AFRL-HE-WP-TR-2003-0161

United States Air Force Research Laboratory

Electronic Flight and Technical Manual Design Guidelines

Sarah Swierenga
Jesse Walker
Andrea Snead
Carlton Donahoo
Laurie Quill

University of Dayton Research Institute
Human Factors Group
300 College Park
Dayton, OH 45469

Jill A. Ritter

Air Force Research Laboratory

October 2003

Interim Report for the Period August 2002 to October 2003

Human Effectiveness Directorate
Deployment and Sustainment Division
Logistics Readiness Branch
2698 G Street
Wright-Patterson AFB OH 45433-7604

NOTICES

When US Government drawings, specifications or other data are used for any purpose other than a definitely related Government procurement operation, the Government thereby incurs no responsibility nor any obligation whatsoever, and the fact that the Government may have formulated, furnished, or in any way supplied the said drawings, specifications or other data, is not to be regarded by implication or otherwise, as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

Please do not request copies of this report from the Air Force Research Laboratory. Additional copies may be purchased from:

National Technical Information Service
5285 Port Royal Road
Springfield, VA 22161

Federal Government agencies registered with the Defense Technical Information Center should direct requests for copies of this report to:

Defense Technical Information Center
8725 John J. Kingman Rd., Ste 0944
Ft. Belvoir, VA 22060-6218

DISCLAIMER

This Technical Report is published as received and has not been edited by the Air Force Research Laboratory, Human Effectiveness Directorate.

TECHNICAL REVIEW AND APPROVAL

AFRL-HE-WP-TR-2003-0161

This report has been reviewed by the Office of Public Affairs (PA) and is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nations.

This technical report has been reviewed and is approved for publication.

FOR THE COMMANDER

//signed//
Deputy Chief
Deployment and Sustainment Division
Air Force Research Laboratory

REPORT DOCUMENTATION PAGE	Form Approved OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	October 2003	Interim - August 2002 - October 2003

4. TITLE AND SUBTITLE
Electronic Flight and Technical Manual Design Guidelines

5. FUNDING NUMBERS
C: F33615-99-D-6001
DO: 06
PE: 62202F
PR: 1710
TA: D0
WU: 09

6. AUTHOR(S)
Sarah Swierenga, Jesse Walker, Andrea Snead, Carlton Donahoo, Laurie Quill, Jill A. Ritter

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
University of Dayton Research Institute
Human Factors Group
300 College Park
Dayton, OH 45469

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Air Force Research Laboratory, Human Effectiveness Directorate
Deployment and Sustainment Division
Air Force Materiel Command
Logistics Readiness Branch
Wright-Patterson AFB OH 45433-7604

10. SPONSORING/MONITORING AGENCY REPORT NUMBER
AFRL-HE-WP-TR-2003-0161

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*
The United States military is dedicated to improving processes associated with electronic manuals including generation, dissemination, and utilization of manuals used for reference while working. In this case, electronic manuals refer to both manuals used in flight (electronic flight manuals), and manuals used in support of maintaining aircraft (electronic technical manuals). Among those military organizations working in this area, The United States Air Force (USAF) has been a leader in many of the challenging discussions surrounding the complex issues associated with creating electronic manuals, distributing those manuals to thousands of users, and ensuring that those user populations can effectively navigate through and manipulate the information to support the task at hand. The purpose, therefore, of this document is to provide an up-to-date framework for the display of both electronic flight manuals (EFM) and Interactive Electronic Technical Manuals (IETM) data within a common web browser interface. Structurally, this document will review literature and provide a source of guidance relevant to the display of electronic flight and technical manual information. Each design guideline leverages knowledge gained in the field of user interface design. Guidance is provided first through an overview of the process that should be taken to successfully implement a user interface. Then detailed guidance is provided in terms of why it is important to consider various points and how to address the given issue through constructive design practices.

14. SUBJECT TERMS
Electronic Flight Manuals Electronic Technical Manuals User Interface Design
Web Interface Design Human Factors Display Guidelines

15. NUMBER OF PAGES
52

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	UL

THIS PAGE INTENTIONALLY LEFT BLANK

PREFACE

The research documented in this technical report for the Electronic Flight and Technical Manual Design Guidelines program sponsored by the Air Force Research Laboratory, Human Effectiveness Directorate, Logistics Readiness Branch (AFRL/HESR), Wright-Patterson Air Force Base, OH. The University of Dayton Research Institute performed the work under Delivery Order #06 of the Technology for Readiness and Sustainment (TRS) contract F33615-99-D-6001. Jill A. Ritter (AFRL/HESR) was the program manager for the effort.

Table of Contents

THIS PAGE INTENTIONALLY LEFT BLANK

Introduction

The United States military is dedicated to improving processes associated with electronic manuals including generation, dissemination, and utilization of manuals used for reference while working. In this case, electronic manuals refer to both manuals used in flight (electronic *flight* manuals), and manuals used in support of maintaining aircraft (electronic *technical* manuals). Among those military organizations working in this area, The United States Air Force (USAF) has been a leader in many of the challenging discussions surrounding the complex issues associated with creating electronic manuals, distributing those manuals to thousands of users, and ensuring that those user populations can effectively navigate through and manipulate the information to support the task at hand.

Paper manuals have proven to be instrumental in supporting many aspects of both flight and technical tasks. Since the advent of personal computers, the USAF has been working to convert these paper manuals to electronic formats. In the early 1990's, the USAF participated in creation of a series of Tri-Service Specifications that specified how a database should be created for information contained in technical manuals (United States Government, 1992a), and how the user interface should be created for display of technical manuals (United States Government, 1992b). These documents drew from leading edge thoughts on database creation and user interface design. Since that time, great strides have been made in creating databases that allow for the aircraft specific requirements to be met while providing structure that can apply across a variety of aircraft platforms. Likewise, since the advent of these Tri-Service Specifications, great thought has been given to how manuals should be distributed, and how they should be displayed. In the midst of these advances in thought, technology has also changed in unexpected ways. The Internet has provided a platform that extends beyond the "desktop" metaphor, which was inherent in early user interface designs for electronic technical manuals. Additionally, these electronic platforms have stabilized to the point where they are now viable for presentation of electronic flight manual information. As the sole reference available during flight, aircrew have demanded high reliability from supporting technologies before converting from paper to electronic flight manuals.

The three conditions, stated above, provide justification for creation of these guidelines. The first condition relies on the solid foundation provided by the USAF through the Tri-Service Specifications. The second relates to the leadership provided by the USAF in terms of innovative thoughts on how to successfully execute these complex reference systems. And finally, the third condition relates to the necessity of updating guidelines due to the advancement of hardware platforms and Internet capability. The purpose, therefore, of this document is to provide an up-to-date framework for the display of both electronic flight manuals (EFM) and Interactive Electronic Technical Manuals (IETM) data within a common web browser interface.

Scope

Structurally, this document will review literature and provide a source of guidance relevant to the display of electronic flight and technical manual information. Each design guideline leverages knowledge gained in the field of user interface design. Guidance is provided first through an overview of the process that should be taken to successfully implement a user interface. Then detailed guidance is provided in terms of *why* it is important to consider various points and *how* to address the given issue through constructive design practices. Wherever applicable, references to supporting sources are provided, so that designers and developers can obtain more information on the topic. However, it should be noted that the context of this report is for display of electronic manuals; information from the original sources has been extrapolated so that it can be applied to this context. Designers and developers referring to those original sources must be careful to consider this context when interpreting information there.

User Interface Design Process

Integral to the successful design of any system is implementation of a process by which developers may follow. In the case of reference manuals, such as EFMs or IETMs, users of these manuals (in the flight deck or in the field) must be the primary consideration. These "end users" must be considered throughout the design of the user interface. User interface designers have developed steps to ensure that the end product is compatible with the end user needs. The User Interface Design process, presented here, includes

four basic steps. As design requirements change, these four steps should be iterated to ensure effective solutions for the end users.

Step 1. Planning

The first step in user interface design begins with effective planning. Planning includes identification of goals, as well as the functions to be addressed by and requirements of the user interface. Goals must be established by the design team upon initiation of the effort (Koyanl et. al., 2003). Functions and requirements may be identified and obtained through a variety of sources, including reference documentation or state-of-the-art practices. In all cases, end users should be solicited for input as to the goals, functions, and requirements that will support their tasks. Design tools such as User Cases are often helpful for capturing the overall functions needed for and requirements of the end users.

Step 2. Concept Development and Design

Once planning is well underway, initial concepts may be captured. This step is most fruitful when a variety of independent ideas are shared with the design team (Koyanl et. al., 2003). As variations in concepts are brought forward, the better solutions will emerge. Initial concepts may be explored and shared through very crude media, such as paper and pencil. In fact, sophisticated media should be avoided during this step of user interface design. Once shared, these concepts can be merged or synthesized to provide the best overall design concept. References, such as these guidelines, should be followed in a general way. They should provide some guidance, but not limit the creativity associated with concept design. As in all steps, representative end users should be consulted as the concept begins to solidify.

Step 3. User Interface Design

Leveraging all the functions and requirements identified in Step 1 against the Design Concept, a more formal user interface design may be laid out. This design details how the user will navigate, comprehend, and manipulate information presented in the user interface. User interface prototyping tools are typically used for this step. Prototyping tools include paper and pencil storyboards, as well as electronic prototyping tools. Guidelines, such as those provided in this document should be carefully considered on every screen of the prototype. Again, end user considerations are integral to the success of the design. The prototype should allow for a series of representative end user stories

to be told or viewed by moving from one screen to the next. The range of stories, or scenarios, afforded through the design directly relates to the sophistication and value of the design.

Step 4. Testing and Iteration

As the user interface design takes shape, end users should be exposed to the design so that feedback may be obtained. A variety of scenarios or stories should be provided as a means of testing the interface. Feedback from representative users should be captured and used to make changes in the user interface design. When testing, end users must be carefully selected to ensure that they truly represent the range and experiences of the anticipated users. In addition to careful selection of users, experimenters and experimental methods must also be carefully selected to minimize any experimental or design bias that might be present. Feedback must be carefully analyzed to determine the specific changes required to the design.

Using these four steps, and iterating these steps, will provide the foundation for an effective user interface design. Using this process as a foundation, more detailed guidance can be used to identify how the user is to navigate and manipulate information in the user interface. The remainder of this document provides detailed guidance on how to implement navigation, and how to provide manipulation mechanisms in the design of effective user interfaces for electronic reference manuals.

User Interaction Guidelines

1. MACHINE CONSIDERATIONS

1.1. MINIMUM SYSTEM REQUIREMENTS

1.1.1. Ensure the system is up to date and adequate for performing standard tasks. Minimum system requirements should include an Intel Pentium III processor running Windows 2000 Professional or higher, 512 MB of RAM and 40GB of hard disk space (Lynch & Horton, 2002; IBM, 2002).

1.2. Web Browser Considerations

1.2.1. The web browser is used to display pages generated for use within the Air Force Common Viewer. The Air Force Common Viewer should include a modern, standards-compliant browser that can understand, support, and properly display HTML, XHTML, Cascading Style Sheets (CSS), JavaScript (ECMAScript), and the World Wide Web Consortium Document Object Model (DOM).

1.2.2. Many modern web browsers have the ability to display a tabbed interface, allowing the user to view and tab through multiple documents within the same window. This functionality is essential to effectively implement the Air Force Common Viewer. Currently, the web browser Mozilla would be the most appropriate application to employ for the Air Force Common Viewer.

1.2.3. Pages generated for the use within the Air Force Common Viewer should be coded such that the pages are web-browser independent.

2. DISPLAY CONSIDERATIONS

2.1. Safe Area for Printable Web Page

2.1.1. Web pages will print awkwardly if the display area is not properly sized. The recommended area for ensuring that a web page will print properly is a width of 560 pixels and a height of 410 pixels (Lynch & Horton, 2002; IBM, 2002; Koyanl et. al., 2003).

2.2. Safe Area for 800x600 Displays

2.2.1. Web pages will not display properly if coded for a larger display than is being used. The recommended safe area for 800 x 600 displays is a width of 780 pixels and a height of 450 pixels (Lynch & Horton, 2002; Koyanl et. al., 2003).

2.3. Display Viewing Distance

2.3.1. Users should be able to view the display without discomfort. The viewing distance of the display to the user's eyes should be 20 inches. For physiological reasons it is important to keep changes in viewing distance to a minimum in order to reduce accommodation time (Eastman Kodak Company, 1983).

2.4. Display Viewing Angle

2.4.1. Users should be able to view the display without discomfort. It is important to limit the viewing angles for Cathode Ray Tubes (CRT) and Liquid Crystal Displays (LCD) between 15 and 40 degrees. Users should be able to maintain these angles for extended periods of time.

3. INPUT DEVICES

3.1. Keyboards

3.1.1. The layout of the keys should be based upon their importance, function, frequency of use, and sequence of use. The standard QWERTY keyboard usually includes alphanumeric, function, auxiliary numeric, and cursor control key groups. The alphanumeric keys usually receive most of the user's attention. Frequently used function keys, such as the Enter and Shift keys, are incorporated into the periphery of the alphanumeric keys (Salvendy, 1997; Wagner et. al., 1996).

3.1.2. The auxiliary numeric keypad should be placed to the right of the alphanumeric key set within the same keyboard (Salvendy, 1997).

3.1.3. The cursor control keys provide a key based method of control for the position of the cursor. With the cursor control keys it is possible to direct the cursor on the display to left, right, up, down, and occasionally, to a home position (Salvendy, 1997).

3.1.4. In special key set arrangements, the position of the function keys are determined by the importance, frequency of use, and sequence of use of

those keys. Layouts of the function keys are usually based upon the requirements of the application (Salvendy, 1997).

3.1.5. It is helpful to provide users with multiple forms of feedback regarding their keystrokes. Tactile feedback from the actual keys is the major source of feedback. Visual feedback is important for the correction of typing errors. Audio forms of feedback can also help enhance the user's performance (Salvendy, 1997).

3.2. KEYPADS

3.2.1. Keypads provide users an additional way to enter numeric data. The standard numeric computer keypad consists of 10 keys arranged in a three by four layout (Sanders & McCormick, 1987).

3.3. TOUCH SCREENS

3.3.1. Touch screen displays are helpful interfaces that provide users with a direct relationship between the hand and the pointer location on screen. Objects on the touch screen display should be large enough for users to use a finger or stylus without hindering activation accuracy or visibility of other objects within the display (Galitz, 2002).

3.3.2. Objects on touch screen displays should be large enough to provide accuracy when used. Touch screen objects should be at least 3/4 inches by 3/4 inches in size and separated by at least 1/8 inches (Galitz, 2002) (Attachment 4).

3.3.3. Visual feedback should always be available in order to show that activation has occurred. Audio feedback is also helpful to demonstrate that activation has occurred (Galitz, 2002).

3.3.4. It is important to provide an instructional invitation to the user before he/she begins use with the touch screen interface (Galitz, 2002).

3.4. STYLUS

3.4.1. Stylus pens are pointing devices that can maintain the behavior of direct manipulation of on screen objects. Stylus pens are often used as an

alternative to using a mouse as the primary pointing device. When appropriate, use a stylus pen as the primary pointing device. For example, when using a touch screen interface it may be appropriate to use a stylus pen as the primary pointing device (Apple Computer, 1995; Wagner et. al., 1996).

3.5. Voice Commands

3.5.1. It should not be difficult for users to use and interpret the vocabulary for the voice command system. When selecting the vocabulary for voice commands it is important to use terminology that is familiar to users, and to avoid using words that are acoustically similar (Apple Computer, 2003).

3.5.2. Users should be able to determine if the voice commands they have given the system have been accepted or rejected. When designing a voice command system, it is important to provide users with feedback regarding the recognition results and system response to the voice input (Salvendy, 1997).

3.5.3. Voice commands provide a simple and direct way to interact with a computer system. Voice commands can be helpful for individuals that cannot use a keyboard or individuals whose hands are occupied. It is important that use of voice commands be limited to the appropriate environment and situation. The use of voice commands in noisy environments may cause high error rates, while utilizing voice commands in quiet environments may be disturbing to others. Speech recognition errors occurring frequently are due primarily from the speech recognizer's inability to correctly recognize the boundaries between spoken words (Galitz, 2002; Wagner et. al., 1996).

3.5.4. Users should be able to easily correct the command errors that the system interprets. If errors occur, a correction capability should be provided that minimizes demands on users and maximizes the system throughput (Salvendy, 1997).

3.5.5. Voice commands should be both short and provide the system with enough information so that the command is recognized. Use of single word commands should be avoided. Single word commands are less distinctive than multiple word commands and can be confused with the recognition system. Phrases that are three to six words long are more distinctive and can be better recognized (Apple Computer, 2003).

3.5.6. Users should have multiple ways of interacting with the application. Speech should not be the only way for a user to accomplish a task. An alternative method should always be provided (Apple Computer, 2003).

4. GLOBAL INTERACTIONS

4.1. UNDO/REDO

4.1.1. Users should always be given the ability to instantly undo or redo any action just taken. The undo/redo command allows users to reverse any immediate actions. Reversible actions are necessary, particularly when editing documents and including various features such as fonts, colors, and borders. If users are aware of the ability to reverse their actions, they will most likely feel more comfortable with making changes and trying these features (Smith & Mosier, 1986; Wagner et. al., 1996).

4.1.2. The undo/redo function should be a single toggled operation. Make certain that the function button is easily accessible and in a familiar location (Apple Computer, 1995).

4.1.3. In certain situations, it may be beneficial for the application to allow users to perform multiple undo and redo actions (Apple Computer, 1995).

4.2. INTERFACE RESPONSE

4.2.1. Everyone's definition of an acceptable response time is different. Acceptable response times not only vary with each user, but they also vary according to the task being performed. Research indicates that the type of task establishes the acceptable response time, and that most humans consider 2-3 seconds the "now." Generally, regardless of task or user, computer

activities should respond to user actions as quickly as possible to avoid delaying completion of tasks. Provide a 2 to 3 second or faster response time for tasks requiring complex or continuous activity, keeping in mind that users all have different thresholds for acceptable response times (Wagner et. al., 1996; Smith & Mosier, 1986; Koyanl et. al., 2003).

4.2.2. Some research indicates that delays are more acceptable when they are consistent, suggesting that it is not the actual length of the delay that causes frustration. Lengthy delays are, in fact, acceptable in certain situations where large files exist or unusual software packages are used. According to research, delays are more acceptable when they are consistent. Delays should not exceed half the mean response time. For example, if the mean response time is 6 seconds, a 3 second deviation would be acceptable. The response time could be as long as 9 seconds and as short as 3 seconds (Lynch & Horton, 2001).

4.2.3. Inform users of a potentially long delay prior to the start of the application. This can be done by including file formats and sizes in parenthesis after the title or link to the application, or providing an estimated time for processing of the application (Lynch & Horton, 2001).

4.2.4. When an application is processing, a display change, such as an icon or completion bar, should represent computer actions and should prompt some type of user acceptance. Providing users the opportunity to acknowledge the action ensures that users are aware that processing has completed, allowing them to move to the next task. An icon (such as an hourglass) or a completion bar should be present during information processing and after information processing has been completed (Smith & Mosier, 1986; Koyanl et. al., 2003; Wagner et. al., 1996).

4.2.5. Users should receive an immediate, visible response to every action (Wagner et. al., 1996).

4.2.6. Responses to menu selection function key presses, and most entries during graphic interaction, should appear to users to be immediate (Wagner et. al., 1996).

4.2.7. Other response times should match the user's perception of the complexity of the transaction with apparently simpler transactions having faster responses (Wagner et. al., 1996).

4.3. Function Keys

4.3.1. Function keys can be extremely helpful to users. They allow users to perform a single function with the quick press of a button. Generally they should only be used for tasks requiring frequent input. Function keys such as ENTER or DELETE are meaningful because they are simple to execute and are appropriately named to describe their function. Function keys also allow users to make inputs without repositioning their cursor and mouse. Execution of each function should be simple and easy for users to remember. Keys assigned to represent commonly used functions should only require a single activation; users should not be required to activate more than once or multiple buttons at once. If users accidentally activate a function key more than once, no negative actions should initiate as a result (Smith & Mosier, 1986).

4.3.2. Names should, if possible, be universal among software and hardware so as not to confuse users. For example, START should not have a unique, program-specific meaning – it should have the same universal meaning that users are familiar with. This is also true when considering names that may be similar in spelling or appearance. It is important to appropriately name the function key in accordance with the function it executes. When using general terms such as START, STOP, DELETE, etc., make certain that the designer's meaning of the term matches the universal meaning of the term that users are familiar with. Function keys should not be similar in spelling or appearance, such as ON and DN (Smith & Mosier, 1986).

4.3.3. If one key is used for two or more functions, it should be clear to the user which function has been activated upon key press. Function keys, if not offering immediate response once activated, should provide users with feedback indicating that their input was successful. While function keys are

generally helpful, they can also create confusion to those who may never use particular functions, or if the function is not available for the application in process. For this reason, any function that is not being used or cannot be used should be inoperable or represented differently than the other functions. If more than one function is stored on a single key, properly indicate to users which function is currently in use. If the keys are properly labeled, this can be done by illuminating the function label that is current. When neither of the two functions is being utilized, nothing on the key should be illuminated. Another suggestion is to provide users with feedback through a message on the status bar that notifies them when a function has been properly activated. If function keys are present that are of no use to the typical user, these keys should be kept hidden or locked so as not to create confusion (Smith & Mosier, 1986).

4.3.4. The location of each function is important when users are trying to familiarize themselves with the equipment. Functions that are typically used together in an application should be placed together on the equipment. This allows for easy maneuvering from key to key and helps users become more familiar with each function. Additionally, function keys should be appropriately placed based on their importance. The more crucial a function key is, the more noticeable it should be on the equipment. Ensure that functions generally used together be placed collectively to assist with ease in learning and use. If certain functions are used more frequently than others, position those functions in the most suitable location. The locations of the functions keys should correspond with their magnitude. For example, those keys that need to be accessed in emergency situations should be located in a central area. If there are keys that could possibly cause harmful or disturbing consequences, they should be physically protected or covered so they are not easily activated (Smith & Mosier, 1986).

5. WINDOWS

5.1. MAXIMUM WINDOW SIZE

5.1.1. The particular application determines the minimum and maximum window size. The size should ideally be determined by the size of the display. Users should have the option to modify their window size. This is typically done by activating an icon in the upper right corner of an application. When a window has been re-sized, it should only affect the position of the entire window, not the information being displayed (United States Government, 1992).

5.2. MULTIPLE DOCUMENT INTERFACE

5.2.1. The multiple document interface (MDI) is a specification that defines a user interface for applications that enable users to work with more than one document at the same time. With this interface a user can manipulate many documents that all appear in a single traditional window, instead of a separate window for each document (Microsoft, 2003).

5.2.2. The use of a Multiple Document Interface is not recommended for the Air Force Common Viewer.

5.3. TOOL BARS

5.3.1. Providing tool bars is a useful way to give users immediate access to frequently used commands. Those options available in the toolbar should also be available through menu commands. If a toolbar is used, make a toggle switch available to turn the toolbar on and off (Apple Computer, 2002).

5.4. MENU BAR

5.4.1. Menus present lists of items to the user. These items can be commands, attributes or states that the user can choose from the list. Each program has its own set of menus. Menu columns should be no less than five characters in width (Apple Computer, 2003).

5.4.2. Menus should be wide enough such that no items in the menu list are truncated (United States Government, 1992).

5.5. Menu Height

5.5.1. Heights of menus vary among each application. In some applications, all available menu items are not immediately visible when the menu is activated. Typically, the functions that are visible are those that are most frequently used or those that have been used recently. If the entire menu cannot be displayed in the display area, the existence of extra menu data should be indicated to the user. For further information, please refer to Submenu Indicators (Section 5.8) (United States Government, 1992).

5.6. Menu Title

5.6.1. Menu titles should represent the function they provide. Menu titles should consist of one word that appropriately represents the items contained within the menu (Apple Computer, 2003).

5.6.2. When possible, use standard terminology (i.e., File, Edit, View rather than custom phrases such as System, Modify, and Look) when designing your application. Using standard terms will help users become more familiar with the application, as well as prevent user frustration (Apple Computer, 2003).

5.7. Menu Items

5.7.1. Menus should include item labels that declare the action that will occur when the item is clicked, or attributes that convey the changes to be made by clicking the menu item (Apple Computer, 2003).

5.7.2. When a menu item is unavailable to the user, it should appear to be dimmed or grayed out (Apple Computer, 2003).

5.7.3. Capitalize all words of a menu item (Apple Computer, 2003; Microsoft, 1992).

5.8. Submenu Indicators

5.8.1. When all menu items are not visible to users upon the initial click, users should be made aware of additional options or further information available. This can be done using an icon, such as an arrow, which will prompt users to seek more items on the menu (United States Government, 1992).

5.9. Menu Item Separators

5.9.1. Separators are used partially as an aesthetic feature and partially for improved usability. Within a list of menu items, it is important to provide a line or horizontal rule to separate menu items that work or fit together in similar function. By grouping similar function items together and providing separation between the groups, users can easily locate and identify the functions from the menu that are necessary to complete their particular tasks (Apple Computer, 2003).

5.10. Menu Item Grouping

5.10.1. As mentioned previously in Menu height (Section 5.5), most applications place the most frequently used menu items near the top of the menus. By placing the most frequently used items at the top of the menu, users can quickly access the functions they need most often. Items or functions that are typically used together or for similar tasks should also be grouped together. If a menu contains a term more than twice, consider dedicating a menu or hierarchical menu to the term instead (Apple Computer, 2003; Nielsen, 1996; Nielsen, 1999; Nielsen & Tahir, 2002).

5.11. Hierarchical Menus

5.11.1. Hierarchical menus can be used to offer users more choices without taking up additional space in the parent menu. Limit the use of submenus to one level beyond the parent. Only include a submenu in a menu with a logical relationship to the choices they contain (Apple Computer, 2003).

5.12. MENU BEHAVIOR

5.12.1. To activate a menu, users simply place the cursor over the term of the menu they wish to activate. When the cursor hovers over or clicks on a particular menu or menu item, the item is normally highlighted. Typically, the color most applications use for highlighting is blue (Apple Computer, 2003).

5.12.2. Sticky menus are menus that stay open without having to continue to hold the mouse button down. If a user selects and holds a menu item for more than the double-click interval, the menu will behave in the traditional manner, closing directly after the mouse button is released (Apple Computer, 2003).

5.12.3. A toggled menu item changes between two states each time a user chooses it. The state currently in effect should have a checkmark next to it, and when menu space allows, all items should be displayed in menu (Apple Computer, 2003).

5.12.4. Menu items often have functions where the name changes to reflect the state of the function. For example, *Show Ruler* is only a menu item when the ruler is not presently visible, and *Hide Ruler* is only a menu item when the ruler is visible. In most cases, when space for both options is not available, use one item that changes wording. To create this wording, use two verbs that express opposite actions (e.g., off/on, show/hide) (Apple Computer, 2003).

5.12.5. Special characters and text styles in menus should be carefully considered. Do not use arbitrary symbols in menus, and do not use text styles in menus other than a 'style' or 'font' menu. If possible, choose symbols and text styles that are similar to those found in familiar applications. An ellipsis character (...) indicates to the user that additional information is required to complete a command. Use an ellipsis for an action requiring further user input to complete or an action that opens a setting window (Apple Computer, 2003).

5.13. Status Bars

5.13.1. A status bar is a particular area within the viewable screen that provides status information about the application being displayed on the screen. In Microsoft applications, the status bar is located at the bottom of the window, in some cases just above the taskbar. Many times the status bar will contain a range of information, such as the current page number, webpage address, or program status (i.e., loading, done, etc.). Some status bars have clickable functions that allow the user to perform certain tasks directly. While this is a unique and convenient feature, it is not always appropriate or necessary (Microsoft, 1992).

5.13.2. Status bars provide users with pertinent application status information. When designing a status bar, consider giving users the ability to customize the bar to fit their needs. For example, users might prefer to hide their status bar when not in use, or add/delete functions that may not be applicable to their tasks. If space is limited and clickable functions within the status bar are desired, always provide tool tips to help users become more familiar with these functions as this location could be non-standard and unfamiliar to most users (Microsoft, 1992).

5.14. Progress Indicators

5.14.1. Progress indicators, also referred to as progress bars, are used to display the status of an application in progress or the percentage of completion of the application in progress. They are important to users, as they inform them on the status of their application. Typical icons such as the hourglass are good for commonly performed applications that are short in duration, but a progress indicator should supplement the cursor change when applications are larger and more time consuming. A progress indicator is normally displayed as a rectangular horizontal bar that fills as the progress of the application completes, with filling occurring from left to right. Occasionally they may be displayed vertically, and if this is the case, these should be filled from bottom to top. They can be filled with a color (such as blue) or darker

gray. They are usually frozen and not interactive. Many times the pop-up or message containing the progress indicator includes textual information that clarifies the purpose of the particular progress indicator. This is helpful though not always necessary. Include progress indicators on applications that are lengthy and time consuming. If using text, make sure the text is on the outside of the actual progress indicator. (Apple Computer, 2003)

5.14.2. A determinate progress indicator should be used when the full length of the process can be determined so that users can verify how much time remains in the process. This should always associate progress with time (Apple Computer, 2002).

5.14.3. An indeterminate progress indicator should be used when the duration of the process cannot be determined (Apple Computer, 2003).

5.15. Windows Resizing

5.15.1. Default settings for windows should be established to initially present users with the desired information. The application should determine the minimum and maximum size of the window. These sizes should be based upon the resolution of the display and the constraints of the interface. The application should set the default value for the initial size and position of the window. It is important to choose a default value that is best suited for working on the type of document your application creates and that displays as much of the document as possible (Galitz, 2002; United States Government, 1992).

5.15.2. Users should be able to manipulate the size of the window by using the mouse/cursor interface. Users should be able to change the default size of the window by using the cursor to drag the size control in the lower right corner. As the user drags the size control the visible content of the window changes. When the user changes the size of the window the upper left corner of the window should stay in the same position and the visible content should stay the same (Galitz, 2002; United States Government, 1992; Wagner et. al., 1996).

5.15.3. Window resizing is necessary when the user desires to either increase or decrease the size of an active window. Windows should be large enough to display the necessary information. Active windows should always be large enough to present all relevant and expected information for the task. It is important to avoid hiding important information and to avoid the crowding of information. It is also important to minimize the need for scrolling when a window has been resized. To avoid scrolling consider using unfolding dialogue boxes, cascading windows, or tab control. Windows should be as small as possible. The window's default size should not be the entire size of the display. Important, critical, or frequently used information should be maintained on the display at all times. Information that is not important or critical should be included in another window or a dialog box (Galitz, 2002; United States Government, 1992).

5.15.4. Window resizing should be disabled on the Air Force Common Viewer touch screen to alleviate errors that could cause emergency information to be obscured from view.

5.16. Scroll Bars

5.16.1. Scroll bars permit the displaying of information that may not always fit within a window on a screen. A scroll bar should only be included when scrolling may be necessary. Scroll bars should be included in all sizable windows. If scrolling cannot be performed in a particular direction, the appropriate arrow box should be reduced in contrast or grayed out. If all the information in a window is displayed and no information is available for scrolling, both directional arrows should be reduced in contrast or grayed out (Apple Computer, 2003).

5.16.2. Most scroll bar systems consist of a scroll area, a slider box that moves within a track made by the scroll area, and directional or scroll arrows. Vertical scroll bars should be positioned to the right of the data window. The length of the vertical scroll bar should be the full length of the data window. The use of horizontal scroll bars should be avoided if possible;

however, if horizontal scroll bars are required then the scroll bar should be positioned below the data window. The length of the horizontal scroll bar should be at least one half of the length of the data window. Slider boxes should be used to indicate the location within the data window and the amount of information to be viewed. When the slider box is selected it should be highlighted in some visually distinctive way. Scroll arrows should be used to indicate the direction in which the scrolling may be performed. Scroll arrows should be positioned at the opposite ends of the scroll bar. When the size of the data window changes the components of the scroll bar should also change to reflect the current state of the data window (Galitz, 2002; United States Government, 1992; Wagner et. al., 1996).

5.16.3. It is helpful for users to have different ways to scroll through information. There should be multiple methods to manipulate or move the scroll bar: by grabbing the slider box with the cursor and moving the box in the desired direction, by selecting the appropriate scroll arrow, or by selecting a region of the scroll area in order to automatically move the slider box to that position (Apple Computer, 2003).

5.16.4. When necessary, scroll bars should be provided to allow users to manipulate information displayed in a window that is too small to display the entire contents of the information provided (United States Government, 1992; Wagner et. al., 1996).

5.16.5. Scroll bars should appear only along the right side and bottom of a window (United States Government, 1992).

5.16.5.1. When designing a Windows user interface, horizontal scrolling should be avoided when possible (Koyanl et. al., 2003; United States Government, 1992).

5.16.5.2. When designing a web user interface, horizontal scrolling should never be used (Koyanl et. al., 2003).

5.16.5.3. Both horizontal and vertical scrolling should be disabled on the Air Force Common Viewer touch screen device. Since a kneeboard device relies on a touch screen interface, providing functionality to page

through a document would be an appropriate alternative in order to avoid errors.

6. COLOR

6.1. COLOR USAGE

6.1.1. Color provides an excellent means of attracting user attention. Color should be used in moderation and only for a clearly specified requirement. If color is used excessively, it will lose its primary meaning and not be useful to users. It should be used minimally so as not to introduce too much color into the view of the user, and so that it is easy to read the information on the display (Smith & Mosier, 1986; Wagner et. al., 1996) (Attachment 2).

6.1.2. Color shall be used consistently from screen to screen and from application to application (Smith & Mosier, 1986; Wagner et. al., 1996).

6.1.3. Use colors that are socially recognized. For example, the color red indicates stop or unsatisfactory, the color yellow indicates caution or concern, the color green indicates satisfactory or acceptable conditions, and the color cyan is used for notes. When selecting which text should be a particular color, keep in mind that the user's expectations are going to reflect these societal expectations (Smith & Mosier, 1986; United States Government, 1992; Wagner et. al., 1996) (Attachment 2).

6.1.4. A browser-safe color palette should always be considered for web user interfaces. Of the standard 256 colors, there are 216 colors that will always look the same on every monitor at any resolution (Lynch & Horton, 2001).

6.1.5. Each color should represent only one category of displayed data (Wagner et. al., 1996) (Attachment 1).

6.1.6. Color should be used only as redundant information; color cannot convey information that is not also available in written format. This is particularly important for color-blind and non-sighted individuals (United States Government, 2001; Wagner et. al., 1996) (Attachment 1).

7. TEXT AND FONTS

7.1. CHARACTER SPACING (ATTACHMENT 4)

7.1.1. When designing an application, ensure that characters are properly spaced on the display. Character spacing should be no less than 0.1 character height (United States Government, 1992).

7.2. WORD SPACING (ATTACHMENT 5)

7.2.1. Spacing between words should be no less than one (nominal) character (United States Government, 1992).

7.3. LINE HEIGHT (ATTACHMENT 5)

7.3.1. Line height should be .33 of character height, exclusive of the use of sub- or super-scripts (Lynch & Horton, 2002; Koyanl et. al., 2003).

7.4. LINE LENGTH (ATTACHMENT 3)

7.4.1. Lines of text should contain no more than 10 – 12 words or about 40 characters (Lynch & Horton, 2001).

7.5. MARGINS (ATTACHMENT 3)

7.5.1. Margins should always be used to ensure that text is never obscured by borders or by information on adjacent planes (United States Government, 1992).

7.6. FONT SIZE (ATTACHMENT 3)

7.6.1. Use familiar fonts that are at least 12-point type (Bernard & Mills, 2000; Koyanl et. at., 2003).

7.7. TEXT COLOR (ATTACHMENT 3)

7.7.1. Whenever possible, text should be presented using black font color on a white background. For further information on color, please refer to COLOR (Section 6.1) (Williams, 2000; Koyanl et. al., 2003).

7.8. Type Face (Attachment 3)

7.8.1. Serif face (Times New Roman or Georgia) should be used for body text. Sans serif face (Arial or Verdana) should be used for headlines or other contrast (Lynch & Horton, 2002).

7.9. Standard Link Colors (Attachment 3)

7.9.1. Links are usually displayed in standard colors. It is important to retain these standard colors so as not to confuse users. A link that is colored blue informs users that they have not yet activated that particular link. A link that is colored red indicates the current link that has just been activated by the cursor. A purple link informs users that the link has been previously visited. When these standard colors are not used, users are left guessing as to which links they have visited and which links they have not visited. To prevent this confusion, all links should follow this general principle. Because the user expects this behavior, it should not be changed without a valid reason (Galitz, 2002; Nielsen, 2000).

7.10. Lists

7.10.1. Lists should be presented vertically on the screen (Koyanl et. al., 2003) (Attachment 3).

7.10.2. Use ordered lists when the order of entries is important (Levine, 1996) (Attachment 2).

7.10.3. Use unordered lists when the sequence of entries is not important (Levine, 1996) (Attachment 3).

7.10.4. Limit the number of items in a single list to no more than nine (Levine, 1996) (Attachment 2).

7.10.5. Limit lists to no more than two levels of primary and secondary (Levine, 1996).

7.11. Miscellaneous Text Considerations

7.11.1. Italics can be used to make figure captions or emphasized sentences and phrases stand out. Do not use it for large blocks of text, since italics typefaces are slower to read online (Levine, 1996).

7.11.2. Use boldface fonts to highlight single words. This is typically the best way to highlight key words (Levine, 1996).

7.11.3. Paragraph text should be displayed flush left (IBM, 2002).

7.11.4. Use of abbreviations should not be used unless imperative (United States Government, 1992).

7.11.5. If not otherwise specified, all measurements should be in United States standard units (United States Government, 1992).

8. DIALOGS

8.1. Dialogs

8.1.1. A dialog is a window designed to elicit a response from users. Dialog boxes should be used as the principle mean by which users interact with the underlying application. Dialogs can be alerts, fill-in-the-blanks, single/multiple choices, selection in list, or composites. Dialog boxes should appear in a consistent and prominent position on the display (Apple Computer, 2002; United States Government, 1992).

8.1.2. All dialogs should contain at minimum an OK and CANCEL push button (United States Government, 1992).

8.1.3. Headings used in dialog boxes shall be distinctive so that they are not confused with response alternatives (United States Government, 1992).

8.1.4. If a dialog box message does not end with the use of a question mark (?), then the heading should end with a colon (:) (United States Government, 1992).

8.1.5. Dialogs should never be system modal, resulting in being unable to interact with the system further until the dialog is addressed (United States Government, 1992).

9. DIALOG BEHAVIOR

9.1. MODALITIES

9.1.1. A dialog can be nonmodal, document modal, or application modal. If the notification applies to a single document, the dialog should be document modal. If the dialog applies to the application or program as a whole, or more than one document within the application, then the dialog should be application modal (Apple Computer, 2003; Wagner et. al., 1996).

9.1.2. Modeless dialogs should enable users to change settings in a dialog while still interacting with other documents or applications (Apple Computer, 2003; Wagner et. al., 1996).

9.1.3. Document modal dialogs prevent users from doing anything else within a particular document until the dialog is addressed. Users can switch to other documents and/or programs without addressing this type of dialog (Apple Computer, 2003; Wagner et. al., 1996).

9.1.4. Application modal dialogs prevent users from taking any other actions within the parent application until the dialog has sufficiently been addressed (Apple Computer, 2003).

10. ALERTS

10.1. ALERTS (ATTACHMENT 1)

10.1.1. Alerts are dialogs that appear when the system or an application needs to communicate information to users about error conditions and to warn about potentially hazardous situations or actions. Alerts shall not contain procedural steps other than those necessary to deal with the issue presented to users (Apple Computer, 2003; United States Government, 1992).

10.2. ALERT COLORS (ATTACHMENT 1)

10.2.1. Alert colors should follow universal standards so as not to confuse users. Red should be used to notify users of a warning, yellow should be used to indicate cautions, and cyan/blue should be used for any notes intended for users (United States Government, 1992; Wagner et. al., 1996).

10.3. ALERT ICONS (ATTACHMENT 1)

10.3.1. Alert icons should follow universal standards so as not to confuse users. Alert icons relating to the issue presented to users should be used when available (United States Government, 1992).

10.4. MESSAGE TEXT (ATTACHMENT 1)

10.4.1. Message information contained in alerts shall be presented in simple and brief manner. This is usually accomplished by using single keywords or phrases that would capture the users' attention (United States Government, 1992).

10.5. INFORMATION TEXT (ATTACHMENT 1)

10.5.1. The information content shall contain all necessary information to deal with the issue the user has been alerted to. It should provide comprehensive information in a brief and simple manner (United States Government, 1992).

11. CONTROLS/FORM ELEMENTS

11.1. CONTROLS/FORM ELEMENTS

11.1.1. Controls are graphic objects that cause instant actions or visible results when the user manipulates with an input device, such as a mouse. Controls can be buttons, radio buttons, check boxes, etc. Use controls and form elements when input is needed from users to modify settings or modify future actions. They allow users to assign parameters to their actions and make decisions about their tasks (Apple Computer, 2003).

11.2. FIELD LABELS

11.2.1. Labels should always appear in close proximity to an associated interactive field or form to allow users to understand what action needs to be taken (Smith & Mosier, 1986).

11.2.2. Labels should appear to the left of text fields.

11.2.3. Labels should appear above text areas, radio button groups, or check box groups.

11.2.4. Labels should appear to the right of individual radio buttons or check boxes (Apple Computer, 2003).

11.3. Push Buttons

11.3.1. A push button is a rounded rectangle with a text label on it. Clicking the push button should immediately execute a command or action. If a button initiates an indeterminate process, the button should be dimmed or status feedback should be provided. Button names should be verbs that describe the action to be taken (Apple Computer, 2003).

11.3.2. A default button should always be provided, such that if a user were to select enter it would activate the default button choice (Apple Computer, 2003).

11.4. Radio Buttons

11.4.1. Radio buttons are to be used for lists of selectable items that are related and mutually exclusive. A set of radio button should contain at least two items and a maximum of seven items. If more than seven items, consider using a pop-up menu (Apple Computer, 2003; Bailey, 1996).

11.4.2. Radio buttons should never be dynamic (Apple Computer, 2003).

11.4.3. Radio buttons should never initiate an action (Apple Computer, 2003).

11.4.4. Radio buttons should typically be displayed vertically (Apple Computer, 2003).

11.4.5. Text should follow the radio button, rather than precede it (Apple Computer, 2003).

11.4.6. Radio button groups should be limited to three columns (Apple Computer, 2003).

11.5. Check Boxes

11.5.1. Check boxes are used to select from a list of items that are not mutually exclusive. Check boxes should be displayed vertically (Apple Computer, 2003).

11.5.2. Check box groups should be limited to 3 columns.

11.5.3. Text should follow the check box, rather than precede it (Apple Computer, 2003).

11.6. Pop-Up/Pull-Down Menus

11.6.1. Menus are used to allow users to select one item from a list of mutually exclusive options. Pull-downs should be limited to 12 items per list (Apple Computer, 2003).

11.6.2. If there are less than four items, use radio buttons instead (Apple Computer, 2003).

11.6.3. An exploratory click to view contents of the menu should not select an item in the menu (Apple Computer, 2003).

11.6.4. Do not use submenus within pop-up menus (Apple Computer, 2003).

11.6.5. Pull-downs should be used only when the items in the list are easily misspelled (Jerritt & Gaffney, 2002).

11.6.6. List items within a menu should be organized in a sensible fashion (Jerritt & Gaffney, 2002).

11.6.7. List items should be visually distinctive (Jerritt & Gaffney, 2002).

11.7. Text Fields

11.7.1. Text fields are used to gather short text or numerical data from a user. Ensure that fields are set to accommodate the length of corresponding data (Smith & Mosier, 1986).

11.8. Text Area

11.8.1. Text boxes are used to gather lengthy or multiple-line text responses from a user. If text is lengthy, consider including a word wrap capability to avoid

strings of text that are difficult to interpret. Consider vertical scrolling for paragraphs of text, but avoid horizontal scrolling (Apple Computer, 2003).

11.9. Expand/Collapse Controls

11.9.1. A disclosure triangle is a triangular-shaped icon that reveals more information when clicked by users. A higher-level topic usually labels the disclosure triangle, and when the triangle is opened, more specific information or subtopics will be available. Traditionally, if the triangle is pointed down, this implies that all information associated with the triangle is viewable to users. Users should be able to see all information to the right of the triangle. A second click on the open triangle will prompt the triangle to turn to the right, closing and again hiding all of the subtopics. Use disclosure triangles only when it is important for the primary interface to remain clear and simple, yet still provide users with the option to drill down on specific topics of interests. The disclosure triangle can also be replaced with plus-minus (+/-) icons if necessary (Apple Computer, 2003).

12. GRAPHICS

12.1. Images

12.1.1. All images should be displayed using a 256-color palette when possible. This will result in a much smaller image size with quicker page load.

12.1.2. Gif images should be restricted to the use of the interlaced variety. The image will load as a full blurry image and will clear up as loading continues, instead of the image loading line by line.

12.1.3. All images must include alternative text (United States Government, 2001).

12.1.4. Images that open in a separate window should open in a window whose dimensions do not require scrolling to view the entire image.

12.2. IMAGE MINIMUM SIZE

12.2.1. Graphics should be displayed at no smaller of a scale than that required to meet the minimum displayable size that has been designated for each individual graphic (United States Government, 1992).

12.3. IMAGE MAXIMUM SIZE

12.3.1. Graphical images should not exceed 600 x 600 pixels for the Air Force Common Viewer unless extenuating circumstances exist.

12.4. IMAGE GRAPHIC DENSITY/LEVEL OF DETAIL

12.4.1. Graphic images should display only the detail that is needed to properly convey the information being described (United States Government, 1992).

12.4.2. Graphic images should be presented in a scale that allows all essential information to be entirely legible (United States Government, 1992).

12.5. IMAGE ANGLE OF VIEW

12.5.1. Graphic images shall be drawn from the same general angle of view that the equipment presents to the user (United States Government, 1992).

12.6. CAPTIONS

12.6.1. Provide captions except when the context is so clear that captions would be redundant.

12.6.2. Make sure that the caption uniquely identifies the illustration or table (Levine, 1996).

13. TABLES

13.1. TABLES

13.1.1. Tabular information should be displayed as cells of textual or graphic content (United States Government, 1992).

13.1.2. Tables should be displayed as a left-to-right, top-to-bottom array of cells (United States Government, 1992).

13.1.3. When not using a table for HTML layout purposes, tables should use both column and row headers (United States Government, 1992, 2001).

14. FRAMES

14.1. FRAMES

14.1.1. Frames should be titled with text for simple frame identification and navigation (United States Government, 2001).

15. AUDIO INFORMATION

15.1. VERBAL

15.1.1. Provide verbal audio information to notify users that something has happened in the background, such as, "Your download is finished". When using verbal audio information to notify a user it is important to pause for a few seconds between the visual display of the event and the spoken message (Apple Computer, 2003).

15.1.2. Users should be provided with control over the amount of spoken verbal output that they receive. Users should have the ability to turn the verbal output on or off within the application. Users should also have the ability to control the volume, voice, and speaking rate of the verbal output (Apple Computer, 2003).

15.1.3. Users should have verbal confirmation of the data/information that they input into an application. Information that the user enters, such as typing text into a field, should be spoken back by the computer (Apple Computer, 2003).

15.1.4. Users should be able to understand the spoken output provided by the computer application. Ensure that spoken alerts are clear and well written. Avoid long sentences and awkward phrasing (Apple Computer, 2003).

15.2. NONVERBAL

15.2.1. It should be easy for users to determine when an auditory signal is present. Users should also be able to differentiate between auditory signals with

relative ease. The selection of signal dimensions and their encoding should take advantage of the learned relationships of the user. For example, high frequencies with an up and wailing signal should be associated with emergencies. Auditory signals should be easy to distinguish from any ongoing audio input (noise or meaningful input). Auditory signals should not provide users with more information than is necessary. It is important that the same signal always represent the same information (Sanders & McCormick, 1987).

15.2.2. The dimensions for an auditory signal should be set so that users can easily interpret the information. It is important to avoid using extremes of auditory dimensions. High intensity signals may startle a user and disrupt performance. The intensity of the signal should be set so that it is not masked by ambient noise level. In order to minimize perceptual adaptation it is important to use interrupted or variable signals rather than steady state signals. It is also important to minimize the number of signals used for a given situation. Using too many signals may confuse and overload users (Sanders & McCormick, 1987).

15.2.3. It is important that sounds and auditory signals be coupled with another method of conveying the same information. Sound should not be the only means of conveying information. It is best used as a redundant or secondary form of information, or supplemented with alternative forms of communication. When sound is the primary form of information it can be supplemented with visual representation of the same information (Sanders & McCormick, 1987).

16. CURSORS

16.1. CURSORS

16.1.1. In most applications the cursor is the focus of the user's attention. Keeping that in mind, users should be able to maintain a sense of the cursor's location. It is important that the cursor be visible at all times. It should be easy to locate the cursor and to track its movement. The cursor

should contrast well with the display background. It should be designed so that it maintains its size across all screen locations and during movement. The cursor should not obscure other symbols and objects within the display (Galitz, 2002; United States Government, 1992; Wagner, 1996).

16.1.2. Having control over the cursor is one of the user's primary means of interacting with a system. It is important that users have full control over positioning the cursor. The position of the cursor should not change without commands from the user (Galitz, 2002; United States Government, 1992; Wagner, 1996).

16.1.3. The shape of the cursor can give users information regarding the purpose of the cursor. The shape of the cursor should clearly indicate the meaning and purpose of the cursor. The cursor should be created from existing defined shapes. It is important that the cursor be used only for its defined purpose. The number of shapes should be limited (Galitz, 2002; United States Government, 1992; Wagner, 1996).

16.1.4. By making the cursor change while moving, users may become confused or distracted. It is important to avoid making frequent changes when moving the cursor across a screen. To avoid such changes it may be helpful to include a short time out before making non-critical pointer changes. It is also important that cursor animations not distract users or interfere with the user's ability to interact with the system (Galitz, 2002; Wagner, 1996).

16.1.5. The arrow cursor should be used for interacting with scroll bars, other controls, menu bars, desktops, minimizing windows, maximizing windows, and cascading windows (Galitz, 2002; Wagner, 1996).

16.1.6. The arrow cursor changes to a hand cursor when the cursor is over a hot spot. This cursor change notifies users that they can obtain more information by selecting the hot spot or link (Wagner, 1996).

16.1.7. I-beam pointer, also known as the I-cursor, is a mouse cursor shaped similar to the capital letter "I" that represents text can be highlighted, inserted or changed (Wagner, 1996).

16.1.8. The blinking cursor provides visual feedback as to the placement of text if users were to immediately start typing. The blinking cursor can be moved to the desired location by either using the arrow keys or by clicking the mouse button while hovering the I-beam cursor over the desired text-insertion point (Galitz, 2002).

16.1.9. The hand cursor provides visual feedback by turning into a human hand with an index finger pointed at the target. This indicates to the user that the cursor is hovering over a hotlink that, when clicked, will lead the user to another section or page within the same or different document.

17. NAVIGATION

17.1. GENERAL NAVIGATION

17.1.1. Screen navigation should be obvious, consistent, rhythmic, and easy to accomplish. The eyes should not be forced or caused to move long distances about the display when looking for the next item. The eyes can be guided through the screen with lines formed through use of white space and display elements. By incorporating borders into the interface, the user's eyes will tend to stay within the border to complete the task. By aligning objects on the screen, scanning will be minimized (Galitz, 2002; Koyanl et. al., 2003).

17.1.2. To improve the navigation of screens that contain data, it is important to position the most significant or frequently used controls to the top left of the screen where initial attention is typically directed. This will help reduce the overall number of eye and manual control movements required to work with the screen (Galitz, 2002; Koyanl et. al., 2003).

17.1.3. It is important to maintain a top-to-bottom, left-to-right screen flow. This orientation should help reduce the eye movements between items, and make groupings more obvious. This orientation should also help users create a visual anchor point for when they move their attention away from and then back to the screen (Galitz, 2002; Koyanl et. al., 2003; Nielsen, 1996; Nielsen, 1999; Nielsen & Tahir, 2002).

17.2. SELECTABLE ELEMENTS

17.2.1. All hot spots should be visually indicated. There are three acceptable modes for depicting visual indication of hot spots. These three modes include a persistent visual indication that an area is hot, a cursor that changes in shape or color, or an object that changes when the cursor is placed over the area (Levine, 1996).

17.2.2. The cursor should provide feedback when it is placed over a hyperlink (Koyanl et. al., 2003).

17.2.3. Use meaningful link labels (United States Government, 2001; Koyanl et. al., 2003).

17.3. BOOKMARKS

17.3.1. Bookmarks are links specifically archived by users for use in the future. Bookmarks are especially helpful as they give users immediate access to information they are familiar with. Provide bookmarks for frequently referenced information or information that requires quick and easy access.

17.4. TABBED BROWSING

17.4.1. Tabbed browsing allows users to open up multiple web pages within one window and switch between them using a tabbed interface built into the window. Tabbed browsing is especially useful when the user requires only one application to be running and that application is being used for the task.

17.4.1.1. The first tab should open aligned to the left edge of the window, while new tabs should begin at the right edge of the tab that is furthest to the right.

17.4.1.2. As more tabs are opened, tab size will shrink to accommodate all currently open tabs. Tabs should never shrink to the point that the title label on the tab is truncated, thus obscuring the title to the user.

17.4.1.3. A button to close the current tab should be included on the tab itself; however, this function should be disabled on the Air Force Common Viewer touch screen.

17.4.1.4. Newly opened tabs should open in the background with the current tab retaining focus.

17.4.1.5. When a new tab is selected, the previously selected tab should save its current state such that when the tab is selected again, the user will return to the desired state.

17.4.1.6. Ability to close or open new tabs should be disabled on the Air Force Common Viewer kneeboard device to alleviate errors that could cause emergency information to be obscured from view.

17.5. PAGING

17.5.1. If a screen contains too much data to display in a single frame, the data should be partitioned into separately displayable pages (Koyanl et. al., 2003; Wagner et. al., 1996) (Attachment 5).

17.5.2. In a multipage display, each page should be labeled with the number of the page and the total number of pages, for example, "Page 2 of 3" (Koyanl et. al., 2003; Wagner et. al., 1996) (Attachment 2).

18. ICONS

18.1. ICON STANDARDIZATION

18.1.1. When the cursor is placed over an icon, the icon should reveal its name or function to the user (IETM User Interaction Guidelines, 1999).

18.1.2. The following recommended icons should use standard images when possible: *Next*, *Previous*, *Return*, *Back*, *Exit*, *Find/Search*, *Undo*, *Warning*, *Caution*, and *Print* (IETM User Interaction Guidelines, 1999).

19. CONTENT

19.1. DYNAMIC CONTENT

19.1.1. Users should always be provided with a visual cue to notify them that something has been selected. In color environments, objects may appear darker and text is highlighted with color. In black and white environments,

objects usually appear in inverse video when selected (Apple Computer, 1995).

19.1.2. Selecting an object should never alter the object itself. Users should not be committed to anything by making a selection. Users should have the ability to undo any selections by selecting another object or by mouse clicking outside the selection (Apple Computer, 1995).

19.1.3. There are several different methods used to make selections through use of the mouse and use of specific keys or key sets on the keyboard (usually the cursor keys). Several methods of making selections using the mouse or pointing device are by mouse clicking, by dragging, and by pressing the shift key in conjunction with a mouse click (Apple Computer, 1995).

19.1.4. The most basic way to select an object is by clicking it once. The user simply positions the pointer over the desired object and then presses and releases the mouse button. Icons and most other objects that can be selected are selected in this manner (Apple Computer, 1995).

19.1.5. Users can select a range of objects by dragging the mouse over the particular objects. To do so, the user will first position the cursor at one corner of the range and press the mouse button. Without releasing the mouse button, the user will then move the pointer in the desired direction. As the cursor moves, visual feedback identifies the objects that will be selected once the mouse button is released. When the visual feedback shows the desired range the user will then release the mouse button (Apple Computer, 1995).

19.1.6. Users can extend a selection by holding down the Shift key and clicking the mouse button (Apple Computer, 1995).

19.1.7. To use the cursor keys to make a selection, users will typically select and hold the Shift key while at the same time selecting the appropriate cursor key (Apple Computer, 1995).

19.2. CONTENT SEARCHING

19.2.1. Two types of searching are recommended when designing an application with search features. Basic search is generally one text field for entering search terms. Advanced search is typically multiple text areas and drop downs specifically for identifying a closer search result or match (Galitz, 2002).

19.2.2. Users should have as much control over their searches as possible. Users should be able to specify and control the extent of their searches, either confining them to within a section, across a site, within specified sources, or globally across the Web (Galitz, 2002).

19.2.3. Users should be provided with multiple ways of conducting a search through parameters such as keywords, phrases, and variants. Users should also be provided with a spell checker to help reduce the typing and detect spelling errors that may impede the search (Galitz, 2002).

19.2.4. The controls for the search interface should include a text box, structured controls, and a command button. The search text box should be large enough to enter a minimum of 20 characters. The size of the input characters should be 10-point. Structured controls such as check boxes, list boxes, or drop-down list boxes should be incorporated into the interface in order to help constrain the searches. A command button labeled "Search" should be positioned to the right of the search text box. The command button should allow activation of the search when it is selected. For a grouping of search controls, the command button should be positioned at the end of the field completion sequence (Galitz, 2002).

19.2.5. Separate interfaces should be provided for simple searches and advanced searches. Typically, a simple search interface will consist of a text box for entering keywords and phrases. When an advanced search is required, place an "Advanced Search" link under the search text box. To assist users with the advanced search, designers should provide clear instructions, the option to receive addition online help, and a search wizard (Galitz, 2002).

19.2.6. Users should be able to refine or limit the size of the results of the search. Upon requesting a search, users should be given either a total of the items in the result set, or a preliminary list of topical matches. Users should then be able to refine the search in order to retrieve the desired result set (Galitz, 2002).

19.2.7. Users should be presented with exactly the information that was requested. That information should be presented in a language and format that is easy to use and understand. Users should be presented with a summary of the search criteria along with the search (Galitz, 2002).

20. MISCELLANEOUS

20.1. PRINTER OUTPUT

20.1.1. Printer output is strongly discouraged for the Air Force Common Viewer kneeboard.

20.1.2. If there is a requirement to capture information through printing, the printed information should always include a version number or date/time stamp (IETM User Interaction Guidelines, 1999).

20.2. PAGE TITLES

20.2.1. All pages should have a descriptive title at the top of the page to describe the purpose and content of the page (Smith & Mosier, 1986).

20.2.2. Leave at least one blank line between the title and body of document (Smith & Mosier, 1986).

References

Apple Computer Inc. (1995). *Macintosh human interface guidelines.* Reading, MA: Addison-Wesley Pub. Co.

Apple Computer Inc. (2002, June 1). *Aqua user interface guidelines.* http://developer.apple.com/documentation/LegacyTechnologies/Conceptual/AquaHIGuidelines/index.html.

Apple Computer Inc. (2003, October 18). *Apple human interface guidelines.* http://developer.apple.com/documentation/UserExperience/Conceptual/OSXHIGuidelines/index.html.

Bailey, R. (1996). Human performance engineering: Designing high quality professional user interfaces for computer products, applications and systems. (3rd Ed.). Englewood Cliffs, NJ: Prentice-Hall.

Bernard, M. & Mills, M. (2000). So, what size and type of font should I use on my website? *Usability News*, 2.2, http://psychology.wichita.edu/surl/usabilitynews/2S/font.htm.

Galitz, W.O. (2002). *The essential guide to user interface design.* (2nd Ed.). New York: Wiley Computer Publishing.

Health, Safety, Human Factors Laboratory & Eastman Kodak Company (1983). *Ergonomic design for people at work.* Belmont, CA: Lifetime Learning Publications.

IBM. (2002). *IBM ease of use design guidelines.* http://www-3.ibm.com/ibm/easy/eou_ext.nsf/Publish/602.

IETM user-interaction ("Look-and-Feel") document. (1999, March). Paper presented at the AIA and Tri-Service IETMTWG Workshop, Naval Surface Warfare Center, Carderock, MD. http://navycals.dt.navy.mil/ietm/LookFeel.pdf.

Jerritt, C. & Gaffney, G. (2000, 2001, 2002). *Forms that work.* Effortmark and Information & Design. http://www.formsthatwork.com/.

Koyanl, S., Bailey, R., & Nall, J. (2003). *Research-based web design & usability guidelines.* NIH Publication No. 03-5424. http://usability.gov/pdfs/guidelines.html.

Levine, R. (1996). *Guide to web style.* Sun Microsystems.

Lynch, P. & Horton, S. (2002). *Web style guide.* (2nd Ed.). New Haven, CT: Yale University Press. http://www.webstyleguide.com/index.html?/contents.html.

Microsoft Corporation. (1992). *The windows interface: An application design guide.* Redmond, WA: Microsoft Press.

Microsoft Corporation. (1995). *Windows interface guidelines for software design.* Redmond, WA: Microsoft Press.

Microsoft Corporation. (2003). *User interface design and usage.* MSDN Library. http://msdn.microsoft.com/library/default.asp?url=/nhp/default.asp?contentid=28000443.

Nielsen, J. (2000). *Designing web usability.* Indianapolis, IN: New Riders Publishing.

Nielsen, J. (1996, May). *Top ten mistakes in web design.* http://alertbox.useit.com/alertbox/9605.html.

Nielsen, J. (1999, May). *"Top ten mistakes" revisited three years later.* http://alertbox.useit.com/alertbox/990502.html.

Nielsen, J. & Tahir, M. (2002). *Homepage usability: 50 sites deconstructed.* Indianapolis, IN: New Riders Publishing.

Salvendy, G. (1997). *Handbook of human factors and ergonomics* (2nd Ed.) New York, NY: John Wiley & Sons Inc.

Sanders, M.S., & McCormick, E.J. (1987). *Human factors in engineering and design (6th Ed.).* New York, NY: McGraw-Hill.

Smith, S. L., & Mosier, J. N. (1986). *Guidelines for designing user interface software.* Report ESD-TR-86-278. Bedford, MA: The MITRE Corporation. http://www.hcibib.org/sam/ftp://ftp.cis.ohio-state.edu/pub/hci/Guidelines/ .

United States Government (1992a, November). *Manuals, interactive electronic technical: General content, style, format, and user-interface requirements.* AMSC F6846 Military Specification, MIL-M-87268.

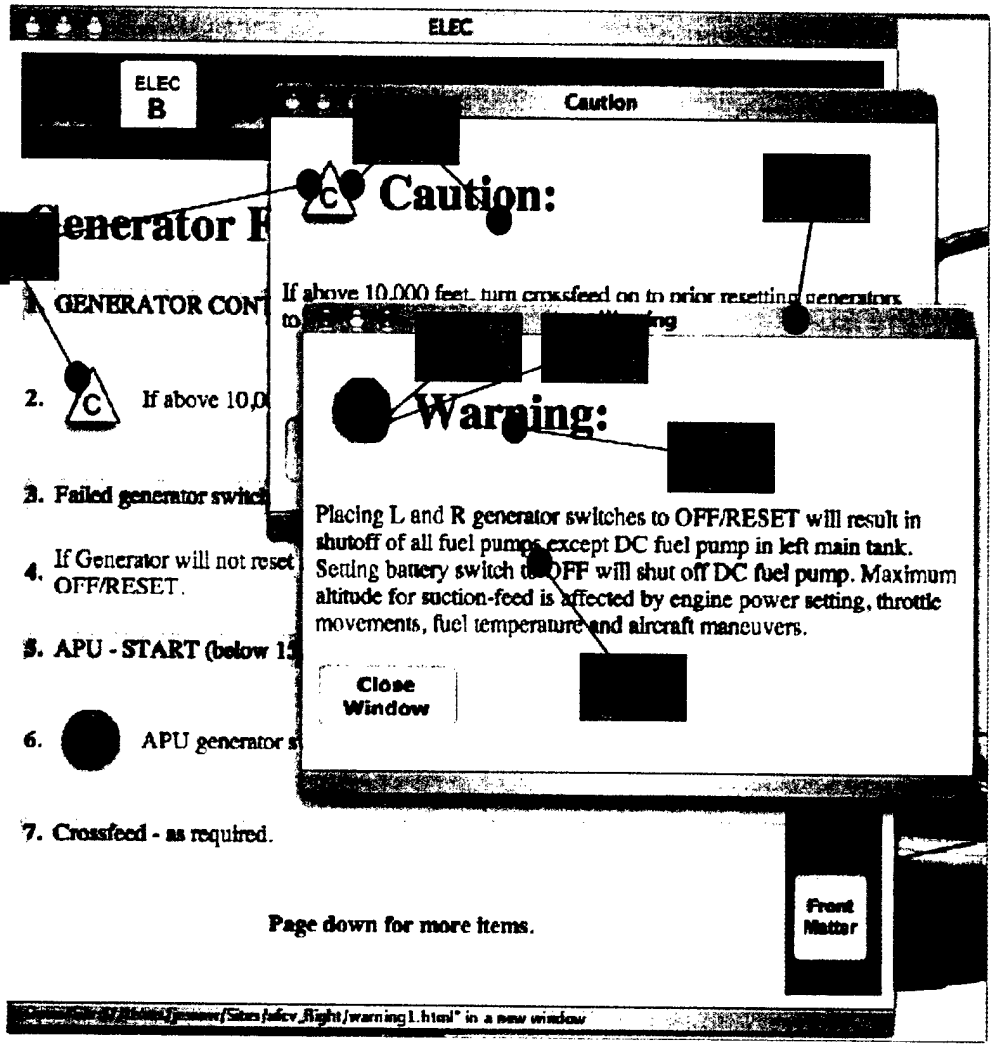
United States Government (1992b, July). *For the support of interactive electronic technical manuals.* Military Specification, MIL-D-87269.

United States Government. (2001). *Rehabilitation Act of 1973 (as amended in 1998), Section 508.* www.section508.gov.

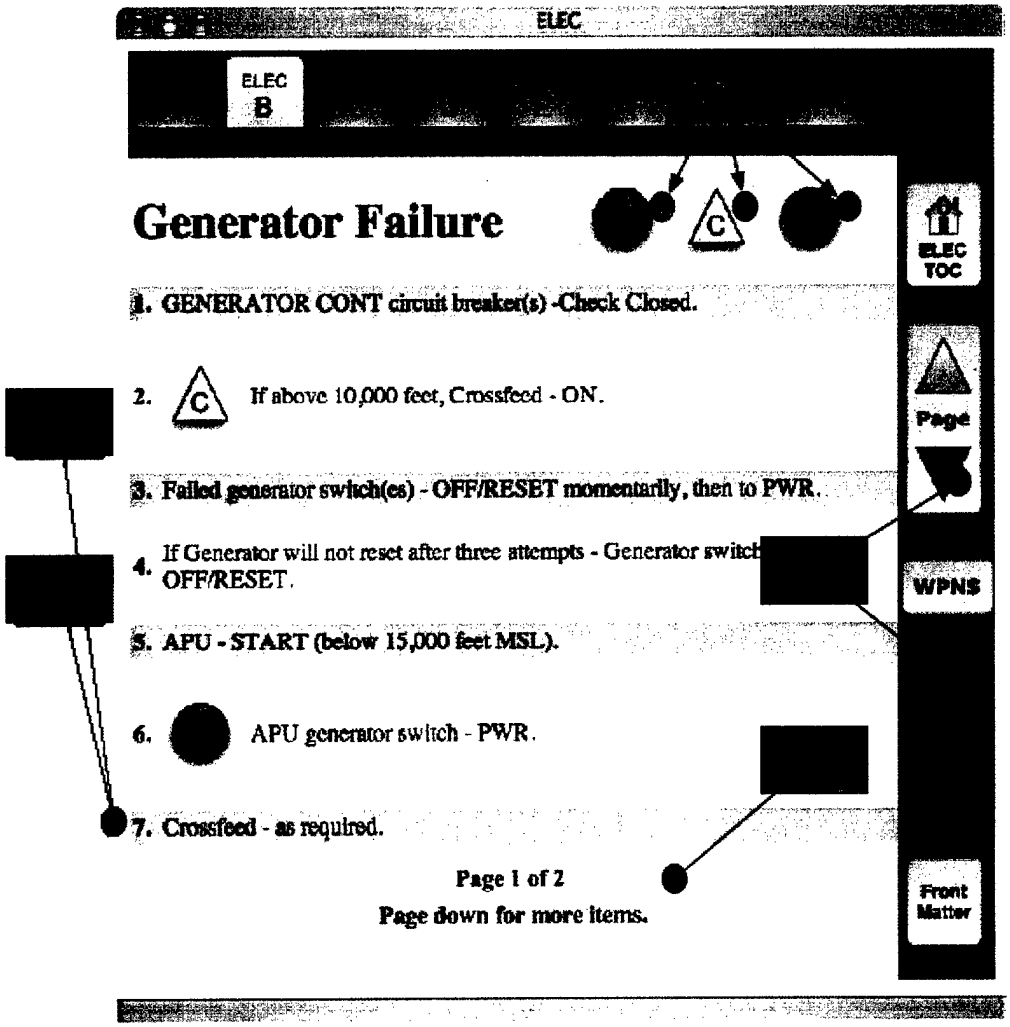
Wagner, D., Birt, J., Snyder, M., & Duncanson. (1996, January). *Human factors design guide: For the acquisition of commercial off-the-shelf subsystems, non-developmental items, and developmental systems.* Final Report and Guide, DOT/FAA/CT-96/1.

Williams, T. (2000). Guidelines for designing and evaluating the display of information on the web. *Technical Communications, 47*(3), 383-396.

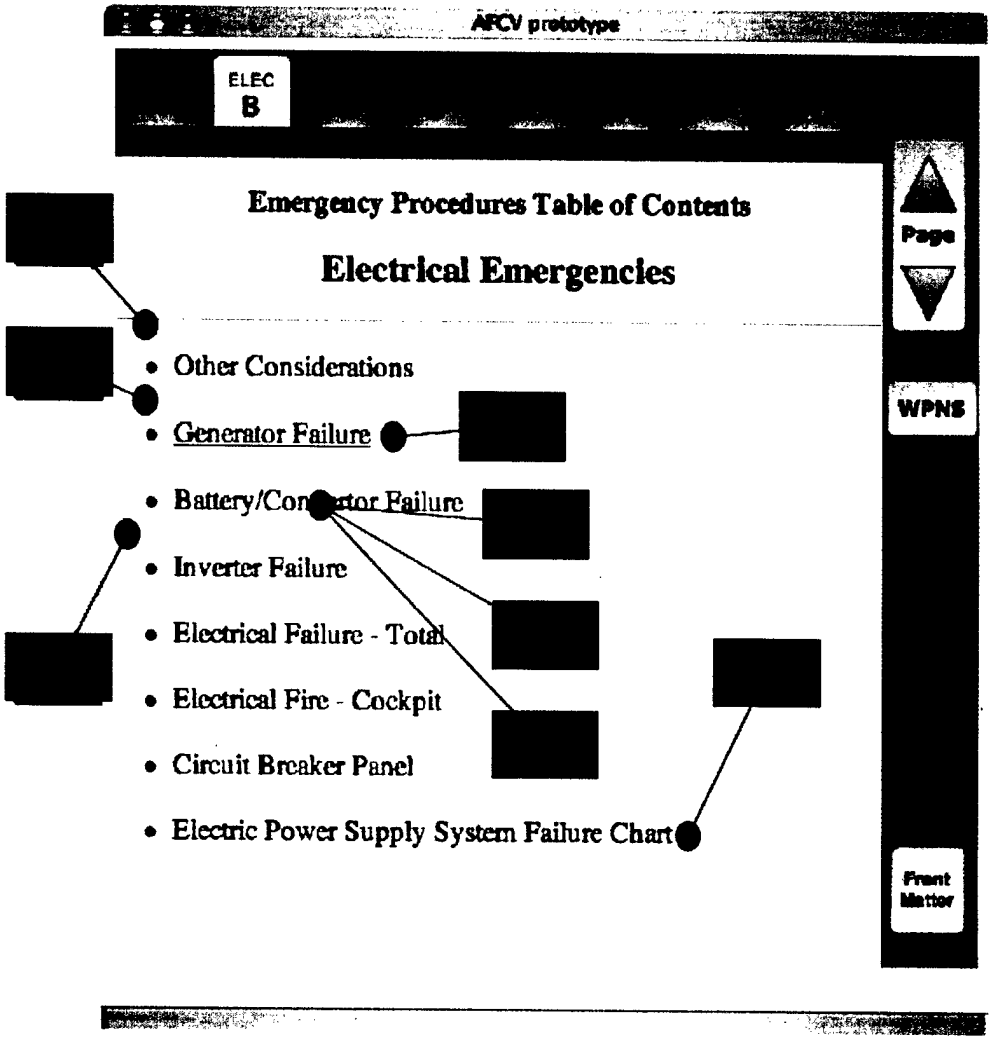
Attachment 1



Attachment 2



Attachment 3



Attachment 4

AFCV

[redacted]

TO 1A-10A-1-2 CL-1

Pilot's Checklist

USAF Series

A-10A/0A-10A

Aircraft

Published under authority
of the Secretary
of the Air Force

[redacted]

Attachment 5