.

Masiaszek, L. A. 1990. Database Design and Implementation. Prentice Hall, New York.

Matthews, M. L. 1987. "The Influence of Color on CRT Reading Performance and Subjective Comfort Under Operational Conditions." Applied Ergonomics 18.4:323-328.

McCann, C. 1988. Final Report from "Research Study Group 12 (RSG.12) on Computer-Human Interaction in Command and Control." Document AC/243 (Panel 8/RSG.12) D/7, DCIEM, Downsview, Ontario.

McCann, P. H. 1983. Methods for Improving the User-Computer Interface. Report No. NPRDC TR 83-29, Navy Personnel Research and Development Center, San Diego, California.

McKeown, P. E., J. J. Fallesen, M. S. Perkins, and C. G. Ross. 1991. Operations Planning Tools (OPT) Functional Description. U.S. Army Research Institute Product 91-09 (AD-A235 665), Fort Leavenworth, Kansas.

McNeese, M. D., and L. Katz. 1986. "Legibility Evaluation of a Large-Screen Display System Under Medium Ambient Illumination." In Society for Information Display 86 Digest, pp. 142-145. SID, New York.

Milgram, P., D. Drascic, and J. J. Grodski. 1991. "Enhancement of 3-D Video Displays by Means of Superimposed Stereo-Graphics." In Proceedings of the Human Factors Society - 35th Annual Meeting, pp. 1457-1461. HFS, Santa Monica, California.

Milheim, W.D., and B. L. Martin. 1991. "Theoretical Basis for the Use of Learner Control: Three Different Perspectives." Journal of Computer-Based Instruction 18(3):99-105.

Miller, P. J. 1991. NMR Display Prototype Experiment Results. Technical Report No. AD-A239 131, Defense Technical Information Center, Alexandria, Virginia.

Minasi, M. 1990. "Bayes and Simple Expert Systems." AI Expert, pp. 13-15.

Mitchell, D. K., and K. P. Kysor. 1992. A Preliminary Evaluation of the Prototype Tactical Computerized Interactive Display. Technical Memorandum 2-92. U.S. Army Human Engineering Laboratory, Aberdeen Proving Ground, Maryland.

Mittal, S. 1985. "Knowledge Acquisition from Multiple Experts. AI Magazine, pp. 32-36.

Miyashita, T., and T. Uchida. 1990. "Cause of Fatigue and its Improvement in Stereoscopic Displays." Proceedings of Society for Information Display, 31(3):249-254. SID, New York.

Moran, T. P. 1981. "The Command Language Grammar: A Representation for the User of Interactive Computer Systems." International Journal of Man-Machine Studies 15:3-50.

Morse, R. S. 1985. "Glare Filter Preference: Influence of Subjective and Objective Indices of Glare, Sharpness, Brightness, Contrast and Color." In Proceeding of the Human Factors Society 29th Annual Meeting, pp.782-786. HFS, Santa Monica, California.

Muracka, T., M. Kawamura, and H. Uesako. 1989. "Readability on the Positive Type Liquid Crystal Display Devices with Multinumerals Influenced by the Irradiation Illuminances." In Work With Computers: Organizational, Management, Stress and Health Aspects, eds. M. J. Smith and G. Salvendy, pp. 542-548. Elsevier Science Publishers B.V., Amsterdam.

Muter, P. and C. Mayson. 1986. "The Role of Graphics in Item Selection from Menus." Behavior and Information Technology 5:89-95.

Nes, F. 1986. "Space, Color, and Typography on Visual Display Terminals." Behavior and Information Technology 5(2):99-118.

Nickerson, R. S. 1986. Using Computers: The Human Factors of Information Systems. The MIT Press, Cambridge, Massachusetts.

Nicol, A. 1990. "Interface for Learning: What Do Good Teachers Know That We Don't?" In The Art of Human-Computer Interface Design, ed. B. Laurel, pp. 113-122. Addison-Wesley, Reading, Massachusetts.

Nielsen, J. 1987. "A User Interface Case Study of the Macintosh." In Cognitive Engineering in the Design of Human-Computer Interaction and Expert Systems. ed. G. Salvendy, pp. 241-248. Elsevier Science Publishers B.V., Amsterdam.

Norman, K. L., and J. P. Chin. 1988. "The Effect of Tree Structure on Search in a Hierarchical Menu Selection System." Behavior and Information Technology 7(1):51-65.

Ogden, W. C., and S. R. Brooks. 1983. "Query Languages for the Casual User: Exploring the Middle Ground between Formal and Natural Languages." In Human Factors in Computing Systems: Proceedings of the CHI '83 Conference, ed. A. Janda. North Holland Publishing Company, Amsterdam.

O'Keefe, R. M. 1989. "The Evaluation of Decision-Aiding Systems: Guidelines and Methods." Information and Management 17:217-226.

Olsen, L. A., and T. N. Huckin. 1991. Technical Writing and Professional Communication (2nd ed.). McGraw-Hill, New York.

Olson, J. M. 1987. "Color and the Computer in Cartography." In Color and the Computer, ed. H. J. Durrett, pp. 205-219. Academic Press, Inc., San Diego, California.

O'Malley, C., P. Smolensky, L. Bannon, E. Conway, J. Graham, J. Sokolov, and M. L. Montry. 1983. "A Proposal for User-Centered System Documentation." In HMI Project. User-Centered System Design: Papers for the CHI '83 Conference on Human Factors in Computer Systems, p. 4-8. DTIC No. AD 136131. (Technical Report). Office of Naval Research/Personnel and Training, Arlington, Virginia.

Open Software Foundation (OSF). 1990. OSF/Motif Style Guide. Prentice Hall, Englewood Cliffs, New Jersey.

Osborn, P. B., and W. H. Zickefoose. 1990. "Building Expert Systems From the Ground Up." AI Expert, pp. 28-35.

OSF - see Open Software Foundation

Otte, F. H. 1982. "Consistent User Interface." In Human Factors and Interactive Computer Systems, pp. 261-275. Ablex, Norwood, New Jersey.

Owen, D. 1987. "Direct Manipulation and Procedural Reasoning." In Cognitive Engineering in the Design of Human-Computer Interaction and Expert Systems, ed. G. Salvendy, pp. 349-356. Elsevier Science Publishers, B.V., Amsterdam.

Paap, K. R., and R. J. Roske-Hofstrand. 1986. "The Optimal Number of Menu Options Per Panel." Human Factors 28(4):377-385.

Paap, K. R., and R. J. Roske-Hofstrand. 1988. "Design of Menus." In Handbook of Human-Computer Interaction, ed. M. Helander, Elsevier Science Publishers, B.V., Amsterdam.

Parker, S. P., ed. 1989. McGraw-Hill Dictionary of Scientific and Technical Terms. 4th ed. McGraw-Hill, New York.

Parkinson, S. R., M. D. Hill, N. Sisson, and C. Viera. 1988. "Effects of Breadth, Depth, and Number of Responses on Computer Menu Search Performance." International Journal of Man-Machine Studies 28:683-692.

Parrish, R. N., J. L. Gates, and S. J. Munger. 1981. Design Guidelines and Criteria for User/Operator Transactions with Battlefield Automated Systems Volume IV: Provisional Guidelines and Criteria. Technical Report 537, U.S. Army Research Institute for the Behavioral and Social Sciences, Alexandria, Virginia.

Parsaye, K., M. Chignell, K. Setrag, and H. Wong. 1990. "Intelligent Databases." AI Expert, Mar. 1990, pp. 38-47, Miller Freeman, Inc.

Patricia Seybold's Office Computing Group. 1989. "Patricia Seybold's Office Computing Report." April 1989. 12(4):1(9).

Pavard, B. 1987. "Design of Graphic Dialogue Without Syntactic Constraints." In Cognitive Engineering in the Design of Human-Computer Interaction and Expert Systems, ed. G. Salvendy, pp. 349-356. Elsevier Science Publishers B.V., Amsterdam.

Payne, S. J. 1983. "Readability of Liquid Crystal Displays: A Response Surface." Human Factors Society, Inc, 25(2):185-190. Santa Monica, California.

Petrun, C. J., W. Hernon, and M. MacDonald. 1985. "An Examination of the Relative Legibility of Text on One Line Vacuum Florescent and Liquid Crystal Displays." In Proceedings of the Human Factors Society - 29th Annual Meeting, pp.1122-1124. HFS, Santa Monica, California.

Pew, R. W. 1988. "Human Factors Issues in Expert Systems." In Handbook of Human-Computer Interaction, ed. M. Helander, pp. 931-940. Elsevier Science Publishers B.V., Amsterdam.

Pleshko, P. 1991. "AC Electroluminescent Display Technology: Challenges and Potential." Proceedings of Society for Information Display, 32(2):106-108. SID, New York.

Rancourt, J., W. Grenawalt, M. W. Hunter, and H. L. Snyder. 1986. "Quantitative Evaluation of the Effect of an Antireflection Filter." In Society for Information Display 86 Digest, pp. 420-423. SID, New York.

Raum, H. G. 1988. Design and Implementation of an Expert User Interface for the Computer Aided Prototyping System. Thesis for U.S. Naval Postgraduate School, Monterey, California.

Raymond, E. S., ed. 1991. The New Hacker's Dictionary. The MIT Press, Cambridge, Massachusetts.

Reger, J. J., H. L. Snyder, W. W. Farley. 1989. "Legibility of Emissive and Non-Emissive Displays Under Florescent and Daylight Illumination." In Society for Information Display 89 Digest, pp. 364-367. SID, New York.

Reinhart, W. F. 1990. "Effects of Depth Cues on Depth Judgments Using a Field-Sequential Stereoscopic CRT Display." Unpublished Doctoral Dissertation, Virginia Polytechnic Institute and State University, Blacksburg, Virginia.

Relles, N. 1981. "A User Interface for On-line Assistance." In Proceedings of the Fifth International Conference on Software Engineering, pp. 400-408. Institute for Electrical and Electronics Engineers (IEEE), New York.

Relles, N., and L. A. Price. 1981. "A User Interface for On-line Assistance." In Proceedings of the Fifth International Conference on Software Engineering. Institute for Electrical and Electronics Engineers (IEEE), New York.

Riedel, S. L. 1988. "User Acceptance and Field Implementation of Decision Support Systems." Research Report 1477. U.S. Army Research Institute Field Office, Fort Leavenworth, Kansas.

Ripley, G. D. 1989. "DVI--A Distal Multimedia Technology". Communications of the ACM 32:811-822.

Roach, J., R. Hartson, R. W. Ehrich, T. Yunten, and D. H. Johnson. 1982. "DMS: A Comprehensive System for Managing Human-Computer Dialogue." In Human Factors in Computer Systems: Proceedings March 15-17, 1982, Gaithersburg, Maryland.

Rockley, A. 1987. "Online Documentation: From Proposal to Finished Product." In Proceedings of 34th International Technical Communication Conference, ATA 58-61. Society for Technical Communication, Washington, D.C.

Roth, J.L., J. A. Fitzpatrick, R. E. Warm, and J. L. Ditzian. 1988. Implementing Embedded Training (ET): Volume 5 of 10: Designing the ET Component. ARI Research Product No. 88-28, NTIS No. AD-A205 697. U.S. Army Research Institute for the Behavioral and Social Sciences, Alexandria, Virginia.

Rupp, B. A. 1981. "Visual Display Standards: A Review of Issues." Proceedings of Society for Information Display, 22(1):63-72. SID, New York.

Salvendy, G. 1987. Cognitive Engineering in the Design of Human-Computer Interaction and Expert Systems. Elsevier Science Publishers B.V., Amsterdam.

Schauer, U. 1983. "The Integrated Data Analysis and Management System - A Generator for Enduser Systems." Enduser Systems and Their Human Factors eds. A. Blaser and M. Zoeppriz, pp. 30-61. Springer-Verlag, Berlin.

Schmitz, J. D., G. D. Armstrong, and J. D. Little. 1990. "CoverStory - Automated News Finding in Marketing." Interfaces 20(6):29-38.

Schur, S. 1988. "The Intelligent Database." AI Expert, Jan. 1988, pp. 26-34, Miller Freeman, Inc.

Schwartz, J. P. 1983. "Lack of Guidance for Decision Aid Interface Design." SIGCHI Bulletin 15:13-17.

Sellen, A., and A. Nicol. 1990. "Building User-Centered On-line Help." In The Art of Human-Computer Interface Design, ed. B. Laurel, pp. 143-153. Addison-Wesley. Reading, Massachusetts.

Shaw, B., and M. McCauley. 1985. Personal Computer Dialogue: A Human Engineering Data Base Supplement. Technical Report No. AFAMRL-TR-85-013. U.S.A.F. Aerospace Medical Research Laboratory, San Antonio, Texas.

Shields, S. E., and W. P. Bleha. 1991. "Liquid Crystal Light Valves for Projection Displays." In Society of Photoptic Instrumentation Engineers, 1455:225-236. SPIE Press, Bellingham, Washington.

Shneiderman, B. 1978. "Improving the Human Factors Aspect of Database Interactions." ACM Transactions on Database Systems 3(4):417-439.

Shneiderman, B. 1982. "Designing Computer System Messages." Communications of the ACM, 25(9):604-605.

Shneiderman, B. 1984. "The Future of Interactive Systems and the Emergence of Direct Manipulation." In Human Factors and Interactive Computer Systems, ed. Y. Vassiliou, pp. 1-27. Ablex Publishing Corporation, Norwood, New Jersey.

Shneiderman, B. 1987. Designing the User Interface: Strategies for Effective Human-Computer Interaction. Addison-Wesley, Reading, Massachusetts.

Shneiderman, B. 1988. "We Can Design Better User Interfaces: A Review of Human-Computer Interaction Styles." Ergonomics 31(5):699-710.

Shneiderman, B. 1991. "Visual User Interfaces for Information Exploration", made available at the TAE Ninth User's Conference, New Carrollton, Maryland. CAR-TR-577, CS-TR-2748, Human-Computer Interaction Laboratory and Department of Computer Sciences, University of Maryland, College Park, Maryland.

Shurtleff, D. A., W. F. Wuersch, and J. G. Rogers. 1981. "How to Make Large Screen Displays Legible." In Proceedings of the Human Factors Society - 25th Annual Meeting, pp. 149-153. HFS, Santa Monica, California.

Shurtleff, D. A., W. F. Wuersch, and J. G. Rogers. 1982. "Applications of Large-Screen Display Legibility Criteria." In Society for Information Display 82 Digest, pp. 202-203. SID, New York.

Sidorsky, R. 1984. Design Guidelines for User Transactions with Battlefield Automated Systems: Prototype for a Handbook. DTIC AD-A153 231, U.S. Army Research Institute for the Behavioral and Social Sciences, Alexandria, Virginia.

Silverstein, L. D., R. W. Monty, J. W. Huff, and K. L. Frost. 1987. Image Quality and Visual Simulation of Color Matrix Displays. Technical Paper No. 871789, The Engineering Society for Advancing Mobility Land Sea Air and Space, Long Beach, California.

Sisson, N., S. R. Parkinson, and K. Snowberry. 1986. "Consideration of Menu Structure and Communication Rate for the Design of Computer Menu Displays." International Journal of Man-Machine Studies 25:479-489.

Slominski, S.E., and I. R. Young. 1988. A User Friendly Design of an Interactive Prototype for the Maintenance and Monitoring of Civilian Training Records. Unpublished Master's Thesis, Naval Postgraduate School, Monterey, California.

Smith, S. L., and J. N. Mosier. 1984. A Design Evaluation Checklist for User Computer Interface Software. MITRE Corporation, Bedford, Massachusetts.

Smith, S. L., and J. N. Mosier. 1986. Guidelines for Designing User Interface Software. MITRE Corporation, Bedford, Massachusetts.

Snyder, H. L. 1980. "Human Visual Performance and Flat Panel Display Image Quality." HFL-80-1/ONR-80-1, Human Factors Laboratory, Virginia Polytechnic Institute and State University, Blacksburg, Virginia.

Snyder, H. L. 1984. "Ergonomics Database for Visual Displays and VDUs." In Ergonomic Data for Equipment Design, ed. H. Schmidtke, pp. 219-234. Plenum Press, New York.

Snyder, H. L. 1987. "The ANSI Human Factors Standard for Visual Display Terminal Workstation: A Process and Progress Report (Invited Address)." In Society Information Display 87 Digest, pp. 14-17. SID, New York.

Snyder, H. L. 1988. "Image Quality." In Handbook of Human-Computer Interaction, ed. M. Helander, pp. 437-474. Elsevier Science Publishers B.V., Amsterdam.

Sormunen, E. 1987. "A Knowledge-Based Intermediary System for Information Retrieval," in Knowledge Engineering Expert Systems and Information Retrieval, ed. I. Wormell, pp. 59-73. Taylor Graham, London.

Steele, C. A., J. Harrold, J. P. Stanton, and R. Daley. 1987. "Performance of Laser-Addressed Liquid Crystal Map Overlay Display." In Society of Photoptic Instrumentation Engineers, 760:70-73. SPIE Press, Bellingham, Washington

Suntola, T. 1989. "VLSI and Computer Peripherals: Thin Film EL-Display." Proceedings of the Institution of Electrical and Electronics Engineers, pp.2-32/2-35. IEEE Computer Society Press, Washington, D.C.

Tannas, L. E., ed. 1985. Flat-Panel Displays and CRTs. Van Nostrand Reinhold, New York.

Tannenbaum, A. 1990. "Installing AI Tools Into Corporate Environments." AI Expert, pp. 54-59.

Tenopir, C., and G. Lundeen. 1988. Managing Your Information. Neal-Schuman Publishers, New York.

Thierauf, R. J. 1988. User-Oriented Decision Support Systems. Prentice Hall, Englewood Cliffs, New Jersey.

Thomas, J. C. 1983. "Psychological Issues in Data Base Management." In Designing for Human-Computer Communications, eds. M. E. Sime and M. J. Coombs, pp. 169-184. Academic Press, London.

Thomas, J.C., and J. M. Carroll. 1981. "Human Factors in Communication." IBM Systems Journal 20(2):236-263.

Thorell, L. G., and W. J. Smith. 1990. Using Computer Color Effectively: An Illustrated Reference. Prentice Hall, Englewood Cliffs, New Jersey.

TRW. 1990a. The AWIS Common User/System Interface Style Guide. AWIS Program Documentation.

TRW. 1990b. UTACCS, Soldier-Machine Development Standards. UTACCS Program Documentation.

Tullis, T. S. 1988. "Screen Design." In Handbook of Human-Computer Interaction, ed. M. Helander, pp. 377-411. Elsevier Science Publishers B.V., Amsterdam.

U.S. Department of the Army. 1984. Human Engineering Guide to Equipment Design. FM 21-26 (sec 11), eds. H. P. Van Cott and R. G. Kinkade. John Wiley and Sons, New York.

U.S. Department of the Army. 1985a. Authorized Abbreviations and Brevity Codes. AR310-50, Army UPDATE Publications, Washington, D.C.

U.S. Department of the Army. 1985b. Operational Terms and Symbols. FM 101-5-1, U.S. Army Combined Arms Center, Fort Leavenworth, Kansas.

U.S. Department of the Army. 1987. Map Reading and Land Navigation. FM 21-26, Washington, DC.

U.S. Department of Defense (DoD). 1989a. Human Engineering Design Criteria for Military Systems, Equipment, and Facilities. MIL-STD-1472D, U.S. Army Missile Command.

U.S. Department of Defense (DoD). 1989b. Human Engineering Guidelines for Management Information Systems. DOD-HDBK-761A, DoD, Washington, D.C.

U.S. Department of Defense (DoD). 1990. Military Training Programs. MIL-STD 1379D, Naval Sea Systems Command, SEA 55Z3.

U.S. Department of Defense (DoD). 1991. Department of Defense Human Computer Interface Style Guide (Draft), Common Operating Environment Working Group, Washington, D.C.

Ullman, J. D. 1988. Principles of Database and Knowledge-Base Systems. Computer Science Press, Rockville, Maryland.

UNIX System Laboratories, Inc. 1991. Open Look Graphical User Interface User's Guide. Prentice Hall, Inc., Englewood Cliffs, New Jersey.

Urban, C. D. 1990. "Design and Evaluation of a Tactical Decision Aid." Proceedings IEEE International Conference on Systems, Man, and Cybernetics. DTIC No. AD-A232 001, IEEE, New York.

Vance, D. W. 1987. "Image Distortions in Video Projection." In Large Screen Projection Displays, 760:54-59. SPIE Press, Bellingham, Washington.

Van Cott, H. P., and R. G. Kinkade, eds. 1984. Human Engineering Guide to Equipment Design. John Wiley and Sons, New York.

Vassiliou, Y., ed. 1984. Human Factors and Interactive Computer Systems. Ablex Publishing Corporation, Norwood, New Jersey.

Vasta, J. A. 1985. Understanding Data Base Management Systems. Wadsworth Publishing Company, Belmont, California.

Veith, R. H. 1988. Visual Information Systems: The Power of Graphics and Video. G.K. Hall & Co., Boston.

Veron, H. 1988. A Resolution Measurement Technique for Large Screen Displays. Technical Report No. M88-59, The MITRE Corp. Bedford, Massachusetts.

Veron, H., D. A. Southard, J. R. Leger, and J. L. Conway. 1990. 3D Displays for Battle Management. Technical Report No. AD-A223 142, Defense Technical Information Center, Alexandria, Virginia.

Verplank, W. L. 1988. "Graphic Challenges in Designing Object-Oriented User Interfaces." In Handbook of Human-Computer Interaction, ed. M. Helander, pp. 365-375. Elsevier Science Publishers B.V., Amsterdam.

Waern, Y., and C. Rollenhagen. 1983. "Reading Text From Visual Display Units (VDUs)." In International Journal of Man-Machine Studies 18, Academic Press, London.

Walker, J. H. 1987. "Issues and Strategies for Online Documentation." IEEE Transactions on Professional Communication, PC-30 (4):235-248.

Walrath, J. D. 1989. "Aiding the Decision-Maker: Perceptual and Cognitive Issues at the Human-Machine Interface." Technical Note 15-89. U.S. Army Human Engineering Laboratory (DTIC No. AD-A217 862), Aberdeen Proving Ground, Maryland.

Wehrer, W., and E. E. Mitchamore. 1992. "Technology/Marketing Case Study: Plug-and-play IR Touch Screens." Information Display, pp. 10-12. SID, New York.

Weingaertner, S. T., and A. H. Levis. 1988. "Evaluation of Decision Aiding in Submarine Emergency Decision-Making." IFAC Conference on the Analysis, Design and Evaluation of Man-Machine Systems, Ouhu, Finland, pp. 195-201.

Wenger, E. 1987. Artificial Intelligence and Tutoring Systems. Morgan Kaufman Publishers, Los Altos, California.

Wexelblat, R. L. 1989. "On Interface Requirements for Expert Systems." AI Magazine 10:66-78.

Williams, K.E., C. J. Hamel, and L. B. Shrestha. 1987a. An Evaluation of Characteristics Contributing Towards Ease of User-Computer Interface in a Computer-Aided Instruction Exercise. Technical Report TR87-030, Naval Training Systems Center, Orlando, Florida.

Williams, K.E., C. J. Hamel, and L. B. Shrestha. 1987b. CAI Evaluation Handbook: Guidelines for the User Interface Design for Computer-Aided Instruction. Technical Report No. TR87-033, Naval Training Systems Center, Orlando, Florida.

Williams, R. D. 1990. "Volume 3D Display Technology." In Stereographics, 17th International Conference on Computer Graphics and Interactive Techniques, *Association for Computing Machinery, Special Interest* Group on Graphics, Dallas Texas.

Wilson, M., P. Barnard, and A. MacLean. 1990. An Investigation of the Learning of a Computer System. In Cognitive Ergonomics: Understanding, Learning and Designing Human-Computer Interaction, ed. P. Falzon, pp. 151-172. Academic Press, London.

Woodson, W. E., B. Tillman, and P. Tillman. 1992. Human Factors Design Handbook: Information and Guidelines for the Design of Systems, Facilities, Equipment, and Products for Human Use, Second Edition. McGraw-Hill, Inc., New York.

Wormell, I., ed. 1987. Knowledge Engineering Expert Systems and Information Retrieval. Taylor Graham, London.

Wyatt, A. L. 1990. Computer Professional's Dictionary. Osborne McGraw-Hill, Berkeley, California.

Yamazaki, T., K. Kamijo, and S. Fukuzumi. 1990. "Quantitative Evaluation of Visual Fatigue Encountered in Viewing Stereoscopic 3D Displays: Near-Point Distance and Visual Evoked Potential Study." Proceedings of Society for Information Displays, 31(3):245-247. SID, New York.

Yeh, Y., and L. D. Silverstein. 1989. "Using Electronics Stereoscopic Color Displays: Limits of Vision and Depth Discrimination." Three-Dimensional Visualization and Display Technologies, Proceedings of SPIE International Society for Optical Engineering, Vol. 1083, pp. 196-204. SPIE Press, Bellingham, Washington.

Yeh, Y., and L. D. Silverstein. 1991. "Human Factors for Stereoscopic Color Displays." In Society for Information Display 91 Digest, pp. 826-829. SID, New York.

Zachary, W. W. 1988. "Decision Support Systems: Designing to Extend the Cognitive Limits." In Handbook of Human-Computer Interaction, ed. M. Helander, pp. 997-1030. Elsevier Science Publishers B.V., Amsterdam.

Ziegler, J. E., and K. P. Fähnrich. 1988. "Direct Manipulation." In Handbook of Human-Computer Interaction, ed. M. Helander, pp. 123-134. Elsevier Science Publishers B.V., Amsterdam.

Zmud, R. W. 1978. "Concepts, Theories and Techniques: An Empirical Investigation of the Dimension of the Concept of Information." Decision Sciences 9:187-195.

APPENDIX B - GLOSSARY

AC: See alternating current.

accelerator key: special key or key combination that performs the same action as a menu selection.

Ada: high-level computer programming language developed by the Department of Defense. Ada is used as the standard programming language for DoD. It is used for real-time processing, is modular in nature, and includes object-oriented features.

alternating current (AC): electrical current that reverses its direction at regularly recurring intervals.

application: classification of computer programs designed to perform specific tasks, such as word processing, database management, or graphics.

applications menu: list of options within an application.

automated tools: *software performing* a sequence of operations to assist the user in achieving a goal (e.g., within graphics software, functions that align objects, smooth curves, or draw circles).

backlighting: lit from the back. When referring to a monitor, light passes through the display screen from the back in order to illuminate screen images.

base-level functions: initial or basic functions.

batch processing: processing data or the accomplishment of jobs accumulated in advance in such a manner that each accumulation thus formed is processed or accomplished in the same computer run.

baud: measure of the transmission speed capability of a communications line or system. In a sequence of binary signals, the rate of one baud equals one bit per second.

bit-mapped display: display in which every picture element (pixel) of the screen can be referenced individually.

bookmarking: method of tagging items of interest to the user for easy referral later. Allows the user to customize the application.

Boolean logic: logical expression that uses Boolean operators such as AND, OR, NOT, XOR, NOR, and NAND to create a statement that, when resolved, is either true or false.

Boolean operators: a keyword in programming that causes two values to be combined in a logical fashion.

branching menu: menu which, if selected, brings up another menu.

bring-to-front: process of moving a window to the foreground.

Candela (cd): unit of luminous intensity expressed in Candela per square meter (cd/m^2). One cd is equal to 0.29 footLambert.

Cathode ray tube: electronic vacuum tube that focuses electrons energizing phosphors on a screen, creating a visible display. The typical computer monitor uses this type of display technology.

cd: See candela.

central processor: portion of the computer that controls execution of applications.

character: single letter, digit, or symbol--synonymous with byte.

character string: series of alphanumeric characters, the contents of which are treated as though they were text.

COBOL: acronym for Common Business-Oriented Language. COBOL is a computer programming language used extensively in mainframes and minicomputers for business applications.

command: entry that instructs the computer to effect a specific action.

command entry: informing the computer that a specific command should be effected.

command language: limited programming language used strictly for executing a series of commands.

command stacking: allows the user to key a sequence of commands as a single "stacked" command entry.

compatible letters: letters easily associated with the function requested, for example, "P" for print, "Q" for quit.

context-sensitive: computer action or response directly related to the cursor position or specific point in the software, for example, a help function that displays information about the specific data entry field in which the cursor was located when help was called.

control action: actions which must be effected to control a window or other graphics object or its contents.

control entry: input action by the computer user that changes some aspect of the appearance or function of the application.

control lockout: processing delay that results in pacing the capability to enter sequences of control commands.

courseware: another name for educational or training materials and software.

crosstalk: optical crosstalk, or bleeding, occurs when the light from the incorrect video image gets through. When referring to stereoscopic images, the right eye's image is visible to the left eye or vice versa.

CRT: See Cathode Ray Tube.

cursor: visual mechanism to mark, on-screen, where current input or output is to happen.

data entry: series of keystrokes used to input information into the computer.

data entry field: space (number of characters and/or digits) allowed for data entry.

data field: location in a file or database that contains a specific type of information.

database: structured or organized collection of information, which may be accessed by the computer.

database management system: computer application program that accesses or manipulates the database.

DC: See direct current.

default: command which is automatically executed if none is specifically indicated.

default value: value of a variable in lieu of a specifically indicated value.

defeated: option that cannot be selected due to another selected option's use.

delimiter: symbols such as commas, spaces, or parentheses which mark the boundaries of a specific block of information.

designate: process of selecting and displaying the current or active window with visual cues.

destructive entries: any entry which will destroy or overwrite information.

dialog: structured series of interchanges between a user and a computer terminal. Dialogs can be initiated by the computer or the user. Interactive dialog consists of an action by the user followed by a response from the computer or vice versa.

dialog box: screen display box containing a message requesting additional information from the user.

direct current (DC): electrical current that flows in one direction only and is substantially constant in value.

direct manipulation: method of data organization (typically involving extensive windowing and iconization) in which the user can select specific displays of information and move them about to facilitate interaction with an application. A system of interaction in which the user's actions directly affect software operations.

display frame: window or page.

display parallax: when used in discussing touch screen technology, display parallax is the apparent displacement of an object viewed on a curved CRT screen and seen through a flat touch interactive display.

display screen: screen of a multipage file.

double keying: each character of the data item does not have an appropriately labeled key and therefore requires more than one keystroke for entry.

dual activation: where two key are used simultaneously to input a command.

dynamic depth displays: stereoscopic displays that are designed to change (move) images during viewing.

electroluminescence: luminescence produced by electrical excitation of phosphate in powder or film form.

electronic mail: communication, processed through a network, from one workstation to another.

end user: person who ultimately uses the computer application or output.

error management: various options within an application that allow the user to eliminate the effects of commands executed accidentally or unwisely.

expand: ability to resize objects to produce better organization of on-screen material, usually a graphic or a window.

feedback: visual acknowledgement that the computer is executing the command or that the command was executed.

field: addressable data location.

file: any specifically identified collection of information stored in the computer.

fc: See footcandle

Fl: See footLambert.

footcandle (fc): unit of measurement of illumination. The amount of light emitted by a standard candle (1 cd) measured one foot away from the candle equals one footcandle.

footLambert: unit of measure of intensity of reflected or emitted light (luminance). The average luminescence of any reflecting surface in footLamberts is the product of the illumination in footcandles by the luminous reflectance of the surface.

frame: single display image or screen.

function key, fixed and variable: key which, when depressed, effects a specific action. It can either be a single, predefined function (fixed), or vary according to the system mode or level within an interactive dialog.

form filling: method of interaction in which the user enters a series of commands or data items in predefined fields. These fields may be mandatory or optional.

FORTRAN: acronym for FORmula TRANslator, which is a high level computer language used extensively in scientific and engineering applications.

freeze: See Option - PAUSE.

graphical interaction: transactions between the user and computer-generated graphical representations of objects (screens, menus, buttons, etc.).

graphical user interface (GUI): system design that allows the user to effect commands, enter into transaction sequences, and receive displayed information through graphical representations of objects (menus, screens, buttons, etc.).

hard copy: printed copy of machine output in a visibly readable form. For example, printed reports, listings, documents, summaries.

hardware architecture: assemblage of a computer's internal components and its attached peripheral, device which determine its capabilities and its limitations.

hatching: graphical pattern characterized by 45 and 135 degree diagonal lines which cross the patterned area.

help screen: separate window that offers advice and information on how to overcome a specific problem and/or to better interact with the computer.

HFE: See Human Factors Engineering.

Human Factors Engineering (HFE): approach that makes use of scientific facts in the design of items (i.e. computer systems, software, etc.) to produce effective human-machine integration and utilization.

hierarchical menu: method of organizing menus in layers. The secondary or tertiary menus are stored within a primary menu.

high level language: programming language that does nor reflect the structure of any one computer or class of computers.

high resolution: screen display within an extremely fine visual reproduction of detail.

highlight: visual method to call attention to a specific piece of text or a graphic through differentiating it from surrounding texts or graphics. This is usually accomplished using contrasting colors or reverse video.

hook: selecting a corner of a window or icon in order to move or resize it.

Human-Computer Interface: hardware and software allowing information exchange between the user and the computer.

icon: graphical representation of an object, concept, or message used by a computer system to represent items such as files, documents, programs, and disk drives.

iconify: process that changes the text representation of an object, concept, or message into an icon.

iconification: process of iconifying.

illuminance: measure of the quantity (density) of light reaching an object or surface. Measured in footcandles.

Image Formation Time (IFT): measurement of the time required to update screen image displays.

Infrared (IR): radiation outside the visible light range on the red side (wavelength 0.75 to 0.8 micrometers).

interactive control: attribute describing the ability of a program and a user to interface with each other during program execution.

interactive dialogue: See dialog.

interactive procedures: methods by which a user interacts with a computer and the computer with the user.

interface: interconnection and interrelationships between two devices, two applications, or the user and an application or device.

interlock: mechanism to connect two or more processes within a computing system to ensure that no one part of a hardware or software system can be operated independently.

interocular: perceptual coordination between the eyes.

IFT: See image formation time.

IR: See infrared.

jump-ahead: capability of moving ahead during a step-wise process to allow quicker performance of an operation; useful for experienced computer users.

justification: alignment of text on a display or a printed page. Left justification means that the left margin is even.

keyword: special word in a programming language that tells the computer which operation to perform.

Lambert: See footLambert.

landscape: screen display or printing orientation parallel to the wide side of the paper.

LCD: See liquid crystal display.

LCSS: See liquid crystal stereoscopic shutter.

left-justified: See Justification.

Liquid crystal display (LCD): display operated by polarizing light in which the nonactive segment reflects incident light and thus appears invisible against its background.

liquid crystal stereoscopic shutter (LCSS): type of display that utilizes liquid crystal shutters, one for each eye synchronized to alternate fields of the display, and representing one of the two images necessary to achieve the third dimension.

lockout: condition of the application locking the keyboard (i.e., not accepting commands from it) while the application is effecting a command.

log on: process of gaining access to the system, usually involving a password and a recognition of the specific user by the computer.

logarithm: the exponent that indicates the power to which a number has been raised to produce the given number.
$N = 10^n \qquad \log_{10} N = n$

luminance: amount of light per unit area reflected from or emitted by a surface. Measured in footcandles.

lux: standard measure of illuminance. One lux is one lumen per square meter.

macro: executable file that stores a series of commands and keystrokes to be used later.

Manpower and Personnel Integration (MANPRINT): an Army program that addresses concerns with manpower, personnel, training, human factors, safety, and health hazards.

MANPRINT: acronym; MANpower and PeRsonnel INTegration.

masking: partial or complete obscuring of one item by another.

memory: place in the computer in which information is stored.

menu: list of options available within a software application.

menu bar: the horizontal menu usually at the top of the screen which, contains menu titles.

metaphor: system-level analogy used for the grouping of processes and/or procedures. Usually associated with icons based on the analogy. As, for example, a desk top metaphor where icons represent office equipment or operations.

minimize: procedure to make the window as small as it can be without being closed; this is usually done through iconization.

mnemonic: word or code symbolic of another word, code, or function.

mode: status of the screen or program process.

Modulation Transfer Function (MTF): a parameter using spatial frequency responses to characterize a screen display. The spatial frequency is stated in lines (line pairs) or minimum/maximum intensity pairs per unit distance. The MTF is used as a performance measurement of many optical systems.

Motif: user interface design approach based upon the "look" and "feel" presented in the OSF/Motif style guide. Motif is marketed by the Open Software Foundation.

MTF: See modulation transfer function.

multifunction keying: interface design where computer keys may perform multiple functions with the use of a combination of keystrokes.

multiwindow: simultaneous display of several windows on the computer screen.

natural language: programming language paradigm exemplified by using English-like commands and syntax to issue commands; interactions in the vernacular of the user.

navigation: manner in which the user moves through the menu structure.

NATO Forces: personnel in the military forces of member nations of the North Atlantic Treaty Organization (NATO).

nit: See normalized intensity.

normalized intensity (nit): metric unit of measure of luminous intensity. A nit is equal to one candela per square meter (cd/m2) or 0.29 footLambert.

null: empty; nothing. A null set contains no elements.

one to many mapping: an icon which represents a category of possibilities within an option is a one to many mapping.

one to one mapping: an icon which represents a single, specific function is a one to one mapping.

Open Software Foundation (OSF): consortium of computer hardware and software manufacturers whose membership includes over seventy of the computer industries leading companies.

open window map: a map (graphic display) that shows windows that are open and how they relate to each other.

open: procedure to cause a window to be displayed from an icon or menu option so a document, directory or file can be viewed.

option: command that may be selected to access a specific function of an application.

option - BACKUP: option that will display the last transaction or the process of saving information to non-volatile memory.

option - CANCEL: command that allows the user to have the computer disregard a previous command.

option - CONFIRM: explicit warning of any possible data loss.

option - CONTINUE: option that resumes a transaction sequence which has been stopped by a PAUSE.

option - GOBACK: option that will display the last transaction. See also BACKUP.

option - PAUSE: option that temporarily causes a transaction sequence to stop running. Use the CONTINUE option to resume after pausing.

option - RESTART: option that will cancel any entries that have been made in a transaction sequence and return to the beginning of a sequence.

option - REVIEW: option that returns to the first display of a transaction sequence, allowing the user to review the transaction and make necessary changes.

option - SUSPEND: option that allows a user to leave the application, then, when he returns, resume at the same point he left off.

option - UNDO: option that immediately reverses any action.

option code: codes associated with the available choices.

OSF: See Open Software Foundation.

output: information the computer displays in response to the user's actions.

overarching guidelines: dominant or all embracing guidelines.

overlapping: windowing system in which one window covers a portion of another.

overlay: printing or drawing on a transparent or semi-transparent medium on the same scale as a map, chart, etc., to show details not appearing or requiring special emphasis on the original.

paging: scrolling through material one page at a time.

paired opposites: set of opposite functions such as up and down, top and bottom.

pan: process to change the displayed region (often of a map) in a regular and smooth manner.

parallax: apparent displacement of an object as seen from two different points not on a straight line with the object.

parameter: quantity or constant whose value varies with the circumstances of the application.

piezoelectric: electric polarity due to pressure especially in a crystalline substance.

pixel: contraction for picture element. A pixel is a single dot on a display screen.

pixel matrix: arrangement of screen dots (pixels) to form text or graphic displays.

pop-up menu: lists of options that appear on the display screen in the form of a window.

portrait: screen or printing orientation parallel to the narrow side of the paper.

predictive modelling: use of a model to predict the actual response of a system or process.

preformatted: screen structure prepared for the user.

presentation graphics: pictorial representations of the relationships between variables (graphs and charts) or representations of systems (diagrams, schematics and graphical renditions).

primitive: code that defines a specific elementary shape, form or color.

programming language: artificial language establishe... r expressing computer programs.

prompt: text or graphic display that indicates the start point for user-generated actions. This term is also used for software generated instructions for process confirmation.

pull-down menu: lists of options attached to a selection on a menu bar.

push-to-back: process of moving a window to the background.

query language: specialized type of command language to elicit information from the computer system.

real time: absence of delay, except for the time required for transmission.

real-time control system: systems capable of responding to external events with negligible delays.

resize: procedure to change the size of a window or graphic.

resize border: border of a window that, if selected, will allow the user to resize the window.

resolution: density of picture display elements of the screen; degree of detail with which an image is displayed or printed.

retrieve: procedure required to display stored information for purposes of viewing and manipulation.

right-justified: See justification.

SAW: See surface acoustic wave.

scroll: method used to move the contents of a window or list in a dialogue box using the scroll bar or scroll arrows.

scroll bar: rectangular bar that may be along the right or bottom of a window. Clicking or dragging in the scroll bar causes the view of the document to change.

secondary coding: providing more than one method for coding displayed information. For example, in coding a particular item with red color, the use of the symbol "R" would provide secondary method for conveying the information when color was not available.

semantics: relationship of characters or groups of characters to their meanings, independent of the manner of their interpretation and use.

sensitivity analysis: study that shows the response of a system to varying conditions. For example, "How sensitive is the system to increased workload?"

sequence control: prescribed control over the order of function performed by the computer; this impacts the way in which a user interacts with the application.

size coding: variations in the size of displayed alphanumerics and symbols. Such coding can be used for categorization.

slider: part of the scroll bar which indicates what part of the file contained in a window is being viewed.

soft keys: visual representation of key functions on the display screen. This is usually associated with software controlled function key capabilities.

specular reflector: Reflecting light in a diffuse manner.

stacked command: single command composed of multiple commands that must be executed individually.

stereopsis: phenomenon of simultaneous vision with two eyes in which there is a vivid perception of distance of objects from the viewer (three dimensional or stereoscopic vision).

stereoscopic: method of seeing objects in three dimensions.

stroke width: width of the line used to create a displayed character.

subordinate window: a window that is opened from and controlled by another window.

subtend: opposite in position.

summary symbols: symbol which categorizes the information portrayed by a group of symbols.

supraordinate window: higher level window, usually the window from which subordinated options or tasks are controlled.

surface acoustic wave (SAW): when used in the context of touch screen technology, an approach that uses ultrasonic sound "beams" transmitted from two perpendicular sides of a display frame.

sw: See stroke width.

system level menu: list of which applications are available for utilization.

system response time: amount of time that elapses between a command being given and its being executed by the computer.

text editor: application that allows text to be created or modified.

text-based systems: method of organization in which the primary form of interaction between the system and user is through text rather than through graphical or voice interaction.

three-dimensional: relating to the three physical dimensions (height, width, depth). Giving the effect of depth or varying distances.

TID: See touch interactive display.

tiling: windowing approach in which multiple windows do not overlap, rather, all lie on the same plane.

theater-type displays: display screens suitable for large group presentations as used in a movie theater or auditorium.

title banner: horizontal bar at the top of a window which shows the name of the window and allows it to be moved.

touch interactive display (TID): uses a physical device between the user and the display which acts as the input mechanism.

transaction: interaction between a user and a computer in which the user inputs a command to receive a specific result from the computer.

transaction sequence: order of transactions required to accomplish the desired results.

transmissivity: measurement of the ability of an image to be transmitted. When used in the context of touch screen technology, refers to the ability of the image to be transmitted through a filter placed in front of a computer screen.

type-ahead: capability of the computer to receive commands faster than it can display their results.

UCI: See user-computer interface.

user-callable: able to be requested by the user as desired.

User-computer interface (UCI): hardware and software allowing information exchange between the user and the computer.

user-specified windows: windows whose content has been selected by the user.

variable: quantity that can assume any of a given set of values.

VDT: See video display terminal.

Video Display Terminal (VDT): terminal comprised of a keyboard for data input and a CRT screen for display of the input/output.

widget: basic graphical object which is a component of a user interface component.

window: typically rectangular display which provides a visual means for interaction with an application.

zoom: graphical tool used to magnify a portion of a document for more detailed viewing.

INDEX

A

INDEX

INDEX

INDEX

INDEX

INDEX

INDEX

INDEX

INDEX

INDEX

INDEX

INDEX

INDEX

INDEX

INDEX

INDEX

INDEX

INDEX

INDEX

INDEX

INDEX

G

INDEX

INDEX

INDEX

INDEX

INDEX

INDEX

INDEX

INDEX

INDEX

INDEX

INDEX

INDEX

N

INDEX

INDEX

INDEX

INDEX

INDEX

INDEX

Q

INDEX

INDEX

INDEX

INDEX

INDEX

INDEX

INDEX

INDEX

INDEX

INDEX

U

INDEX

INDEX

INDEX