

AD-A201 945

DTIC FILE COPY

②

NAVAL POSTGRADUATE SCHOOL Monterey, California



THESIS

DTIC
ELECTE
DEC 29 1988
S D

H

A USER-FRIENDLY DESIGN OF AN
INTERACTIVE PROTOTYPE FOR THE
MAINTENANCE AND MONITORING OF
CIVILIAN TRAINING RECORDS

by

Sharon Elizabeth Slominski
and
Ivon Ralph Young, Sr.

September 1988

Thesis Advisor

T.R. Sivasankaran

Approved for public release; distribution is unlimited.

88 12 29 039

Unclassified

security classification of this page

REPORT DOCUMENTATION PAGE			
1a Report Security Classification Unclassified		1b Restrictive Markings	
2a Security Classification Authority		3 Distribution Availability of Report Approved for public release; distribution is unlimited.	
2b Declassification Downgrading Schedule		5 Monitoring Organization Report Number(s)	
4 Performing Organization Report Number(s)		7a Name of Monitoring Organization Naval Postgraduate School	
6a Name of Performing Organization Naval Postgraduate School	6b Office Symbol (if applicable) 37	7b Address (city, state, and ZIP code) Monterey, CA 93943-5000	
6c Address (city, state, and ZIP code) Monterey, CA 93943-5000		9 Procurement Instrument Identification Number	
8a Name of Funding Sponsoring Organization	8b Office Symbol (if applicable)	10 Source of Funding Numbers	
8c Address (city, state, and ZIP code)		Program Element No Project No Task No Work Unit Accession No	
11 Title (include security classification) A USER-FRIENDLY DESIGN OF AN INTERACTIVE PROTOTYPE FOR THE MAINTENANCE AND MONITORING OF CIVILIAN TRAINING RECORDS			
12 Personal Author(s) Sharon Elizabeth Slominski, Ivon Ralph Young, Sr.			
13a Type of Report Master's Thesis	13b Time Covered From To	14 Date of Report (year, month, day) September 1988	15 Page Count 123
16 Supplementary Notation The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.			
17 Cosati Codes		18 Subject Terms (continue on reverse if necessary and identify by block number)	
Field	Group	Subgroup	
		user friendly, user interface, ease of use, human factors engineering, thesis, (EDU)	
19 Abstract (continue on reverse if necessary and identify by block number)			
The objectives of this thesis are to identify through research and user involvement the issues relating to user satisfaction concerning the man/computer interface and then prototype an interactive design based upon those identified issues. The Civilian Personnel Office, Naval Postgraduate School, was chosen for the area of evaluation and application, since it has on-line computer capabilities that are idle because of the difficulty of use. The thesis process developed a working prototype using guidelines identified through research and user involvement of user friendly software. An important conclusion of this study is the observation that while interactive system development should commence with independent designs of both the functional application and the user interface, recognition of the interrelationships that might be created by the implementation environment can have a significant impact on the quality of system performance and must be thoroughly investigated before final system design. <i>Keywords:</i>			
20 Distribution Availability of Abstract ☒ unclassified unlimited ☐ same as report ☐ DTIC users		21 Abstract Security Classification Unclassified	
22a Name of Responsible Individual T.R. Sivasankaran		22b Telephone (include Area code) (408) 646-2637	22c Office Symbol 54sj

DD FORM 1473,84 MAR

83 APR edition may be used until exhausted
All other editions are obsolete

security classification of this page

Unclassified

Approved for public release; distribution is unlimited.

A User-Friendly Design of an
Interactive Prototype for the
Maintenance and Monitoring of
Civilian Training Records

by

Sharon Elizabeth Slominski
Lieutenant Commander, United States Navy
B.A., Radford College, 1975

and

Ivon Ralph Young, Sr.
Lieutenant, Supply Corps, United States Navy
B.S., University of Maryland, 1980

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN INFORMATION SYSTEMS

from the

NAVAL POSTGRADUATE SCHOOL
September 1988

Authors:

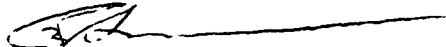


Sharon Elizabeth Slominski



Ivon Ralph Young, Sr.

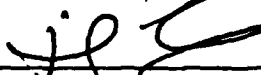
Approved by:



T.R. Sivasankaran, Thesis Advisor



Gerald L. Pauler, Second Reader



David R. Whipple, Chairman,
Department of Administrative Sciences



Kneale T. Marshall,
Dean of Information and Policy Sciences

ABSTRACT

The objectives of this thesis are to identify through research and user involvement the issues relating to user satisfaction concerning the man-computer interface and then prototype an interactive design based upon those identified issues.

The Civilian Personnel Office, Naval Postgraduate School, was chosen for the area of evaluation and application, since it has on-line computer capabilities that are idle because of the difficulty of use.

The thesis process developed a working prototype using guidelines identified through research and user involvement of user-friendly software. An important conclusion of this study is the observation that while interactive system development should commence with independent designs of both the functional application and the user interface, recognition of the interrelationships that might be created by the implementation environment can have a significant impact on the quality of system performance and must be thoroughly investigated before final system design.

Accession For	
NTIS GRA&I	☒
DTIC TAB	☐
Unannounced	☐
Justification	
By	
Distribution/	
Availability Codes	
Dist	Avail and/or Special
A-1	

TABLE OF CONTENTS

I. INTRODUCTION	1
A. BACKGROUND	1
B. OBJECTIVES	1
C. RESEARCH QUESTIONS	1
D. SCOPE AND LIMITATIONS	2
E. METHODOLOGY	2
F. LITERATURE REVIEW	2
G. LIST OF ABBREVIATIONS	2
H. ORGANIZATION OF THE THESIS	3
II. LITERATURE REVIEW	4
A. INTRODUCTION	4
B. INTERFACE DESIGN FACTORS	5
1. General	5
2. Dialog	8
a. User Adaptation to the System	8
b. Minimizing the mental load on the user	9
c. Error detection and correction	9
3. Ergonomics	9
a. Minimize worker effort	9
b. Minimize worker memorization.	10
c. Minimize worker frustration	11
4. Screen Design	12
C. CONCLUSION.	19
III. METHODOLOGY	20
A. BACKGROUND	20
B. TARGET POPULATION	21
C. FUNCTIONAL APPLICATION	22
D. USER INTERFACE	23

IV. ANALYSIS AND RESULTS	27
A. FUNCTIONAL APPLICATION	27
B. USER INTERFACE	27
C. COMPOSITE DESIGN	35
V. CONCLUSIONS AND RECOMMENDATIONS	40
A. CONCLUSIONS	40
B. RECOMMENDATIONS	41
APPENDIX A. FACTORS USED TO EVALUATE USER INTERFACE	43
APPENDIX B. APPLICATION SOFTWARE USER INTERFACE QUESTION- NAIRE	44
APPENDIX C. DATA DICTIONARY OF DATA ELEMENTS	58
APPENDIX D. PROTOTYPE: FIRST ITERATION	66
APPENDIX E. PROTOTYPE: SECOND ITERATION	87
LIST OF REFERENCES	110
INITIAL DISTRIBUTION LIST	114

LIST OF TABLES

Table 1. HUMAN FACTORS/COMPUTER KNOWLEDGE STRUCTURE . . .	7
---	---

LIST OF FIGURES

Figure 1. Principles of Ergonomic Software	10
Figure 2. Screen Design Considerations	13
Figure 3. General Screen Design Guidelines	14
Figure 4. Message Guidelines	15
Figure 5. Data Entry Screen Guidelines	16
Figure 6. Inquiry Screen Guidelines	17
Figure 7. Menu Screen Guidelines	18
Figure 8. Physical Data Flow Diagram (DFD)	28
Figure 9. Logical Data Flow Diagram (DFD)	29
Figure 10. Hierarchy Chart	30
Figure 11. Bachman Diagram	31
Figure 12. Factor Index of User Interface Results from Questionnaire	33
Figure 13. Nonresponse Rate by Factor	34
Figure 14. Factors Containing Widely Scattered Ratings	36
Figure 15. Interactive Styles	37

I. INTRODUCTION

It is fruitless to attempt to build a data base system on technological element alone. Systems management must take into account not only the management of the technical system but of the people system as well (Peck,Peck, 1974). [Ref. 1: p. 10]

This statement made in 1974, is still as relevant today as it was yesteryear. In past decades people went to the computer, while today the computer comes to them, e.g., desktop computers, networking, etc. Consequently, the need for a user-friendly interface in this highly technical environment is a must.

A. BACKGROUND

Personnel records of civil service employees of the Department of the Navy are maintained by the local Civilian Personnel Office (CPO). Each CPO is a separate organization that supports the local Navy commands in administration of civil service personnel. One important area of record keeping is training. Civilian personnel offices are required to utilize and maintain DD Form 1556 for recording training of civilian employees. The tracking of this form through the various phases of the training evolution is burdensome and demanding. The awkwardness and inadequacies of the existing automated mainframe system has resulted in user rejection of the system and the continuation of the manual system. This coupled with the limited staff in the civilian personnel offices hinders the effective and efficient monitoring and maintenance of civilian personnel training.

This thesis is centered around developing a user-friendly design of an interactive prototype for the maintenance and monitoring of civilian training records using computer interface characteristics developed with user involvement.

B. OBJECTIVES

The objectives of this thesis are to identify the issues relating to user satisfaction concerning the man-computer interface and prototype an interactive design using those user-identified attributes desired in a computer interface.

C. RESEARCH QUESTIONS

The focal issues of this thesis are:

- What are the specific attributes of the man-computer interface for the existing mainframe system that are generating user dissatisfaction resulting in user rejection of the system?

- What man-computer interface characteristics will create a user-friendly atmosphere and promote user acceptance?
- What attributes should be used in developing the user-friendly prototype design?
- How should the data concerning the user interface be gathered? In what form?
- What criteria or standards should be used to analyze and evaluate user interface data?
- How can these criteria or standards be validated?
- Will the change in user-friendliness increase the potential for user acceptance and use of the system?

D. SCOPE AND LIMITATIONS

The scope of this thesis is limited to the development of an interactive prototype design of a user-friendly man-computer interface. This design was tailored to the wants and requirements of the local civilian personnel training director.

Unfortunately, the area of focus was bounded to the local civilian personnel office. This was the result of the unwillingness of other training directors to cooperate with the study. Many reasons were given; however, the reasons pointed to the idea that since nothing would be gained, why bother.

E. METHODOLOGY

This thesis followed a three-step process. First, information was gathered through an extensive literature review. Second, information was gathered through a specially-designed questionnaire and with personal interviews. Finally, using the information obtained, development of an interactive prototype design was accomplished.

F. LITERATURE REVIEW

The literature review resulted in the gathering of information that substantiated the need for a user-friendly man-computer interface and provided guidelines to obtain the same.

G. LIST OF ABBREVIATIONS

The following abbreviations are used within this thesis:

ADP	Automated Data Processing
APPSGEN	Application Generator
CPO	Civilian Personnel Office
DD	Department of Defense, e.g., DD Form 1556
ED	Employee Development

DFD	Data Flow Diagram
MIS	Management Information Systems
NCPDS	Navy Civilian Personnel Data System
NPS	Naval Postgraduate School

H. ORGANIZATION OF THE THESIS

Chapter II exhibits a literature review pertinent to current thinking regarding the need for a user-friendly interface for information systems.

Chapter III describes the methodology used for the functional analysis of the application and for the development of a user interface satisfaction measurement questionnaire.

Chapter IV examines the results of the functional analysis and design, the results from the analysis of the user interface questionnaire, and the results of the composite design via the prototype.

Chapter V presents the summary and recommendations.

II. LITERATURE REVIEW

The MIS director of a Fortune 500 company was remarkably candid in a recent meeting when he explained his company's sudden interest in user-friendly software design. "In the past five years, we've produced more than one multimillion-dollar system that failed when we put it in the field because it was too difficult to use. That can't happen again." [Ref. 2: p. 53]

A. INTRODUCTION

The impact of computers and their widespread use in recent years has produced a high demand for information systems that are "easy-to-use" or "user-friendly". This demand has forced increased emphasis on the design of the user-computer interface. [Ref. 3]

To be a successful system today, the system must be efficient, be easy to learn, and be easy to use for the intended users group [Ref. 2 : p. 53]. All factors indicate that if systems are not friendly, then people will avoid using them. Friendliness is no longer an optional extra, but the most important component of system design. Unfriendly systems can frighten off potential or actual users [Ref. 4: p. 99]. Consequently, any standard must incorporate the needs and characteristics of the user [Ref. 5].

A system that is user-friendly for end users not only achieves better user acceptance, but it also provides the following [Ref. 2: p. 53]:

- Will require less support.
- Will require less documentation.
- Will require less training.
- Will produce lower error rates.

Therefore, a user-friendly information system will allow users to perform desired tasks without frustration as well as provide a range of functions and features which will help each individual user reach his or her ultimate efficiency [Ref. 4: p. 105].

User-friendly software facilitates communications between the computer and the user. It adapts to the user's perspective. Therefore, user-friendly software must speak the same language as the user regardless of his or her experience level. The only way to develop user-friendly software is to ensure direct user involvement at the initial step of the software development life cycle and continue user interaction throughout the subsequent steps of the process. [Ref. 6]

Users will have different expectations of computer systems and will approach their tasks in different ways according to background, personality, level of computer literacy, and so on. "A user-friendly system must be able to cater for all categories of users and accommodate their requirements." [Ref. 4: p. 102]

B. INTERFACE DESIGN FACTORS

1. General

A critical design objective for today's information systems is user-friendliness. For visual display terminal-based systems, the design of screens and how a person interfaces with a computer through them is an important ease-of-use determinant [Ref. 7: p. 6]. Stahl proposes three required steps in designing systems that are easy to use [Ref. 2: p. 53]:

1. Understand how people think, see, and use tools to do work, i.e., knowledge resulting from human factors research.
2. Develop a collection of software techniques that take advantage of these principles.
3. Select the right techniques for the target group of end users.

"A well-designed interface can attract new users, and a poorly designed one can turn them away." [Ref. 8: p. 56] One of the hardest problems faced by organizations who implement information systems in the office is the employees' resistance to change [Refs. 9, 10]. Resistance comes in two basic patterns; fight or flight. Fight includes overt aggression, e.g., as hostile behavior or sabotage; and flight includes increased tardiness, reduced employee performance, chronic absenteeism, and apathy [Refs. 11,12].

Research has shown that a user-satisfaction approach, i.e., based on the user's subjective judgement, is preferable to an approach based on objective measures of usage and performance [Ref. 13]. Therefore, if an information system is a technical success but is not used or is used incorrectly, the system is a failure. "The key to system success is a well-designed user interface." [Ref. 8: p. 61]

Stahl suggests that four variables have proven useful for profiling end users and their environment [Ref. 2: p. 65]:

- User sophistication.
- Frequency with which users use the system.
- Time pressure on users.
- Error penalty levied by the system.

User sophistication and frequency of use can be further divided into low and high levels. Users with low computer sophistication report a strong preference for:

- The initiative to come from the computer.
- Assurance the the user cannot hurt anything.
- Clear, limited choices.
- Confirmation of successful completion of all operations.
- Messages that are non-threatening in tone.

However, users with high levels of computer sophistication are highly interested in shortcuts and the availability of status information. Sophisticated users like to know what control facilities are available; they want to set all the operations. End users with a low frequency of use want:

- The initiative to come from the computer.
- Orientation and navigation guides.
- Good help facilities.
- Mnemonic commands.

While on the other hand, end users with a high frequency of use typically want fast paths and minimal keystrokes. [Ref. 2: p. 65]

To ensure user satisfaction, Houghton suggests that before proceeding with full-scale design, developers of information systems and computer software should build a prototype that simulates the user interface. Both developers and users should test the prototype and provide feedback to help uncover user-acceptance problems early in the development cycle. In fact, most user-interface issues can be resolved through simulation. [Ref. 8: p. 61]

Recently, there has been increased emphasis on designing information systems with the intended users of such systems foremost in mind. Yet the evidence suggests that typical information system designs do not always satisfy users [Refs. 14, 15 ,16]. Often this dissatisfaction occurs because information system designers have a limited range of computer knowledge in mind, when the actual range required for a user-friendly system is much greater [Ref. 17: p. 333].

Fried suggests that it is possible to describe criteria for desirable software as clearly as those for hardware. He proposed a classification that focuses on the effect software has on its users [Ref. 18]. Martin devised a human factors/computer knowledge structure, see Table 1 on page 7, using the works of other authors. His results

suggest the following implications for information system designers and educators [Ref. 17: pp. 341-342]:

1. The human factors/computer knowledge structure shown in Table 1, when properly programmed, may result in interactive systems that effectively differentiate between novice and experienced users.
2. When effectively designed, interactive systems designed for a broad range of user knowledge may perform as well as systems matched to user knowledge, and may perform better than systems unmatched to user knowledge.
3. There is a time penalty associated with the general-audience system due to its alternate processing-path triggering question. This may make the general-audience system slower than limited-audience systems under matched conditions. Despite this inherent time penalty, the general-audience system may perform faster than limited-audience systems under unmatched conditions.

Table 1. HUMAN FACTORS/COMPUTER KNOWLEDGE STRUCTURE: [Ref. 17 : p. 335]

Human factor	Human subfactor	User computer knowledge	
		Novice	Experienced
Nature of message	Tone	Explanatory and polite	Short and to the point
	Use of humor	Careful	None
	Bypasses	None	Allow
	Warnings	Many	Rarely
	Screen format	Menu	Inquiry
	Input verification	Always	Rarely
	High lighting	Some (judiciously)	Little
	Defaults	With explanation	Without explanation
Help function	Screen discontinuation	Prompt and keyed response	Keyed response without prompt
Help function	Procedures	Full, unsolicited	Upon request
	Values	Full, unsolicited	Upon request
Response time	Mean	Minimize within variance	Minimize
	Variance	Minimize	Minimize within mean
Path process	Menu structure	Depth	Breadth
	Overall screen density	Minimize	Maximize

Those systems designed for a target group at a specific level of user-computer knowledge ignore important differences among users. Systems designed for experienced users lose the novice user, while systems designed for novice users frustrate the experienced user. Yet, interactive computer models designed for both novice and expert are harder to develop because alternate user knowledge menus and processing paths must first be decided upon and then developed. This requires additional programming code, produces higher complexity, is more expensive, and is more time-consuming. Therefore, general-audience systems must demonstrate increased user effectiveness to offset the additional development resources required. Martin has provided experimental results that indicate that systems can be designed to differentiate between novice and experienced users, and that a general-audience system may be viable. [Ref. 17: p. 333]

2. Dialog

Gaines proposes the following dialog programming guidelines which are concerned with the user's adaptation to the system and which will minimize the mental load on the user [Ref. 19]:

a. *User Adaptation to the System*

Programming the system to minimize the mental workload on the user is advantageous for close user-computer interaction. However, in so doing one concentrates upon the weaknesses of the human mind, taking into account limited short-term memory capacity, inaccuracy of calculation, etc. It becomes easy to begin to think of the computer as the senior partner which must somehow compensate for the inadequacies of the users, its weaker partners. The only problem is that in concentrating on the weaknesses of the person, we forget their strengths, one of which is the modelling of the environment.

(1) *The user will model the system.* Do not assume that the user is a passive static system to be controlled, modelled, and directed by the computer. Evaluate all actions of the system in terms of their effect on an actively changing user who is attempting to comprehend the system.

(2) *User should dominate computer.* Either computer or user should dominate interaction or there will be instability. If the computer is to dominate, it must be programmed to, and have sufficient information to, model the user. However, if the user is to dominate, then the computer system must be simple to understand. Consequently, with the present state-of-the-art, the user should dominate the system.

(3) *Avoid acausality.* Make the activity of the system a clear consequence of the user's actions.

(4) *Parallel-sequential trade-off.* Allow the user maximum flexibility to make his responses in parallel or in sequence according to his wishes.

b. *Minimizing the mental load on the user*

(1) *Uniformity and consistency.* Ensure that all operational procedures and terminology are consistently applied and are uniformly available throughout the system activities.

c. *Error detection and correction*

(1) *Validate data on entry.* Check syntax and values, but beware of rejecting data or querying too much. Have the user revalidate important updates prior to acting upon them.

(2) *Provide a reset command.* The user should be able to clearly abort a transaction at any time in a transaction with a system command that will return him back to a well-defined checkpoint as if the transaction had never been initiated.

(3) *Provide a backtrack facility.* Allow the user to return through the dialog sequence in reverse.

(4) *Make corrections through re-entry.* Use the entry dialog with default field printouts from a record as a means of correcting the record.

3. *Ergonomics*

Frequently, computer ergonomics is considered to involve only operator setting, i.e., special furniture, keying in data with equipment optimized for light, glare, low muscle fatigue, and other traditional ergonomic factors. However, Knittle suggests it is just as important for the messages, menus, graphs, prompts, interfaces, syntax, and other features of the user software to be "humanized". Ergonomic software, see Figure 1 on page 10, refers to any feature which make a difference in the user's output, with particular emphasis on those characteristics not related to the aspects of the physical work place. [Ref. 20: p. 164]

a. *Minimize worker effort*

"A worker should be required to perform only that work which is essential and cannot be performed by the system." [Ref. 20: p. 164]

Work done in the past should not be repeated [Ref. 20: p. 164]. Schmidt showed that repetition leads to boredom which may negate any production gain made by automating the task [Ref. 21].

- Minimize Worker Effort
- Minimize Worker Memorization
- Minimize Worker Frustration
- Maximize Use of Habit Patterns
- Maximize Tolerance for Human Differences
- Maximize Tolerance for Environmental Change
- Notify Users of Problems Promptly
- Maximize Worker Control of Tasks
- Maximize Task Support

Figure 1. Principles of Ergonomic Software: [Ref. 20: pp. 164-171].

"Workers should not be required to search for system information." [Ref. 20: P. 164] On-line documentation, i.e., help routines should be available. The answers to the user's questions should be immediately available in order to minimize user effort, and the user should not have to pick up the manual, search the index, and find the pages to solve a problem. [Ref. 20: p. 164]

Morland recommends that the data-entry screen be presented to the worker in a format as similar as possible to the written-data-entry form being used. The worker knows where to find each piece of information on the written form and the effort required to transfer that datum to the system can be minimized by providing a similar screen layout. [Ref. 22]

b. Minimize worker memorization.

"Workers should be required to memorize as little as possible." [Ref. 20: p. 165] Less training will be required if the system requires a minimum level of memorization of the user [Ref. 20: p. 165]. Research has shown that people can remember approximately seven things, plus or minus two, and this short-term memory decays in about 15 seconds [Ref. 2 : p. 53]. Therefore, "the worker should not be required to learn anything not necessary to the task." [Ref. 20: p. 165] When learning a relatively small part of the system, the user should be rewarded with the ability to perform some limited amount of real work. "Terminology should be consistent throughout all software with which a worker will interface." [Ref. 20: p. 165] This concept could be critical to minimizing worker memory. If the user is required to memorize many sets of

terminology, i.e., one in order to update his data file, another in order to run some analyses, another in order to format a report, and yet another in order to send that report to his employer's terminal, the amount of memorization required is enormous. [Ref. 20: p. 165]

Warner also agrees with this principle of consistent terminology throughout a system, and more importantly says it needs to be in "the user's vocabulary" in order to orient the user's task domain [Ref. 23].

c. Minimize worker frustration

"Systems should spare the worker frustrations that may arise from a delay in the accomplishment of a task." [Ref. 20: p. 166] Chafin recommends that a program notify the user "if an operation will take longer than 15 seconds." He suggests that worker frustration caused by long response time can be minimized by a system message indicating expected duration. [Ref. 24] According to Stahl, people are happier if they know how long they will have to wait, if they feel that the wait is justified, and if they know how things are going [Ref. 2: p. 56].

"User fatigue and mental laps are brought on by comparatively slow system response times." [Ref. 20: p. 166] A phenomenon apparently related to the individual's attention span is that for each second of a system response degradation, a similar degradation will be added to the user's time for the following command [Ref. 20: p. 166].

"If menus, prompting, or other guidance techniques are used, the system should permit the experienced user to bypass them." [Ref. 20: p. 166] Guidance techniques are definitely an advantage to the beginner or the casual user, but a stream of system messages and prompts that are no longer read, frustrates the user [Ref. 20: p. 166].

Dean notes that some system messages are useful when learning a program. As the worker gains experience, these messages are ignored. He recommends that the user should be able to select those system prompts which he no longer wants displayed. However, if the user is allowed to turn off system messages that warn of critical errors, more frustration may be created than removed. [Ref. 25]

"If a worker is interrupted in the performance of a related series of actions, the system should (upon request) provide a summary of the actions performed prior to the interruption." [Ref. 20: p. 166] If a user's thought process is broken by the system or by some external occurrence, the user will probably not remember precisely at what point that the interruption occurred. Consequently, the user is forced to retrace his

actions manually. He could easily become irritated, especially, if he is involved in a complicated task. [Ref. 20: 166]

4. Screen Design

Other than the computer hardware, the screen layout will be the user's first encounter with the user interface. It can either be user-friendly or a barrier. User-friendly guidelines for general, message, data entry, inquiry, and menu screens guidelines will be presented. Refer to Figure 2 on page 13, Figure 3 on page 14, Figure 4 on page 15, Figure 5 on page 16, Figure 6 on page 17, and Figure 7 on page 18.

"Screen design is not yet a precise science;" however, the knowledge derived from experimental studies is growing, and a wealth of information derived from research is available to provide interim guidance until more research questions are answered [Ref. 7: p. 7]. "Screen design is important not because screens are the heart of an interface, but because good screens serve as a prerequisite for building a useable interface." [Ref. 2: p. 60]

Unfortunately much screen design occurs with little to guide it. The human factors involved are not well understood or neglected entirely. It often tends to be unsystematic, inconsistent, and fails to adequately reflect human perceptual and processing capabilities. As a result, many screens in today's systems are difficult to use and lack visual clarity. [Ref. 7: p. 6]

In information systems the frequent result of poor screen design results in far less productivity than need be. For large systems, a few extra seconds for processing each system screen can translate into many people-years of improved productivity. Dunsmore reports an instance where improving the screens readability yielded a twenty percent increase in productivity [Ref. 26]. At worst infrequent users of systems i.e., professionals and managers, may reject a system entirely if screens leave the impression that understanding them will take far more time than they have available [Ref. 7: p. 6].

Screen design may be contributing to the visual fatigue which is being reported by some system users. Eye movement studies of data-entry operators have shown instances where visual movements between screen and source documents exceed several thousands within one work day. A significant difference in the brightness level between source document and display screen can also result in eye muscle fatigue. Consequently, this has led to attempts to brighten the display screen, or lower the illumination to try to achieve the proper balance. In fact, several thousand eye movements a day may actually reflect poor screen design rather than an unsatisfactory environment. Thus, in

A well designed screen

- Reflects the needs of its users.
- Is developed within the physical constraints imposed by the terminal.
- Utilizes the capabilities of its software effectively.
- Is developed, if used for data entry, within the constraints imposed by related source materials, such as worksheets, forms, or manuals.
- Is consistent within itself, with related screen formats and other screens within the application and the organization.
- Achieves the business objectives of the system for which it is designed.

Figure 2. Screen Design Considerations: [Ref. 7: p. 7]

some cases it appears that it's the symptoms of the problem rather than the cause that are being addressed. [Ref. 7: pp. 6-7]

Stahl recommends the following guidelines for designing good screens [Ref. 2: p. 60]:

- Do not crowd the screen. Good screens look good.
- Use highlighting, blinking and reverse video sparingly. Over use will lead to operator fatigue.
- Use color sparingly.
- Limit the amount of information on each screen to what is necessary. Do not force the end user to remember things from one screen to the next.
- Minimize cursor movement.
- Minimize keystrokes.
- Show the maximum permissible length of an input field with underscores, highlighting, or brackets.
- Exceptions for data entry:
 - Make the screen mirror the input document even if it is a mess.
 - Let operators correct errors all at once at the end of a document, rather than as each error comes up.

Galitz suggests a properly designed screen will reflect the considerations outlined in Figure 2. In addition, design consistency is a vital human consideration in

- **APPEARANCE**
 - Clutter-free
 - Cohesive groupings
 - Obvious indication of what is being displayed
 - Clear indication of what relates to what
- **COMMANDS**
 - Consistent field positioning
 - Clear field identification
 - Relate to users's conception of what is being accomplished
 - English language oriented
 - Capable of abbreviation or concise notation
 - Consistent meaning between applications
- **FIELD CAPTIONS/LABELS**
 - Meaningful
 - Distinguishable from one another
 - Distinguishable from data fields
 - Obvious association with data fields
 - Contained on one line (not stacked)
 - Upper case characters
- **DATA FIELDS**
 - Distinguishable from field captions/labels
 - Obvious association with field caption/label
 - Directly usable form

Figure 3. General Screen Design Guidelines: [Ref. 7: p. 7]

screen design. "Design consistency enables a screen user to learn concepts and apply these concepts to a family of screens, and a family of systems." [Ref. 7: p. 7] Learning capacities will then be devoted to how to use the system to enhance one's job, and would not be consumed in understanding meaningless differences [Ref. 7: p. 7].

The quantity which constitutes too much information has not yet been determined. However, the answer should reflect the application requirements and the screen

- Consistent positioning
- Consistent identification
- Short, meaningful, common and fully-spelled-out words
- No word contractions, short forms or abbreviations
- Brief and simple
- Directly and immediately usable
- Affirmative
- Active voice
- Non-authoritarian
- Non-threatening
- Non-anthropomorphic
- Non-patronizing
- In temporal sequence of events
- Humor carefully used

Figure 4. Message Guidelines: [Ref. 7: p. 9]

format. People have subjective preferences for the amount of information presented on a display, and those subjective ratings will decline as the amount of information displayed deviates either way from the preferred amount. "One researcher reports that a well-designed page of printed material has a density loading of only 40 percent and that qualitatively judged good screens possessed a loading of about 15 percent." [Ref. 7: p. 8] These mentioned examples should not be construed as guidelines or absolutes. The ultimate determination of the values depends upon various complicated factors, many of which remain poorly understood. However, "screens should display only relevant information." [Ref. 7: pp. 8-9] As information increases, competition among screen components increase for the user's attention. Screens that flood a person with too much information will only prolong visual search times and make meaningful patterns more difficult to perceive. [Ref. 7: p. 9] Figure 3 on page 14 is a checklist of the basic attributes a screen should possess.

An information system communicates with users through the many kinds of screen messages, i.e., prompts, status messages, error messages, and diagnostic messages.

1. GENERAL

- Information Grouping
 - Logical, orderly and meaningful to user
- Keying Procedures
 - Manual tabbing for large volume/many screen tasks
 - No recording, including, omitting or changing data based on special rules or logical transformation
- Character Entry
 - Accomplished by direct character replacement
 - Keyed entries always visible (except secure entries such as passwords)
 - Data keyed without separators, delimiters or dimensional units
 - Data keyed without leading zeros
 - Right or left justification not required
 - Removal of unused underscores not required
- Screen Transmission
 - Accomplished by single explicit action when all entries are completed

2. WITH SOURCE DOCUMENT

- Screen Organization
 - Image of source document
- Field Captions/Labels
 - Abbreviations/contractions separated by hyphens
- Data Fields
 - Optimally, field identified by characters such as underscores
 - Minimally, starting point of field identified

3. WITHOUT SOURCE DOCUMENT

- Screen Organization
 - Columnized for optimum visual clarity
- Field Captions/Labels
 - Fully spelled out in natural language
- Data Fields
 - Optimally, field identified by characters such as underscores
 - Minimally, starting point of field identified

Figure 5. Data Entry Screen Guidelines: [Ref. 7: p. 10]

A message needs to minimize confusion and ambiguity while allowing easy, correct, and fast interpretation. Searching through reference material to translate a message is unacceptable, as are requirements for transposing, computing, or interpolating. [Ref. 7: p. 9]

- Screen Organization
 - Logical, orderly and meaningful to user
 - Most frequently requested information on earliest screens
 - Most frequently requested information on a screen in the upper-left portion
 - Perceptually organized in a balanced manner
 - Columnized for optimum visual clarity
 - Not packed with information
- Field Captions/Labels
 - Fully spelled out in natural language
- Data Fields
 - Natural split or breaks included
 - Recognizable orders
 - Accepted organizations and formats
 - Justified for ease in scanning
 - Visually emphasized

Figure 3. Inquiry Screen Guidelines: [Ref. 7: p. 11]

Many people are still threatened by the computer, and until an optimal user-friendly interface emerges, messages need to remain factual, informative, and avoid any attempt to humor or punish. [Ref. 7: p. 9] Figure 4 on page 15 provides guidelines for developing effective screen messages. [Ref. 7: p. 9]

If used, a specially-designed source document from which data is keyed is the most important variable in data-entry-screen design. The main visual focus of the user will be towards the source document, with the screen assuming a secondary role. However, if a source document is not developed, the user's primary visual focus will be the screen. This distinction is important since it determines whether keying aids are built into the screens or into source documents. The resulting screens will have fundamental conceptual differences in data organization, content, and structure. [Ref. 7: pp. 9-10] Figure 5 on page 16 provides broad guidelines for developing data entry screens. [Ref. 7: pp. 9-10]

- Structure
 - Hierarchic structure of logically related elements
 - Only relevant alternatives
 - All relevant alternatives
 - Critical or frequently chosen alternatives immediately accessible
 - Consistency in terminology and ordering
 - Distinctive levels for groupings
 - Location in hierarchy described
 - ESCAPE mechanism
- Screen Organization
 - Perceptually organized in a balance manner
 - Columnized for optimum visual clarity
- Ordering Rules
 - Seven or less option--sequenced or frequency of occurrence
 - Eight or more options--alphabetic order

Figure 7. Menu Screen Guidelines: [Ref. 7: p. 11]

Another consideration when designing screens is to optimized human scanning. Scanning ease is accomplished by providing interpretable and easily identifiable items of data or information on the screen format, i.e., an inquiry screen. "The design objective of an inquiry screen is human ease in locating data or information." [Ref. 7: p. 10] The normal human visual search pattern is to locate the grouping within which the items reside and then to find the specific item within this group. Most often, data fields themselves are used for this visual search. Consequently, data fields should be visually emphasized. [Ref. 7: p. 10] Figure 6 on page 17 summarizes broad guidelines for effective inquiry screen design.

All systems need to have a method of identifying the available transactions and/or screens within the system for use by the user, as well as a method of allowing the user to select the proper transaction and/or screen for display. A common technique to accomplish this is through the use of a menu screen. [Ref. 7: p. 10]

One basic function of all menu systems is to provide information to the user. The adaptability of menu-based systems to many diverse applications and their simplistic approach to user interaction has contributed significantly to the widespread acceptance of menu-driven systems. [Ref. 27]

"Menu screens are particularly effective because they also utilize the more powerful human capability of recognition rather than recall." [Ref. 7: p. 11] All choices and alternatives can be listed. "The primary design objective for menu screens is ease of visual scanning. The secondary objective is ease of alternative or choice selection." [Ref. 7: p. 11] Figure 7 on page 18 provides broad menu screen design guidelines.

C. CONCLUSION.

The Susan B. Anthony dollar is an example of nonacceptance by its intended users. It parallels a system with no moving parts, i.e., simple, efficient, and was a designer's dream. Of most importance, it worked. The coin is legal tender for a dollar, is portable, and technically surpasses the one dollar bill; the system it was to replace. It even costs less to produce; however, the public has rejected this new system so flatly that today the U.S. Government has half a billion Susan B. Anthony coins locked away in storage. The coin failed solely on the issue of user friendly. Coupled with its appearance being too much like the quarter and cash register drawers would require redesigning to provide a place for it, its small size implied the shrinking value of a dollar. Consequently, the coin was a technically-superior 100 million dollar flop. It failed because it didn't fit nicely into the environment for which it was made. The analogy to software is obvious. [Ref. 2: p. 66]

III. METHODOLOGY

A. BACKGROUND

Previous informal discussions with management at the Civilian Personnel Office at NPS indicated the need for assistance in developing user-friendly automated systems for certain functions that were still based on cumbersome manual methods of operation. The duties of the Civilian Training Director were identified as one critical area where automation could significantly enhance both the efficiency and effectiveness of the function. Though the duties of the Civilian Training Director were collateral, they directly impacted the individual's primary duties as Employee Development Officer for NPS.

The preliminary stages of the research involved gathering background information to understand the duties and responsibilities of the Civilian Training Director and how her current system basically operated. Information was obtained through interviews and a review of formal organizational documents. The only current source of automation to assist the Training Director was the Employee Development (ED) Subsystem of the Navy Civilian Personnel Data System (NCPDS); a mainframe system operated at a contractor site in Oakridge, Tennessee. It was evident from interviews with both management and the perspective end-user that the NCPDS (ED) was perceived to be non-user-friendly and incapable of providing the specific type of support required to effectively perform the tasks required by the Civilian Training Director. The user's primary dissatisfaction with the NCPDS (ED) system centered around the complexity of the user interface. The perception was that there was too much one had to read and learn before even using the system; that the commands employed by the NCPDS (ED) were merely codes that did not relate to the functions nor terminology of the user's environment. The result was that the user rejected the NCPDS (ED) entirely, and continued the perpetuation of a cumbersome manual system using mass storage techniques for filing and tracking records. As expected, the manual system does not support rapid data retrieval, ad hoc queries and report generation capabilities. The following conditions were also observed:

- It is difficult to sort records for different phases of processing.
- Too much time is spent in the clerical duties of training record maintenance.
- Too much is spent gathering and collating report data.

- Scheduling of training is not optimal in regards to funds allocated for training and accomplishing mandatory command training.

To better understand the user's dissatisfaction with the NCPDS (ED) system, it was imperative that we receive a demonstration of the system and review some of the reference documentation. Arrangements were made for the ADP Technical Coordinator at NPS CPO to provide both a brief and demonstration of the NCPDS system itself since the individual was unfamiliar with the ED subsystem. After reviewing the reference material provided, it was apparent that the user's perceptions of the ED subsystem were valid and not merely the result of a lack of computer experience or system training.

Clearly, the rejection of the NCPDS (ED) system was the result of a two-fold problem; specifically lack of functionality and poor user interface. Therefore, we decided that the methodology required for our study to develop a user- friendly interactive system to support the Civilian Training Director's functions would be based on two separate design approaches, specifically, application functionality and user- friendly interface. Only by addressing both issues individually would user satisfaction be insured.

B. TARGET POPULATION

Ideally, we would have liked a target population of civilian training directors at major CPO sites throughout the continental U.S.. However, this endeavor was impractical to coordinate due to costs, communication factors, and time constraints. Thus, the original decision was to focus on those training directors located at CPO's within the driving distance of NPS Monterey. Five CPO sites were identified. Contact was made with the regional civilian training director to inform and ensure approval and cooperation of the effort from upper management. Points of contacts were acquired for the respective CPOs. Phone contact was then made with the CPOs with disappointing results. Some were not interested in participating due to their own endeavors to develop a similar system to meet their individual requirements; others were uninterested due to staffing/time constraints, and some simply failed to return phone calls.

Therefore, we were left with the local CPO site to draw upon for available subjects to assist in defining functional and user interface specifications. We decided not to worry about attempting to design a Navy-wide generic functional training and tracking system, but allow the local civilian training director to specify the system's functional requirements according to local policies and procedures. However, since there are periodic turnovers in the position of civilian training director, we wanted to ensure a generic user- friendly design of the human-computer interface. To accomplish this, we decided

to draw upon the other employees within the local CPO to assist in defining the user interface specifications. The only criteria required of the subjects was to have experience using the NCPDS system. This was essential in order to benefit from previous experience gained from using an interactive system. We felt that this approach would provide a more realistic foundation for the creation of a more objective and readily acceptable user interface by preventing a reoccurrence of past mistakes. It was determined that ten employees fit the criteria.

C. FUNCTIONAL APPLICATION

Shneiderman noted that when designing interactive systems functionality is central and must be ensured before proceeding. "If the functionality is inadequate, it doesn't matter how well the human interface is designed." [Ref. 28: p. 9] A review of research literature indicated that researchers and practitioners of Software Engineering had provided us with a plethora of methodologies for conducting a structured analysis and design of a functional application. Most consist of a step by step approach evolving from the general to the specific and keyed to the system life cycle concept. As stated by Page-Jones, the importance of using the structured approach is that it "resists making decisions on how the problem is to be solved until what the problem is has been determined." [Ref. 29]

Using previously acquired information and applying a composite of techniques based on data flow-oriented analysis and design as presented by McMenamin, Stevens, Page-Jones, and Davis [Refs. 29,30,31, 32], detailed information regarding the existing system utilized for maintenance of civilian training records was gathered over a series of intensive interviews with the training director. Relevant organizational documents and currently employed paper records/reports that pertained to the system and its maintenance were examined. A physical data flow diagram (DFD) was developed and refined via the user which depicted the entire functional processes as currently performed by the user. From the physical DFD, a logical DFD evolved that addressed the essential functions required to perform the desired tasks. The logical DFD was validated by the civilian training director.

Based on the analysis, a relatively small database system was designed. Due to limited computer resources available to CPO, the database system had to be designed to function on an XT or At clone capable of running dBase III Plus software.

D. USER INTERFACE

While software engineering provides numerous tools and techniques to aid the programmer/analyst in software development, it has yet to derive a similar framework for designing the user interface. Prototyping has been one approach to designing the user interface. The main advantage to prototyping is that it allows a high degree of user involvement and thus ensures that the design is effective and meets user acceptance. However, the iterative nature of the process can be very costly in both time and money. Literary research has revealed numerous checkoff lists or rules delineating basic design guidelines to "user-friendly" systems. Little guidance, however, has been established other than "know your user" that offers the designer a methodology to select from these lists the particular factors which are of most importance to his user's needs regarding the task to be accomplished. Costs, time, and system constraints prohibit inclusion of all criteria from the checklists, so how does one proceed to determine the user's preferences such that the environment the system presents to the user allows the user to perform his tasks with relatively little effort?

The approach we selected for this phase of the study was based on the concept proposed by Kerlinger called research design [Ref. 33: p. 300]. The research design approach involves planning a strategy of investigation in order to obtain answers to research questions. Kerlinger stated that "research designs are invented to enable the researcher to answer research questions as validly, accurately, objectively, and economically as possible." [Ref. 33: p. 301] With the characteristics of validity, accuracy, objectivity, and economics in mind, we began to formulate a framework for our user interface design approach.

First, we had to identify what aspects of the user interface might be used to assess a successful design. Shneiderman [Ref. 34: p. 14] identified five measurable human factor goals that can be readily applied to the evaluation of the user interface. These are:

- Time to learn. How long does it take the average user to learn to use the system?
- Speed of performance. How long does it take the user to perform his primary set of tasks on the system?
- Rate of errors by users. How often and what types of errors are made when performing the primary set of tasks?
- Subjective satisfaction. Are the users satisfied with aspects of the interface?
- Retention over time. How well do users remember how to use various aspects of the system?

Obviously, no system can be perfect in all the above areas; trade-offs are inevitable. These trade-offs should be based on user and not designer preference. However, a minimum level of performance should be expected in all categories.

Next, we had to determine how to rank the relative importance of each of these categories with the selected user group. It was decided that this might be readily accomplished by evaluating these issues from the perspective of having the users evaluate a functionally similar interactive system that they were already familiar with. Our rationale was that even though people are evaluating the user interface with respect to a particular system, they are nonetheless indicating a preference for what properties they want an interface to possess. Therefore, we would be able to conclude what particular factors affected user satisfaction and their relative importance to the user.

Different methods for evaluating the user interface issues were investigated through research literature to determine what methods had previously been employed. While interviews may elicit valuable unforeseen information, surveys appeared to be the best option for collecting the required data for the following reasons [Refs. 35,36]:

- They can be self-administered.
- They are usually the least costly means.
- They avoid any bias of the interviewer.
- They are less disruptive to the perspective participants schedules.

Shneiderman's questionnaire was determined to be the easiest to adapt to our purposes [Ref. 34: p. 400]. It addressed those factors which, we had determined from earlier interviews, were of significant concern to the users. The measurement instrument the questionnaire employed utilized an ordinal scale, specifically, a semantic differential scale. We felt this would not only aid the user in interpreting the questions better, but also aid us in a more accurate assessment of the responses. As discussed by Buzzell, Cox, and Brown, interpreting the degrees of response alone can be misleading [Ref. 37].

Although it is possible to attach numbers to such a verbal scale for purposes of analysis, it should be remembered that this is still an ordinal scale, strictly speaking, and estimates with regard to intervals or degrees of responses are simply estimates. Quantify them if you will, but there is no direct means of determining the validity of such quantifications. [Ref. 37]

We chose to assume that by using an existing questionnaire, we were relieved of the responsibility to test the questionnaire for its content validity and reliability.

The questionnaire was modified according to the following criteria:

- Questions were organized to follow specific user-interface issues.
- Questions wording were enhanced for clarity and closure.
- Scales supported the question being asked.
- Scales were made understandable and unambiguous.
- Comment section was offered for every question.
- Instructions provided were clear and concise.

Since the perspective subjects were required to have experience using the NCPDS system, we felt that the need to provide definitions on user interface terminology was not necessary. The questions were grouped into 28 specific user interface factors. A list of these factors can be found in Appendix A. Where applicable, questions were further organized such that preceding questions would help focus the issue of the next question.

The questionnaire was designed to rate specific aspects of the user interface that could easily be related back to Shneiderman's five human factor goals. Though the rating process is subjective and highly dependent upon user experience with computers and the particular system under evaluation, we felt that the questionnaire would provide us the necessary research tool to help us obtain accurate and objective information on the group's preference for those particular factors of the user interface design that most impacted user satisfaction. Thus, the initial design for our user interface would be built on an analysis of the composite view of the group's preferences for dialog and interactive styles. From the vantage of this preliminary design, prototyping would commence to determine the specific dialogs and screen designs.

The first draft was evaluated for completeness and understandability. Though length was of some concern, we felt it was better to err on the side of detail vice brevity. Modifications recommended by NPS faculty concerned wording of the scale to classify the question and presentation sequence of a few of the questions. An additional question was recommended to capture the experience level of the respondent. The major problem with the questionnaire was dealing with the degree of rating. The original eleven seemed too numerous, making it difficult to draw conclusions; however, 5 seemed too few to capture or distinguish finer variations. If people did not have a strong opinion, the tendency would probably be to mark 3 or 4. The number originally proposed was 9. However, it was later recommended that the scale be dropped to 7 degrees. The final draft of the questionnaire is contained in Appendix B.

It was decided that the questionnaire would be hand-delivered, with a brief explanation of our objectives to the ADP Technical Coordinator who would be distributing

the questionnaire. Arrangements were made to pickup the completed forms one week later, so as to not impose on the subjects' schedules. It was, however, requested that the questionnaires be completed at one seating vice intermittently since it would negatively impact the progressive building of the questions.

IV. ANALYSIS AND RESULTS

The purpose of this study is to design a "user-friendly" application system that will monitor and maintain civilian personnel training records. This section will first examine the results of the functional analysis and design, next it will examine the results from the analysis of the user interface questionnaire, and then it will examine the results of the composite design via the prototype.

A. FUNCTIONAL APPLICATION

The structured analysis produced both a physical DFD and a logical DFD. The physical DFD, Figure 8 on page 28, reflects the various sources and sinks involved plus the inputs-processes-outputs currently performed in the existing Civilian Training System. The logical DFD, Figure 9 on page 29 reflects a high level view of the essential activities of the Civilian Training System that are scheduled for automation. Appendix C contains the list of the 58 data elements identified to fulfill the required functions. Figure 10 on page 30 is the complete hierarchy chart for the system. Figure 11 on page 31 is the Bachman Diagram of the database design.

B. USER INTERFACE

Of the ten questionnaires provided to the subject group, six were completed. Factor analysis was used to reduce the individual variables into their respective underlying user interface factors. Factor analysis techniques allow analysis of various attributes of some product to derive a small number of underlying dimensions of product quality [Ref. 37: p. 198].

A factor index was created to empirically analyze the user group's rating of the system's user interface. The index was constructed by accumulating the scores assigned to the individual attributes from every questionnaire. The semantic differential scale used in the questionnaire provided a rating system based on an ordinal scale that goes from most negative feelings to the most positive feelings. The labels used allowed the respondent to evaluate his feelings with respect to the extremes, then place himself at the proper rating on the continuum. Since the scale was qualified from worst case to best case, it is not unlike the Likert Scale, and could be evaluated similarly. We felt at liberty to interpret the ratings this way based on the following supposition. Our rationale was that even though the respondents were evaluating particular attributes of the user

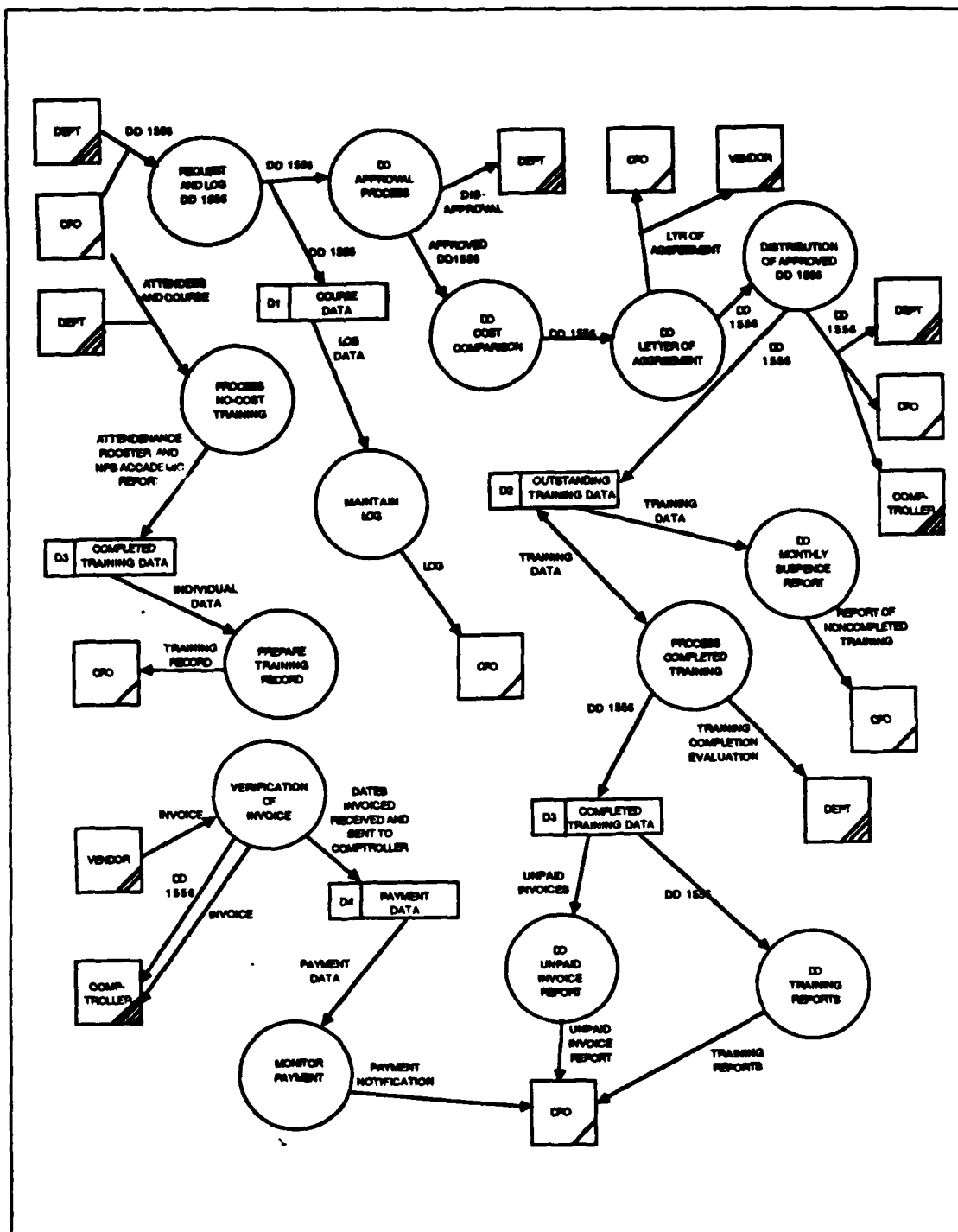


Figure 8. Physical Data Flow Diagram (DFD)

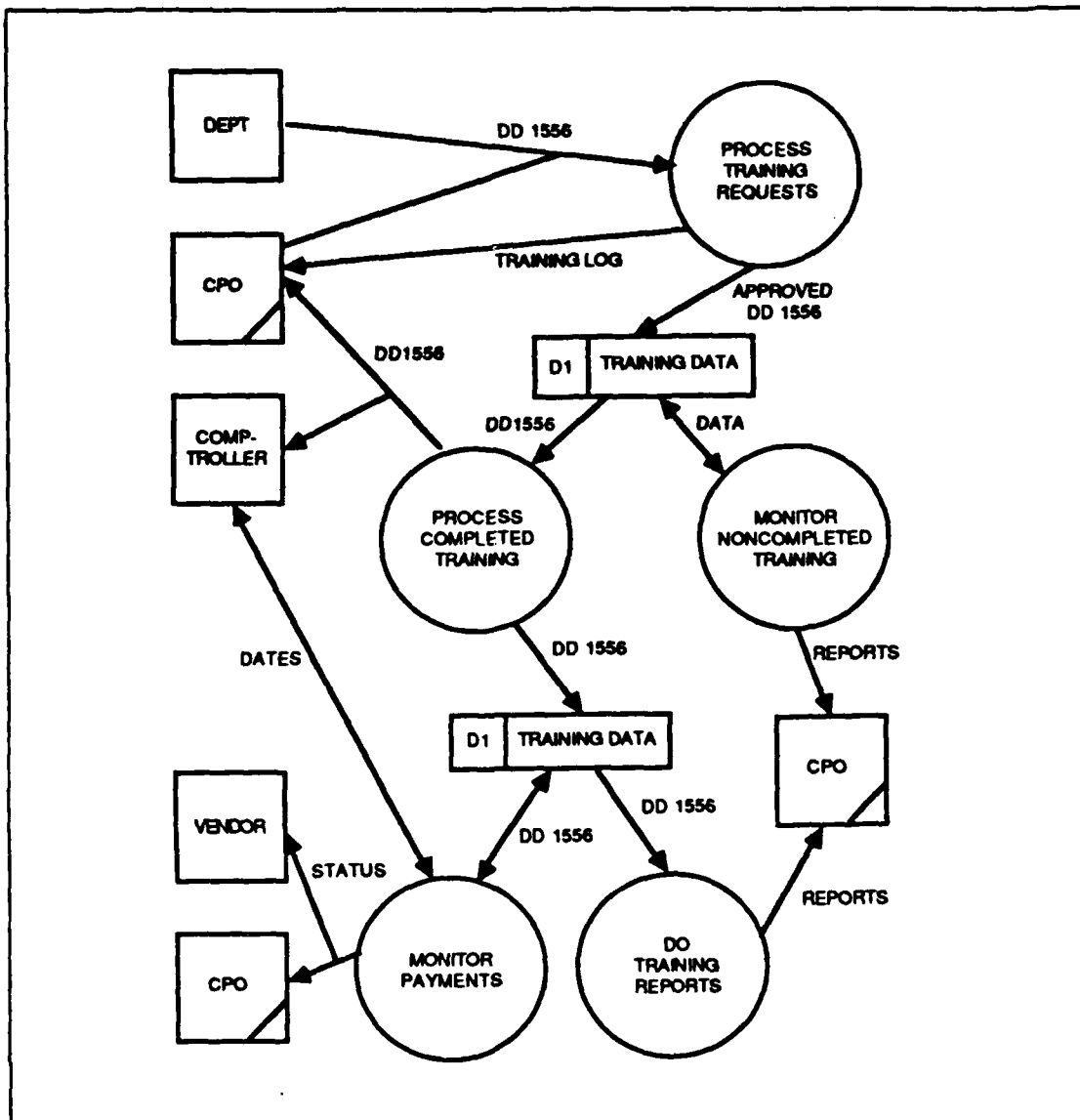


Figure 9. Logical Data Flow Diagram (DFD)

interface with respect to a particular system, their ratings are subjective in that they are based on a comparison with other systems that they have been exposed to; therefore, we felt that the ratings do in fact reflect a degree in preference for the attribute; specifically, they were either satisfied or dissatisfied with the attribute. The nonapplicable choice was available in the event that the attribute did not exist.

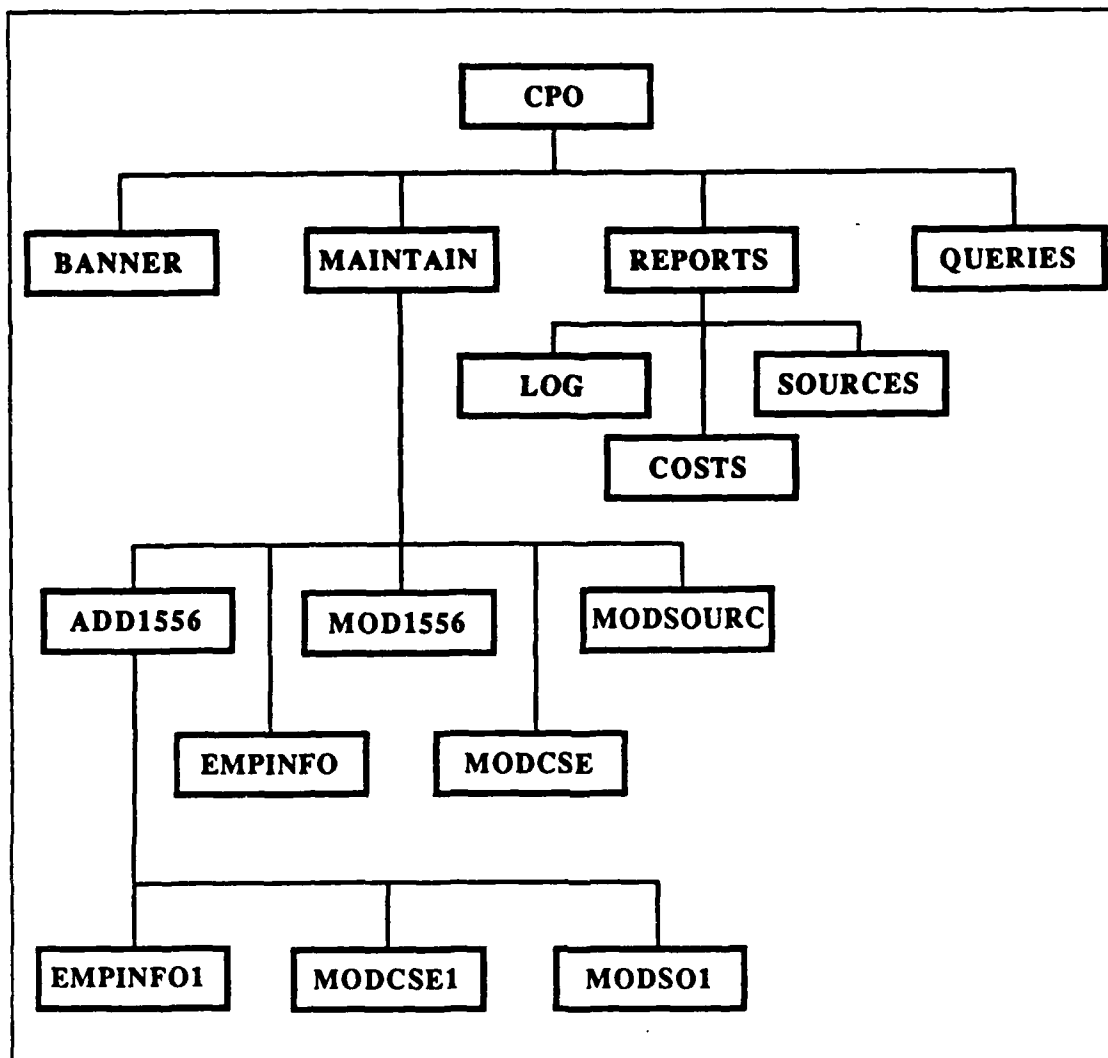


Figure 10. Hierarchy Chart

The scale was quantified by assigning the values 1, 2, 3, 4, 5, 6, and 7 ranging from the worst to the best respectively. Nonapplicable scores were not included in the development of the index. Using the values indicated, the scores from each questionnaire were averaged for every factor to arrive at the following factor index formula:

$$F_j = \frac{1}{N} \sum_{k=1}^{v_j} f_k$$

, where

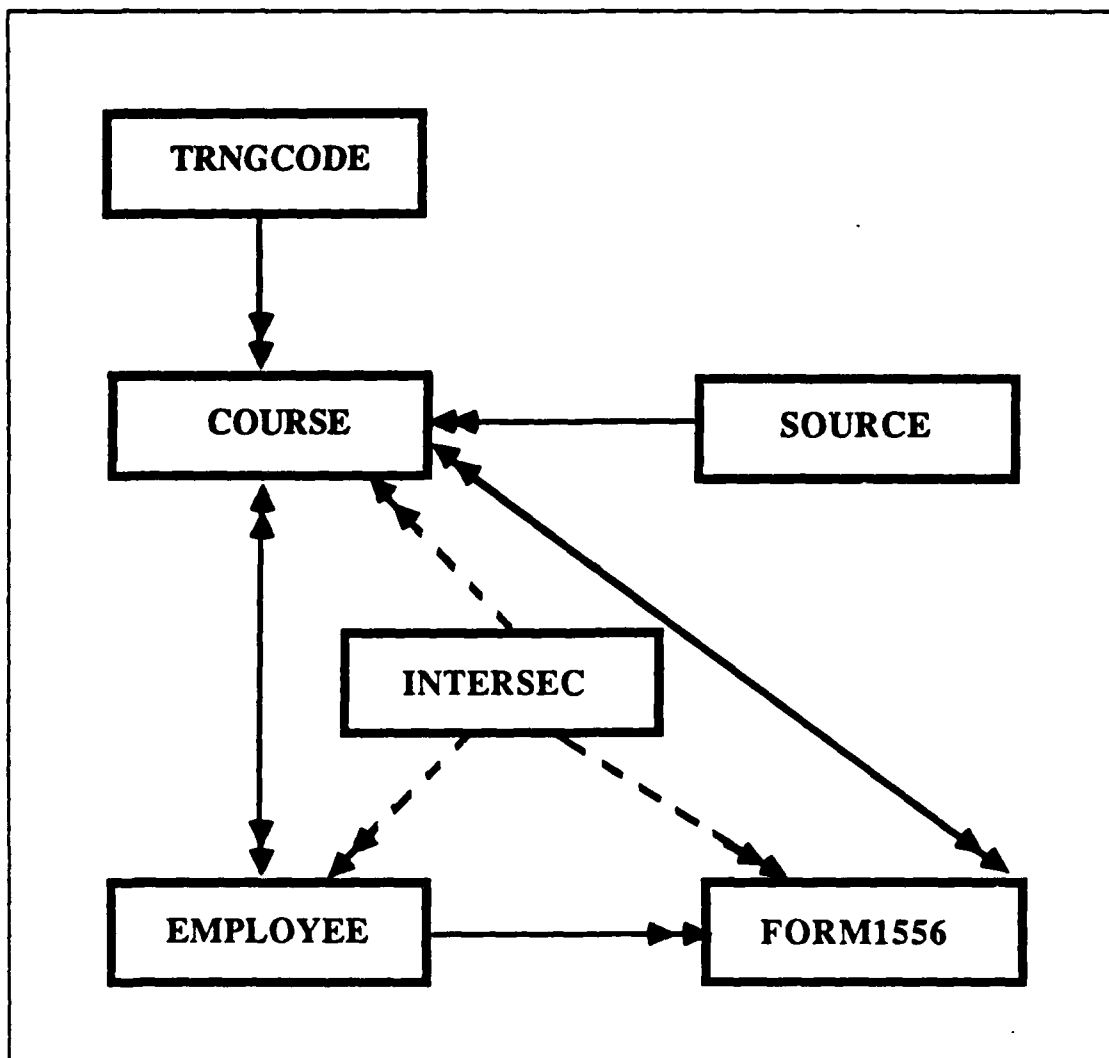


Figure 11. Bachman Diagram

F_j = individual factor index

N = number of applicable responses

v_j = total number of variables for the factor

I_{jk} = numeric score for item k of factor,

= 1, 2, 3, 4, 5, 6, 7

Figure 12 on page 33 shows the results of the factor index. The results were rather surprising since previous discussions with some of the participants indicated a higher level of dissatisfaction with the user interface than the questionnaire results reflected.

Though the index to the Overall Satisfaction factor was 3.9, if one examines all the factors in the index, the impression is that the respondents feel that the user interface is adequate; it's not perfect, but it's not imperfect either.

We were also disconcerted with the unusually large number of unanswered questions for some of the factors. Since our sample size was extremely small, the results are exceedingly sensitive to statistical analysis. Figure 13 on page 34 shows a histogram of the nonresponse rate by factor. If just one person failed to answer every variable within a factor, then the nonresponse rate for the factor was automatically 17%. Therefore, we concentrated our study on those factors whose variables consistently had 2 or more nonanswers. Four factors were identified, they were Display Layouts, Instruction Use, Operation Relation, and Feedback. It was noted, however, that the indices for those factors were all 4.0 or above. Next, we examined the questionnaires to determine if the same individual respondents were responsible for these nonanswers. Only two individuals consistently failed to answer most of the variables for all four factors. Since these people rated their computer experience levels as four and five respectively, we were unable to derive any significant relationship between the respondents and the cause for nonanswer.

The objective of the questionnaire was to serve as a design tool to aid in establishing user interface design goals by identifying those user interface issues of special concern to the user in the performance of the tasks to be accomplished. In order to garner the desired information from the questionnaire, it was evident that the questionnaire analysis must proceed in a two-fold manner. First, of special interest to us, were those factors whose index fell below the average of 4.0. This would tend to indicate less than satisfactory design implementation. Second, to prevent being misled by numbers, we also took a close look at the response dispersal for every variable within each underlying factor to determine if an averaging process was distorting or hiding other problem areas.

In examining the results of the factor index, four factors and one interactive style possessed less than 4.0 indices. These were Instruction Descriptiveness, Error Correction, Learning, System Exploration, and Command Language, respectively. Based upon the individual variables comprising each of these factors, a clear relationship between Shneiderman's human factor goals measuring successful user interface design can be discerned. A less than satisfactory performance exists for three of the five human factor goals indicating distinct problems with the user interface design. Instruction Descriptiveness, Learning, and Command Language are all linked to Shneiderman's time

USER INTERFACE FACTOR	INDEX
Character Appearance	5.7
Highlighting Utilization	5.1
Display Layouts	5.0
Display Sequence	5.4
Interaction Pace	4.1
Terminology Relation	4.4
Terminology Use	5.3
Instruction Descriptiveness	3.9
Instruction Use	4.5
Operation Relation	4.0
Feedback	4.6
Error Messages	4.3
Error Correction	3.7
Learning	3.8
Experience Levels	4.5
Memory Limitations	4.4
Reference Materials	4.4
System Exploration	3.7
Overall Satisfaction	3.9
INTERACTIVE STYLES	INDEX
Menu Selection	N/A
Command Language	
Novice	3.5
Experienced	3.8
Menu & Command Language	N/A
Data Entry	
Novice	4.5
Experienced	4.6
Respondents' Computer Experience Level	3.8

Figure 12. Factor Index of User Interface Results from Questionnaire

to learn and retention over time human factor goals; Error Correction and System Exploration are linked to rate of errors by users.

In examining the individual variables of the factors, what we searched for in particular, were variables where the ratings were not centrally clustered around a 2 to 3 degree spread, but instead, were widely dispersed over a 4 to 5 degree range that leaned toward the negative side. We reasoned that this disparity of opinions amongst the respondents

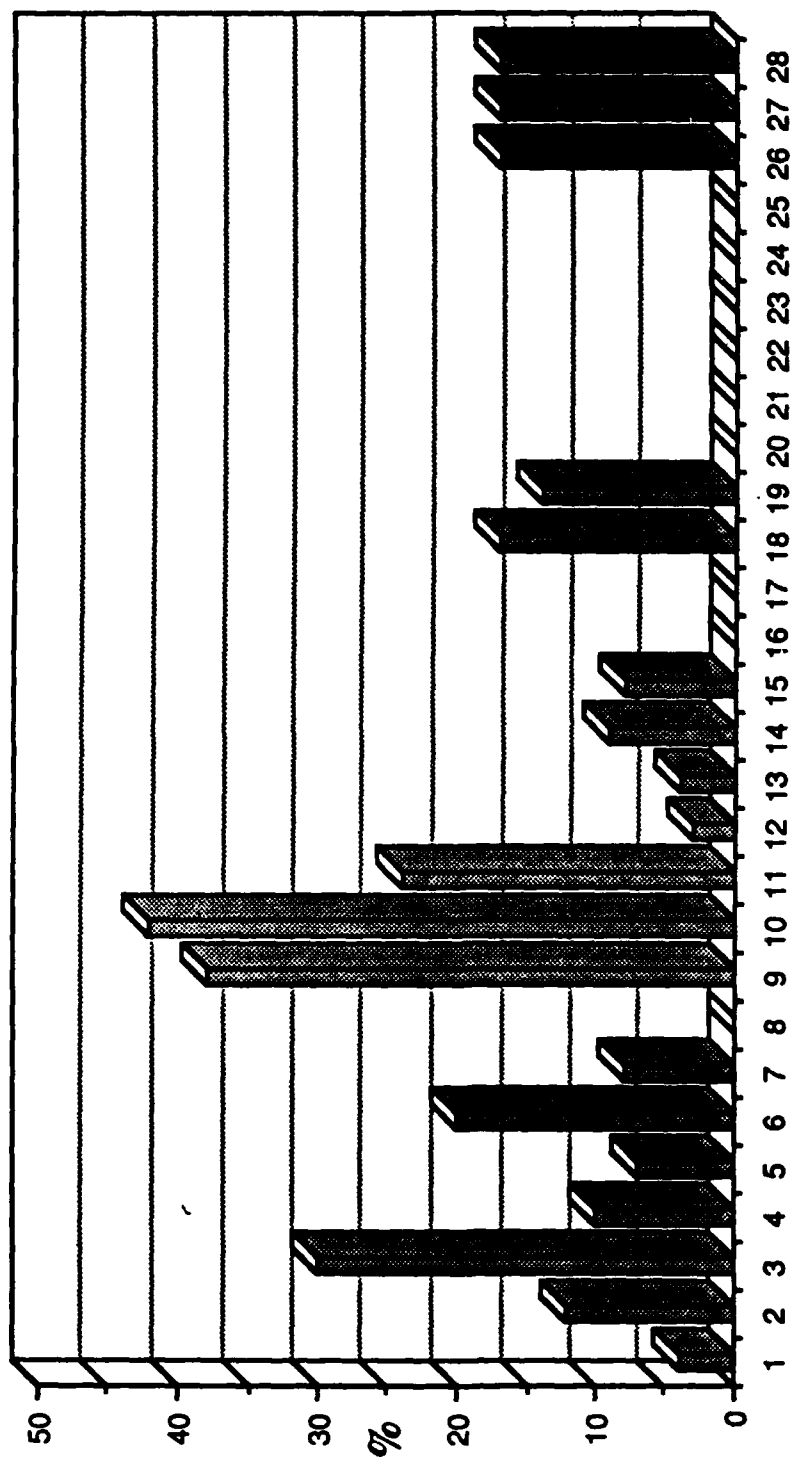


Figure 13. Nonresponse Rate by Factor

is a flag indicating that the applicable factor is in fact causing serious problems for at least some of the users. Figure 14 on page 36 contains a list of the factors which contained more than one variable with widely scattered ratings. These were Instruction Descriptiveness, Error Messages, Error Correction, Experience Levels, Memory Limitations, Reference Materials, System Exploration, and Data Entry. The recurrence of three factors, Instruction Descriptiveness, Error Correction, and System Exploration, from the previous findings clearly reinforces the indication that serious problems exist with these aspects of the user interface. The factors again relate to a less than satisfactory performance of Shneiderman's human factor goals identified above.

All of the factors identified implied deficiencies within the dialog design of the user interface. Most of these deficiencies could be rectified by offering more than one interactive style. The NCPDS system is driven by command language only. It does not even offer an on-screen form fill-in option. Therefore, based on the above information, it was decided that the user interface design must include menu selection, command language, and an on-screen form fill-in option to accommodate novice, intermittent, and frequent users. As cited by Shneiderman in Figure 15 on page 37, the advantages listed under these interactive styles address the factors creating most of the user dissatisfaction with the user interface.

C. COMPOSITE DESIGN

An analysis of the results above led us to the conclusion that the design and implementation of a relational database system using both menu selection and on-screen form fill-in interactive styles would be appropriate for a prototype of the application system. (See Appendix D.)

In order to minimize functional complexity at this time, the prototype is a scaled down version of the anticipated final system. Its primary purpose is to aid the final stages of detailed design in determining the user's specific preferences in system dialog and display layouts. It was hoped that if we were successful in the design of the menu selection, the display and selection mechanisms might prove fast enough to satisfy experienced users and thus eliminate the requirement to implement the command language option.

A simple tree structured menu was implemented where functionally similar tasks were partitioned into groups at logically equivalent levels. Terminology from the user's task domain was used to orient the user and aid the decision-making process. Menu items were ordered according to most frequently used items first. Items became menu

USER INTERFACE FACTOR

Instruction Descriptiveness

- 36. Of tasks
- 37. For commands or choices
- 38. Correcting errors
- 39. For getting help

Error Messages

- 53. Helpful
- 54. Clarify problems
- 55. Indicate required actions
- 56. Are specific

Error Correction

- 59. Of typos or slips
- 60. To change previous value
- 61. To undo operations

Experience Levels

- 66. Accommodates different levels
- 67. Accommodates novices

Memory Limitations

- 71. To complete tasks
- 72. Information patterns

Reference Materials

- 73. Supplemental
- 74. Manuals

System Exploration

- 75. Of features
- 76. Destructive Operations
- 77. Meaningful prompts

INTERACTIVE STYLE

Data Entry

- 6. Errors made as novice
- 7. Ability to remember
- 11. Entry speed

Figure 14. Factors Containing Widely Scattered Ratings

titles as one proceeded down the tree. Where applicable, menu layout and terminology was consistent.

Form fill-in is usually a preferred approach by most subjects over command language when updating a database [Ref. 38: p. 542]. Therefore, form fill-in was utilized for all database maintenance. Form titles, field labels, and other terminology were again

Advantages	Disadvantages
Menu Selection shortens learning reduces keystrokes structures decision-making permits use of dialog management tools	danger of many menus may slow frequent users consumes screen space requires rapid display rate
Form Fill-In simplifies data entry requires modest training assistance is convenient permits use of form management tools	consumes screen space
Command Language flexibility appeals to "power" users supports user initiative convenient for creating user defined macros	poor error handling requires substantial training and memorization
Natural Language relieves burden of learning syntax	requires clarification dialog may require more keystrokes may not show context unpredictable
Direct Manipulation visually presents task concepts easy to learn easy to retain errors can be avoided encourages exploration high subjective satisfaction	may be hard to program may require graphics display and pointing devices

Figure 15. Interactive Styles: [Ref. 28: p.].

based on the user's task domain. Since data elements had been logically divided into five relations, each display reflected the elements within a relation. An attempt was made to keep display layouts uncluttered with the input fields occurring in a natural ordered sequence. All instructions consistently appeared in the bottom of the screen. Since the system was a prototype, data input error traps and help facilities have not been implemented at this time.

Utilizing test data for over 20 training events, basic performance tests were run on the prototype to evaluate the following system performance criteria: usability, reliability, response time, and user-friendliness. The system, though not fully implemented, was clearly capable of supporting the desired functions of the Training Director. The system also had the potential to support management information systems (MIS) reports for management personnel within the CPO environment through ad hoc queries and generation of various standard reports. At present, the only system reliability problem involved accurate input of data input since minimum error traps exist. However, once data input error traps are added to the system, this problem will be eliminated.

Our problems began with the response time. While the response time for record maintenance is good, a potential problem for serious response time degradation exists with report and ad hoc queries that require join operations. The response time using the test data were good; however, the Training Director expects to monitor over 400 training events in a given fiscal year. These events are not uniformly distributed throughout the year, but are "front loaded" at the beginning of the fiscal year and at the beginning of each quarter. As the storage of records increases, the response times for the report and the query functions of the system will begin to deteriorate since most major reports and queries involve more than one join operation. Though most reports are run only on a monthly or quarterly basis, this could still pose as a potential threat to eventual user rejection of the system.

The next problems were all linked to the user interface. While the system was user-friendly, the use of the dBase III Plus application generator (APPSGEN) prevented the system from achieving good system transparency. To prevent duplicate entries of certain records, the user was requested to enter key information. If no previous record was found, an empty display form was brought up on the screen. The user would then have to reenter this information via the display. This problem and a few more like it could be overcome by hand coding vice using the APPSGEN. The second user interface problem involved the form fill-in screens. Since the screens had been based on the underlying data relation, they did not duplicate the exact order of items as they appeared on DD Form 1556, the input document. The user indicated that this might cause some problems with data entry since junior clerks would most likely be performing data input. It was requested that the input screens more closely follow DD Form 1556. This substantiates what we found in research literature that regardless of how poor the source document format is screen design should emulate the document as closely as possible.

The user also requested that the record maintenance option on the main menu be decomposed into its composite options at the main menu level for better visibility of available options.

Based upon the above problems and observations, a second prototype was developed to accommodate both a redesign of the database, and thus eliminate the need for the join operations, and a redesign of the input screens. (See Appendix E.)

V. CONCLUSIONS AND RECOMMENDATIONS

A. CONCLUSIONS

The main goal originally established for this project was to design a prototype for a "user-friendly" interactive system which could monitor and maintain civilian personnel training records for training directors at any CPO. Based upon this design, a prototype of the system was implemented to test and refine the design as necessary.

During the process of defining the user problem, it became apparent that a two-fold approach to the system design was required to effectively accommodate both the functional application and the user interface components. Standard software engineering structure analysis and design techniques were utilized to adequately define the functional application specifications. However, lacking any standard framework for the user interface design, a research design approach was developed and investigated as a means of conducting the analysis of the user interface requirements. Prototyping, which is more typically used when dealing with uncertainty in analysis and design, was rejected as the sole means of defining user interface requirements due to time constraints. The objective of the research design approach was to develop a methodology that reduced the time and costs involved in the iterative process required by prototyping alone.

It was decided that a more immediate and direct approach to establishing user interface goals would be to have a group of perspective typical users tell us what user interface factors affected their satisfaction with a system. Unable to obtain cooperation from any CPO other than the local CPO at NPS, we were left to draw upon only those employees within the local CPO who had some experience using interactive systems. Since the users were already familiar with the NCPDS system, it was decided to use that system as a bench mark for evaluating the users' preferences since it would help them to focus and understand the questions better. A questionnaire was selected as the method of data collection since it was the easiest, least costly, and least disruptive means of acquiring the information. It was also hoped that the questionnaire would draw out system inadequacies that people would normally be reluctant to admit to because they feel it might reflect their own shortcomings vice the system's. Through the process of performing personal interviews and reviewing literary research, 28 specific user interface factors that affected user satisfaction were identified. An existing questionnaire found during literary research was modified for the study. Since modifications to the

questionnaire were deemed insignificant, the questionnaire was not tested for content validity or reliability. Of the ten people identified for the study, only six completed the questionnaire.

The questionnaire was analyzed from two perspectives. A factor index was constructed to empirically represent the group's ratings of various factors of the user interface. If an index fell below the average 4.0, it was flagged as having a less than satisfactory implementation. Then, to identify other potential problem areas, a close inspection of the response dispersal for every variable within each factor was conducted to determine if an averaging process was distorting a less than ideal implementation.

Based upon the results of the above analyses, it was concluded that the design and implementation of a relational database system using both menu selection and on-screen form fill-in interactive styles would be appropriate for a prototype of the application system. However, after an analysis of the prototype, it became apparent that while individual analysis arrived at an ideal design for each separate component, the independent approach to implementing the system had failed to recognize certain interrelationships between functional design and user interface design that were created by limitations of the software required to implement the system. A second prototype rectifying the deficiencies of the first was then successfully implemented, and thus accomplishing the primary objective of this study.

An important conclusion of this study is the observation that while interactive system development should commence with independent designs of both the functional application and the user interface, recognition of the interrelationships that might be created by the implementation environment can have a significant impact on the quality of system performance and must be thoroughly investigated before final system design. Specifically where system implementation is restricted to a particular hardware or software environment, then all potential constraints imposed by that ADP environment must be identified. Preliminary design alternatives can then be evaluated in view of these limitations, and based on prioritized performance criteria, unsuccessful designs can be identified and eliminated.

B. RECOMMENDATIONS

This study additionally provided an innovative framework for establishing specific user interface design goals for a particular subject group which relied heavily on user involvement to learn from their corporate past experiences. Though we feel the basic premise of the methodology is sound, there are several strongly recommended strategies

for implementing the questionnaire that may help avoid the pitfalls we encountered and result in a higher degree of successful fact gathering.

1. A higher user interest must be stimulated in the significance of participating in the survey by either direct management intervention or sponsor presentation of the potential benefits.

2. A revalidation of the questionnaire for every user group must be conducted to ensure subjects understanding of the questions and hopefully eliminate the high nonresponse rate to certain questions.

3. When employing a long questionnaire form, a summary check-off list of sections required to be completed should be included to ensure no sections are inadvertently omitted.

4. Provide a glossary of computer terminology definitions to minimize inconsistent interpretations.

5. Present the questionnaire to subjects as a group in order to provide background brief regarding objectives of the study and the importance of full participation.

6. In conjunction with the questionnaire, a simulation of new and potentially more practical user interface techniques should be demonstrated to broaden users' awareness of alternatives.

APPENDIX A. FACTORS USED TO EVALUATE USER INTERFACE

User Interface Factor

1. Character Appearance
2. Highlighting Utilization
3. Display Layouts
4. Display Sequence
5. Interaction Pace
6. Terminology Relation
7. Terminology Use
8. Instruction Descriptiveness
9. Instruction Use
10. Operation Relation
11. Feedback
12. Error Messages
13. Error Correction
14. Learning
15. Experience Levels
16. Memory Limitations
17. Reference Materials
18. System Exploration
19. Overall Satisfaction

Interactive Styles

- Menu Selection
20. Novice
21. Experienced
- Command Language
22. Novice
23. Experienced
- Menu & Command Language
24. Novice
25. Experienced
- Data Entry
26. Novice
27. Experienced
28. Respondents' Computer Experience Level

APPENDIX B. APPLICATION SOFTWARE USER INTERFACE QUESTIONNAIRE

Please circle the number which best describes your impressions regarding the NCPDS computer system. (Additional comments are welcomed.)

Screen Appearance

Non applicable = NA

- | | |
|--|--|
| <p>1. Characters in the displays are</p> <p>_____</p> <p>_____</p> <p>_____</p> | <p style="text-align: center;">unreadable readable</p> <p style="text-align: center;">1 2 3 4 5 6 7 NA</p> |
| <p>2. Character definition is</p> <p>_____</p> <p>_____</p> <p>_____</p> | <p style="text-align: center;">fuzzy sharp</p> <p style="text-align: center;">1 2 3 4 5 6 7 NA</p> |
| <p>3. Character contrast with background is</p> <p>_____</p> <p>_____</p> <p>_____</p> | <p style="text-align: center;">poor excellent</p> <p style="text-align: center;">1 2 3 4 5 6 7 NA</p> |
| <p>4. Character shapes (fonts) are</p> <p>_____</p> <p>_____</p> <p>_____</p> | <p style="text-align: center;">unreadable readable</p> <p style="text-align: center;">1 2 3 4 5 6 7 NA</p> |
| <p>5. Highlighting facilitates task</p> <p>_____</p> <p>_____</p> <p>_____</p> | <p style="text-align: center;">poorly very well</p> <p style="text-align: center;">1 2 3 4 5 6 7 NA</p> |
| <p>6. Levels of intensity or boldfacing is</p> <p>_____</p> <p>_____</p> <p>_____</p> | <p style="text-align: center;">hard to see clear</p> <p style="text-align: center;">1 2 3 4 5 6 7 NA</p> |
| <p>7. Letter or shape size changes are</p> <p>_____</p> <p>_____</p> <p>_____</p> | <p style="text-align: center;">hard to see clear</p> <p style="text-align: center;">1 2 3 4 5 6 7 NA</p> |
| <p>8. Use of underscoring is</p> <p>_____</p> <p>_____</p> <p>_____</p> | <p style="text-align: center;">inappropriate appropriate</p> <p style="text-align: center;">1 2 3 4 5 6 7 NA</p> |

9. Use of reverse video is

inappropriate appropriate
1 2 3 4 5 6 7 NA

10. Use of blinking is

inappropriate appropriate
1 2 3 4 5 6 7 NA

11. Use of color changes is

inappropriate appropriate
1 2 3 4 5 6 7 NA

Display Layouts

12. Display layouts simplify tasks

never always
1 2 3 4 5 6 7 NA

13. Displays are

cluttered uncluttered
1 2 3 4 5 6 7 NA

14. Displays are

disorganized organized
1 2 3 4 5 6 7 NA

15. A title identifies the display

never always
1 2 3 4 5 6 7 NA

16. Work proceeds from top to bottom

never always
1 2 3 4 5 6 7 NA

17. Sequence of displays are

confusing clear
1 2 3 4 5 6 7 NA

18. Next screen in a sequence is

unpredictable predictable
1 2 3 4 5 6 7 NA

19. Maintaining a sense of position is

 _____ impossible easy
 1 2 3 4 5 6 7 NA
20. Going back to a previous display is

 _____ impossible easy
 1 2 3 4 5 6 7 NA
21. Beginnings, middles, and ends of tasks
 are marked

 _____ confusingly clearly
 1 2 3 4 5 6 7 NA
22. Pace of system interaction is

 _____ too slow too fast
 1 2 3 4 5 6 7 NA
23. Data entry operations are echoed on
 the screen

 _____ too slowly fast enough
 1 2 3 4 5 6 7 NA
24. Response time for most operations is

 _____ too slow fast enough
 1 2 3 4 5 6 7 NA
25. Error messages appear

 _____ too slow fast enough
 1 2 3 4 5 6 7 NA
26. Display rate for most displays is

 _____ too slow fast enough
 1 2 3 4 5 6 7 NA

Screen Dialog Design

27. Terminology relates to the task

 _____ distantly closely
 1 2 3 4 5 6 7 NA
28. Computer-related terms are used

 _____ too frequently appropriately
 1 2 3 4 5 6 7 NA

29. Task-related terms are used

 never 1 2 3 4 5 6 7 NA
30. Terms on the screen are

 ambiguous 1 2 3 4 5 6 7 NA
 precise
31. Abbreviations used are

 confusing 1 2 3 4 5 6 7 NA
 clear
32. Terminology used on the screen is

 inconsistent 1 2 3 4 5 6 7 NA
 consistent
33. Task terms used are

 inconsistent 1 2 3 4 5 6 7 NA
 consistent
34. Computer terms used are

 inconsistent 1 2 3 4 5 6 7 NA
 consistent
35. Abbreviations used are

 inconsistent 1 2 3 4 5 6 7 NA
 consistent
36. Instructions describing tasks are

 confusing 1 2 3 4 5 6 7 NA
 clear
37. Instructions for commands or choices are

 confusing 1 2 3 4 5 6 7 NA
 clear
38. Instructions for correcting errors are

 confusing 1 2 3 4 5 6 7 NA
 clear
39. Instructions for getting more help are

 confusing 1 2 3 4 5 6 7 NA
 clear

- | | | | | | | | | |
|--|-----------|---|---|---|---|----------|---|----|
| 40. Consistent instructions are used | never | | | | | always | | |
| _____ | 1 | 2 | 3 | 4 | 5 | 6 | 7 | NA |
| _____ | | | | | | | | |
| 41. Instructions consistently have the same position | never | | | | | always | | |
| _____ | 1 | 2 | 3 | 4 | 5 | 6 | 7 | NA |
| _____ | | | | | | | | |
| 42. Instructions use consistent grammar | never | | | | | always | | |
| _____ | 1 | 2 | 3 | 4 | 5 | 6 | 7 | NA |
| _____ | | | | | | | | |
| 43. Instructions use consistent tone | never | | | | | always | | |
| _____ | 1 | 2 | 3 | 4 | 5 | 6 | 7 | NA |
| _____ | | | | | | | | |
| 44. Operations relate to tasks | distantly | | | | | closely | | |
| _____ | 1 | 2 | 3 | 4 | 5 | 6 | 7 | NA |
| _____ | | | | | | | | |
| 45. Number of operations per task are | many | | | | | few | | |
| _____ | 1 | 2 | 3 | 4 | 5 | 6 | 7 | NA |
| _____ | | | | | | | | |
| 46. Operations related to tasks are | obscure | | | | | clear | | |
| _____ | 1 | 2 | 3 | 4 | 5 | 6 | 7 | NA |
| _____ | | | | | | | | |
| 47. Operations prevent mistakes | never | | | | | always | | |
| _____ | 1 | 2 | 3 | 4 | 5 | 6 | 7 | NA |
| _____ | | | | | | | | |
| 48. Informative feedback is appropriate | never | | | | | always | | |
| _____ | 1 | 2 | 3 | 4 | 5 | 6 | 7 | NA |
| _____ | | | | | | | | |
| 49. Link between operations and results are | confusing | | | | | clear | | |
| _____ | 1 | 2 | 3 | 4 | 5 | 6 | 7 | NA |
| _____ | | | | | | | | |
| 50. Amount of feedback is | too much | | | | | adequate | | |
| _____ | 1 | 2 | 3 | 4 | 5 | 6 | 7 | NA |
| _____ | | | | | | | | |

51. Amount of feedback is	too little				adequate			
_____	1	2	3	4	5	6	7	NA

52. Amount of feedback is user-controlled	never				always			
_____	1	2	3	4	5	6	7	NA

53. Error messages are helpful	never				always			
_____	1	2	3	4	5	6	7	NA

54. Error messages clarify the problem	never				always			
_____	1	2	3	4	5	6	7	NA

55. Error messages indicate actions to be taken	never				always			
_____	1	2	3	4	5	6	7	NA

56. Error messages are specific	never				always			
_____	1	2	3	4	5	6	7	NA

57. Error messages are	nasty				pleasing			
_____	1	2	3	4	5	6	7	NA

58. Error correction is	confusing				clear			
_____	1	2	3	4	5	6	7	NA

59. Correcting typos or complex slips is	complex				simple			
_____	1	2	3	4	5	6	7	NA

60. Going back to change values is	complex				simple			
_____	1	2	3	4	5	6	7	NA

61. Undoing operations is	complex				simple			
_____	1	2	3	4	5	6	7	NA

Human Factor Orientation

62. Learning how to operate the system is

difficult easy
1 2 3 4 5 6 7 NA

63. Getting started is

difficult easy
1 2 3 4 5 6 7 NA

64. Learning more features is

difficult easy
1 2 3 4 5 6 7 NA

65. Relearning after intermittent use is

difficult easy
1 2 3 4 5 6 7 NA

66. Use by different levels of experience is

not accommodated accommodated
1 2 3 4 5 6 7 NA

67. Novices can use the system

with difficulty conveniently
1 2 3 4 5 6 7 NA

68. Experts can add features/shortcuts

with difficulty conveniently
1 2 3 4 5 6 7 NA

69. User can tailor the interface

with difficulty conveniently
1 2 3 4 5 6 7 NA

70. Human memory limitations are

overwhelmed are respected
1 2 3 4 5 6 7 NA

71. Information to complete tasks

must be memorized is visible
1 2 3 4 5 6 7 NA

- _____
- _____
- _____

1	2	3	4	5	6	7	NA
---	---	---	---	---	---	---	----

- _____
- _____
- _____

confusing				clear			
1	2	3	4	5	6	7	NA

- _____
- _____
- _____

confusing				clear				
1	2	3	4	5	6	7	NA	

- _____
- _____
- _____

discouraged encouraged
1 2 3 4 5 6 7 NA

- _____
- _____
- _____

not recoverable				recoverable			
1	2	3	4	5	6	7	NA

- _____
- _____
- _____

not provided				provided			
1	2	3	4	5	6	7	NA

- _____
- _____
- _____

terrible				wonderful			
1	2	3	4	5	6	7	NA

frustrating				satisfying				
1	2	3	4	5	6	7	NA	

dull stimulating
1 2 3 4 5 6 7 NA

difficult				easy				
1	2	3	4	5	6	7	NA	

inadequate power				adequate power			
1	2	3	4	5	6	7	NA

INTERACTIVE STYLES

Menu Selection

1. Is your system menu driven? yes no

(If you answer no, please skip to the Command Language section.)

When you first began using the system:

2. Learning was
_____ difficult easy
_____ 1 2 3 4 5 6 7 NA

3. Time required to learn was
_____ extensive modest
_____ 1 2 3 4 5 6 7 NA

4. How to operate the system was
_____ confusing clear
_____ 1 2 3 4 5 6 7 NA

5. Option descriptions in the menu were
_____ obscure meaningful
_____ 1 2 3 4 5 6 7 NA

6. Errors made while operating the
system were
_____ frequent infrequent
_____ 1 2 3 4 5 6 7 NA

After you had been using the system for an
extended period of time:

7. Ability to remember how to use the
system was
_____ easy recall not easy recall
_____ 1 2 3 4 5 6 7 NA

8. Speed of operating the system was
_____ too slow fast enough
_____ 1 2 3 4 5 6 7 NA

9. Errors made while operating the system were

frequent infrequent
1 2 3 4 5 6 7 NA

Command Language

1. Is your system command driven? yes no

(If you answer no, please skip to the Menu Selection and Command Language section.)

When you first began using the system:

2. Learning was

difficult easy
1 2 3 4 5 6 7 NA

3. Time required to learn was

extensive modest
1 2 3 4 5 6 7 NA

4. How to operate the system was

confusing clear
1 2 3 4 5 6 7 NA

5. Command language was

obscure meaningful
1 2 3 4 5 6 7 NA

6. Errors made while operating the system were

frequent infrequent
1 2 3 4 5 6 7 NA

After you had been using the system for an extended period of time:

7. Ability to remember how to use the system was

easy recall not easy recall
1 2 3 4 5 6 7 NA

8. Speed of operating the system was _____

 too slow 1 2 3 4 fast enough 5 6 7 NA
9. Errors made while operating the system were _____

 frequent 1 2 3 4 infrequent 5 6 7 NA

Menu Selection and Command Language

1. Does your system allow you to chose between menu driven or command driven modes? yes no

(If you answer no, please skip to question 13 in this section.)

When you first began using the system:

2. You used the _____

 menu mode 1 2 3 4 command mode 5 6 7 NA
3. Learning was _____

 difficult 1 2 3 4 easy 5 6 7 NA
4. Time required to learn was _____

 extensive 1 2 3 4 modest 5 6 7 NA
5. How to operate the system was _____

 confusing 1 2 3 4 clear 5 6 7 NA
6. Task terminology was _____

 obscure 1 2 3 4 meaningful 5 6 7 NA
7. The same task terminology was used for menu options and command language _____

 never 1 2 3 4 always 5 6 7 NA

8. Errors made while operating the system were

frequent infrequent
1 2 3 4 5 6 7 NA

After you had been using the system for an extended period of time:

9. You used the

menu mode command mode
1 2 3 4 5 6 7 NA

10. Ability to remember how to use the system was

easy recall not easy recall
1 2 3 4 5 6 7 NA

11. Speed of operating the system was

too slow fast enough
1 2 3 4 5 6 7 NA

12. Errors made while operating the system were

frequent infrequent
1 2 3 4 5 6 7 NA

(If you answered no to question 1 above, please answer question 13.)

13. Would you like your system to have the option to choose between menu driven and command driven modes? Why?

yes no

Data Entry

When you first began using the system:

1. Learning how to input data was

difficult easy
1 2 3 4 5 6 7 NA

2. Time required to learn was

extensive modest
1 2 3 4 5 6 7 NA

3. Actual data entry was

 difficult 1 2 3 4 5 6 7 NA
 easy
4. Actual data entry was

 confusing 1 2 3 4 5 6 7 NA
 clear
5. Actual data entry was

 frustrating 1 2 3 4 5 6 7 NA
 satisfactory
6. Errors made while entering the data were

 frequent 1 2 3 4 5 6 7 NA
 infrequent

After you had been using the system for an extended period of time:

7. Ability to remember how to enter the data was

 easy recall 1 2 3 4 5 6 7 NA
 not easy recall
8. Actual data entry was

 difficult 1 2 3 4 5 6 7 NA
 easy
9. Actual data entry was

 confusing 1 2 3 4 5 6 7 NA
 clear
10. Actual data entry was

 frustrating 1 2 3 4 5 6 7 NA
 satisfactory
11. Speed of entering the data was

 too slow 1 2 3 4 5 6 7 NA
 fast enough
12. Errors made while entering the data were

 frequent 1 2 3 4 5 6 7 NA
 infrequent

13. Does your system provide an on screen
form for data input?

yes no

14. If no, would you like your system to
provide an on screen form for data
input? Why?

yes no

15. I would rate myself as a computer

novice expert
1 2 3 4 5 6 7 NA

APPENDIX C. DATA DICTIONARY OF DATA ELEMENTS

data element name: LOGNUM
aliases: DD Form 1556 Log Number
type/width: character/5
values:
location:

data element name: FYCRS
aliases: fiscal year of course
type/width: character/2
values:
location:

data element name: MULEMPS
aliases: multiple employees in course
type/width: logical
values: Y/N
location:

data element name: DTREC1556
aliases: date received DD Form 1556
type/width: date
values: YYMMDD
location:

data element name: DTLTRAGR
aliases: date letter of agreement sent
type/width: date
values: YYMMDD
location:

data element name: DTCOPY002
aliases: date copy DD Form 1556 sent to 002
type/width: date
values: YYMMDD
location:

data element name: DTCOPYSCR
aliases: date copy DD Form 1556 sent to source
type/width: date
values: YYMMDD
location:

data element name: DTCOPYDEPT
aliases: date copy DD Form 1556 sent to department
type/width: date
values: YYMMDD
location:

data element name: DTEVALDEPT
 aliases: date evaluation sent to department
 type/width: date
 values: YYMMDD
 location:

data element name: COMPLETED
 aliases: course completed
 type/width: logical
 values: Y/N
 location: DD Form 1556 item 30

data element name: NCOMPLCODE
 aliases: course non-completion code
 type/width: character/1
 values: F - failed
 C - source cancelled course
 W - withdrew prior to start of course
 T - employee transferred
 I - incompleted by employee
 location: DD Form 1556 item 30

data element name: ELASTNAME
 aliases: employee last name
 type/width: character/20
 values:
 location: DD Form 1556 item 1

data element name: EFIRSTNAME
 aliases: employee first name
 type/width: character/10
 values:
 location: DD Form 1556 item 1

data element name: EMINITIAL
 aliases: employee middle initial
 type/width: character/1
 values:
 location: DD Form 1556 item 1

data element name: ESSN
 aliases: employee social security number
 type/width: character/9
 values:
 location: DD Form 1556 item 2

data element name: EUIC
 aliases: employee unit identification code
 type/width: character/5
 values:
 location: DD Form 1556 item 3a

data element name: ESUPVCODE
 aliases: employee supervisory code
 type/width: character/1
 values: 0 - supervises 1 or 2
 S - supervises 3 or +
 M - manager
 X - N/A
 location: DD Form 1556 item 6

data element name: EDEPTCODE
 aliases: employee department code
 type/width: character/5
 values:
 location: DD Form 1556 item 7

data element name: EFEDSERV
 aliases: employee length of federal service
 type/width: character/4
 values: YYYY
 location: DD Form 1556 item 9

data element name: ETITLE
 aliases: employee job title
 type/width: character/30
 values:
 location: DD Form 1556 item 11

data element name: EHANDICAP
 aliases: employee handicap
 type/width: logical
 values: Y/N
 location: DD Form 1556 item 11b

data element name: EPAYPLAN
 aliases: employee pay plan
 type/width: character/2
 values: SA - summer aide
 ST - student aide
 GS - general schedule
 GM - general merit
 AD - faculty
 WG - wage grade
 WS - wage supervisor
 location: DD Form 1556 item 12a

data element name: ESERIES
 aliases: employee occupational series
 type/width: character/4
 values:
 location: DD Form 1556 item 12b

data element name: EPAYGRADE
 aliases: employee pay grade
 type/width: character/2
 values:
 location: DD Form 1556 item 12c

data element name: EAPPTMT
aliases: employee appointment status
type/width: character/2
values: T1 - temporary NTE 1 year
T2 - temporary over 1 year
C1 - career conditional
C2 - career
location: DD Form 1556 item 13

data element name: ESCHEDULE
aliases: employee schedule
type/width: character/1
values: F - full time
P - part time
I - intermittent
location:

data element name: ESEX
aliases: employee sex
type/width: character/1
values: M/F
location:

data element name: ETHNICGRP
aliases: employee ethnic group
type/width: character/1
values: C - caucasian
B - black
N - native american
A - asian - pacific islander
H - hispanic
location:

data element name: SUIC
aliases: source unit identification code
type/width: character/5
values:
location: DD Form 1556 item 15a

data element name: SNAME
aliases: source name
type/width: character/40
values:
location: DD Form 1556 item 15a

data element name: SCITY
aliases: source city
type/width: character/20
values:
location: DD Form 1556 item 15a

data element name: SSTATE
aliases: source state
type/width: character/2
values:
location: DD Form 1556 item 15a

data element name: STYPE
aliases: source type code
type/width: character/2
values: GA - Army
GN - Navy
GP - Navy, NPS
GF - Air Force
GM - Marines
GG - Coast Guard
GO - OPM, regional
GI - NISC, correspondence course
GS - state
GC - county
LN - local non-government, non-profit
LP - local non-government, profit
OR - distant non-government, non-profit
OT - distant non-government, profit
location: DD Form 1556 item 20c

data element name: CCITY
aliases: course city
type/width: character/20
values:
location: DD Form 1556 item 15b

data element name: CSTATE
aliases: course state
type/width: character/2
values:
location: DD Form 1556 item 15b

data element name: TRNGTYPE
aliases: training type
type/width: character/1
values: U - basic supervisory
V - advanced supervisory
M - basic managerial
N - advanced managerial
S - safety
A - administrative/clerical
C - computers
I - interpersonal skills
E - equal employment opportunity
X - other
O - orientation
location: DD Form 1556 item 16a

data element name: CTITLE
 aliases: course title
 type/width: character/30
 values:
 location: DD Form 1556 item 16b

data element name: CNUMBER
 aliases: course number
 type/width: character/10
 values:
 location: DD Form 1556 item 17a

data element name: CSTARTDT
 aliases: course start date
 type/width: date
 values: YYMMDD
 location: DD Form 1556 item 18a

data element name: CENDDT
 aliases: course end date
 type/width: date
 values: YYMMDD
 location: DD Form 1556 item 18b

data element name: CDUTYHRS
 aliases: # of duty hours of course
 type/width: numeric/3
 values:
 location: DD Form 1556 item 19a

data element name: CNONDUTYHR
 aliases: # of non-duty hours of course
 type/width: numeric/3
 values:
 location: DD Form 1556 item 19b

data element name: CPURPOSE
 aliases: purpose of course attendance
 type/width: character/1
 values: 1 - improve performance
 2 - new equipment/technology
 3 - training plan
 location: DD Form 1556 item 20a

data element name: CPRIORITY
 aliases: priority of course
 type/width: character/1
 values: 1, 2, 3
 location: DD Form 1556 item 20e

data element name: CMETHOD
 aliases: method of course
 type/width: character/1
 values: C - correspondence
 R - classroom
 F - conference
 location: DD Form 1556 item 20g

data element name: ETRNGPLAN
 aliases: employee training plan
 type/width: character/1
 values: V - VRA appointment
 U - upward mobility appointment
 M - new manager
 S - new supervisor
 location: DD Form 1556 item 20h

data element name: CREASON
 aliases: reason for course selection
 type/width: character/1
 values: C - cost effective
 T - timeliness
 O - sole source (only one available)
 Q - quality of source
 location: DD Form 1556 item 20i

data element name: PAYMETHOD
 aliases: payment method
 type/width: character/1
 values: B - billing/invoice and TNA #
 T - advance on DD1610 and TOA #
 R - SF1164 and RVA #
 location:

data element name: DIRCOSTS
 aliases: total direct costs
 type/width: numeric/8
 values: 99999.99
 location: DD Form 1556 item 21a

data element name: FUNDSOURCE
 aliases: source of funds
 type/width: character/1
 values: R - research/reimbursables
 O - OPTAR
 location: DD Form 1556 item 21d

data element name: TANGONO
 aliases: TANGO number for direct costs
 type/width: character/7
 values:
 location: DD Form 1556 item 21e

data element name: JOBORDNO
aliases: job order number for direct costs
type/width: character/5
values:
location: DD Form 1556 item 22

data element name: INDIRCOSTS
aliases: total indirect costs
type/width: numeric/8
values: 99999.99
location: DD Form 1556 item 25a

data element name: TRAVORDNO
aliases: travel order number
type/width: character/5
values:
location: DD Form 1556 item 25d

data element name: GOVDEDUC
aliases: government deduction
type/width: logical
values: Y/N
location: .

data element name: INVOICENO
aliases: invoice number
type/width: character/10
values:
location:

data element name: DTINVOIRCV
aliases: date invoice received
type/width: date
values: YYMMDD
location:

data element name: DTINVOIFWD
aliases: date invoice forwarded to 002
type/width: date
values: YYMMDD
location:

APPENDIX D. PROTOTYPE: FIRST ITERATION

* Title: CPO.PRG
* Authors: LCDR SHARON SLOMINSKI, USN
* LT IVON YOUNG, SC, USN
* Date: 19 NOV 1987
*
* The purpose of this program is to oversee the maintenance
* of the data base system for civilian personnel training.
*
* This program is the main program and calls procedures
* BANNER, MAINTAIN, REPORTS, and QUERIES

SET COLOR TO W+/B+,W+/R, R+
CLEAR
SET TALK OFF
SET BELL OFF
SET STATUS ON
SET ESCAPE OFF
SET CONFIRM ON

SET PROCEDURE TO CALLS

* Call procedure BANNER to give title page and instructions
* to user.

DO BANNER

* Set menu and hold for user to make selection.

DO WHILE .T.

* ---Display menu options, centered on the screen.
* draw menu border and print heading
CLEAR
@ 2, 0 TO 13,79 DOUBLE
@ 3,13 SAY (C P O T R A I N I N G T R A C K I N G S Y S T E M)
@ 4,1 TO 4,78 DOUBLE
* ---display detail lines
@ 7,32 SAY (1. RECORD MAINTENANCE)
@ 8,32 SAY (2. PRINT REPORTS)
@ 9,32 SAY (3. AD HOC QUERIES)
@ 11, 32 SAY '0. EXIT'
STORE 0 TO selectnum
@ 13,33 SAY " select "
@ 13,42 GET selectnum PICTURE "9" RANGE 0,3
READ

* Case statement. Depending on user's selection of 0-3
* the correct procedure will be called.

DO CASE

* No case is called. User is exiting the CPO program.

```
CASE selectnum = 0
  SET TALK ON
  CLEAR ALL
  CLEAR
  @ 10,10 SAY 'HAVE YOU BACKED UP YOUR DATA FILES RECENTLY?'
  WAIT
  CLEAR
  RETURN
```

* Call procedure MAINTAIN

```
CASE selectnum = 1
  DO MAINTAIN
  SET CONFIRM OFF
  STORE ' ' TO wait_subst
  @ 23,0 SAY 'Press any key to continue...' GET wait_subst
  READ
  SET CONFIRM ON
```

* Call procedure REPORTS

```
CASE selectnum = 2
  DO REPORTS
  SET CONFIRM OFF
  STORE ' ' TO wait_subst
  @ 23,0 SAY 'Press any key to continue...' GET wait_subst
  READ
  SET CONFIRM ON
```

* Call procedure QUERIES

```
CASE selectnum = 3
  DO QUERIES
  SET CONFIRM OFF
  STORE ' ' TO wait_subst
  @ 23,0 SAY 'Press any key to continue...' GET wait_subst
  READ
  SET CONFIRM ON
```

ENDCASE

ENDDO T

RETURN

* EOF: CPO.PRG

PROCEDURE MAINTAIN

* Procedure MAINTAIN

*

* This procedure is called by CPO and is used to maintain the
* data base file. This procedure is a menu which allows the
* user to call one of five other procedures, i.e., ADD1556,
* EMPINFO, MOD1556, MODCSE, and MODSOURC.

SET TALK OFF

SET BELL OFF

SET STATUS ON
SET ESCAPE OFF
SET CONFIRM ON

* Display menu options, centered on the screen.
* draw menu border and print heading

DO WHILE .T.

CLEAR

@ 2, 0 TO 15,79 DOUBLE

@ 3,18 SAY (R E C O R D M A I N T E N A N C E M E N U)

@ 4,1 TO 4,78 DOUBLE

* ---display detail lines

@ 7,26 SAY (1. ADD A 1556 RECORD)

@ 8,26 SAY (2. MODIFY EMPLOYEE INFORMATION)

@ 9,26 SAY (3. MODIFY 1556 INFORMATION)

@ 10,26 SAY (4. MODIFY COURSE INFORMATION)

@ 11,26 SAY (5. MODIFY SOURCE INFORMATION)

@ 13, 26 SAY '0. EXIT'

STORE 0 TO selectnum

@ 15,33 SAY " select "

@ 15,42 GET selectnum PICTURE "9" RANGE 0,5

READ

DO CASE

* No procedure is call. Allows user to exit.

CASE selectnum = 0

SET TALK ON

CLEAR ALL

RETURN

* Call procedure ADD1556

CASE selectnum = 1

DO ADD1556

SET CONFIRM OFF

STORE ' ' TO wait_subst

@ 23,0 SAY 'Press any key to continue...' GET wait_subst

READ

SET CONFIRM ON

* Call procedure EMPINFO

CASE selectnum = 2

DO EMPINFO

SET CONFIRM OFF

STORE ' ' TO wait_subst

@ 23,0 SAY 'Press any key to continue...' GET wait_subst

READ

SET CONFIRM ON

* Call procedure MOD1556

CASE selectnum = 3

DO MOD1556

```

        SET CONFIRM OFF
        STORE ' ' TO wait_subst
        @ 23,0 SAY 'Press any key to continue...' GET wait_subst
        READ
        SET CONFIRM ON

* Call procedure MODCSE

        CASE selectnum = 4
        DO MODCSE
        SET CONFIRM OFF
        STORE ' ' TO wait_subst
        @ 23,0 SAY 'Press any key to continue...' GET wait_subst
        READ
        SET CONFIRM ON

* Call procedure MODSOURC

        CASE selectnum = 5
        DO MODSOURC
        SET CONFIRM OFF
        STORE ' ' TO wait_subst
        @ 23,0 SAY 'Press any key to continue...' GET wait_subst
        READ
        SET CONFIRM ON
ENDCASE

ENDDO T
RETURN
* EOF: Procedure MAINTAIN

PROCEDURE ADD1556

* Procedure ADD1556
*
* This procedure is called by procedure MAINTAIN and allows
* the user to add records to the data base files INTERSEC
* and FORM1556.
*
* This procedure calls procedures EMPINFO1, MODCSE1, and
* MODS01

SET TALK OFF
SET BELL OFF
SET STATUS ON
SET ESCAPE OFF
SET CONFIRM ON

* DO WHILE loop to verify that the log number being entered
* has not been used before.

MDUP = "FALSE"
DO WHILE MDUP = "FALSE"
    CLEAR
    MLOGNUM = "      "
    @ 10,10 SAY 'ENTER LOG NUMBER: ' GET MLOGNUM

```

```

READ
USE INTERSEC
LOCATE FOR LOGNUM =MLOGNUM
IF FOUND ( )
    @ 12,10 SAY "LOG NUMBER ALREADY EXISTS.  ENTER A NEW LOG NUMBER."
    WAIT
ELSE
    MDUP = "TRUE"  ENDIF
ENDDO

* Adds new record to file INTERSEC.

SET FORMAT TO INTERSEC
APPEND
USE
CLEAR

* Adds new record to file FORM1556.

USE FORM1556
SET FORMAT TO MOD1556
APPEND
USE
CLEAR

*      Display menu options, centered on the screen.
*      draw menu border and print heading

DO WHILE .T.
    CLEAR
    @ 2, 0 TO 13,79 DOUBLE
    @ 3,16 SAY (A D D I N G   A   F O R M   1 5 5 6   R E C O R D)
    @ 4,1 TO 4,78 DOUBLE
    * ---display detail lines
    @ 7,28 SAY (1. FORM 1556 EMPLOYEE INFO)
    @ 8,28 SAY (2. FORM 1556 COURSE INFO)
    @ 9,28 SAY (3. FORM 1556 SOURCE INFO)
    @ 11, 28 SAY '0. EXIT'
    STORE 0 TO selectnum
    @ 13,33 SAY " select      "
    @ 13,42 GET selectnum PICTURE "9" RANGE 0,3
    READ

    DO CASE

* No procedure is called.  Allows user to exit.

        CASE selectnum = 0
            SET TALK ON
            CLEAR ALL
            RETURN

* Calls procedure EMPINFO1

        CASE selectnum = 1
            DO EMPINFO1
            SET CONFIRM OFF

```

```

        STORE ' ' TO wait_subst
        @ 23,0 SAY 'Press any key to continue...' GET wait_subst
        READ
        SET CONFIRM ON

* Calls procedure MODCSE1

        CASE selectnum = 2
        DO MODCSE1
        SET CONFIRM OFF
        STORE ' ' TO wait_subst
        @ 23,0 SAY 'Press any key to continue...' GET wait_subst
        READ
        SET CONFIRM ON

* Calls procedure MODS01

        CASE selectnum = 3
        DO MODS01
        SET CONFIRM OFF
        STORE ' ' TO wait_subst
        @ 23,0 SAY 'Press any key to continue...' GET wait_subst
        READ
        SET CONFIRM ON
ENDCASE

ENDDO T
RETURN
* EOF: Procedure ADD1556

PROCEDURE EMPINFO1

* Procedure EMPINFO1
*
* This procedure is called by procedure ADD1556 to add data to
* the data base file EMPLOYEE.

CLEAR
* Allows user to enter SSN.

MSSN = ' '
@10,10 SAY 'ENTER EMPLOYEE SSN: ' GET MSSN
READ

* Searches file EMPLOYEE for SSN.  If found brings data to screen
* for editing/verification.

USE EMPLOYEE
LOCATE FOR ESSN = MSSN
IF FOUND()
    CLEAR
    SET FORMAT TO EMPLOY
    EDIT
    USE
    CLEAR
ELSE

```

* If not found, states so and allows user to add a new record.

```
CLEAR
@10,10 SAY 'EMPLOYEE SSN WAS NOT FOUND. THIS IS NEW INFORMATION.'
?
WAIT
CLEAR
SET FORMAT TO EMPLOYEE
APPEND
USE
CLEAR
ENDIF
RETURN
```

*EOF: Procedure EMPINF01

PROCEDURE MODCSE1

* Procedure MODCSE1

*
* This procedure is called by procedure ADD1556. This procedure
* use a composite key (CNUMBER, CSTARTDT) to search the data
* base file COURSE. If the composite key is found, the user is
* allowed to edit the record of file. If not found, the user
* is allowed to add a new record.

CLEAR

* Allows user to enter course number and course start date.

```
MCSNO = ' '
MSTARTDT = ' '
@10,10 SAY 'ENTER COURSE NUMBER: ' GET MCSNO
@12,10 SAY 'ENTER COURSE START DATE (MM/DD/YY): ' GET MSTARTDT
READ
```

* Uses file COURSE to search for composite key. If found, allows
* the user to edit the record.

```
USE COURSE
LOCATE FOR CNUMBER = MCSNO .AND. CSTARTDT = CTOD(MSTARTDT)
IF FOUND()
    CLEAR
    SET FORMAT TO COURSE1
    EDIT
    USE
    CLEAR
ELSE
```

* If not found, states so and allows user to add a new record.

```
CLEAR
@10,10 SAY 'COURSE NUMBER WAS NOT FOUND. THIS IS NEW INFORMATION.'
?
WAIT
```

```
CLEAR
SET FORMAT TO COURSE
APPEND
USE
CLEAR
ENDIF
RELE MCSNO, MSTARTDT
```

```
RETURN
```

```
*EOF: Procedure MODCSE1
```

```
PROCEDURE MODS01
```

```
* Procedure MODS01
```

```
*
```

```
* This procedure is called by procedure ADD1556 and allows user  
* to either add or edit a record in data base file SOURCE.
```

```
CLEAR
```

```
* Allows user to enter source unit identification code.
```

```
MSUIC = '  
@10,10 SAY 'ENTER SOURCE UIC: ' GET MSUIC  
READ
```

```
* Searches file SOURCE for SUIC. If found allows user to  
* edit the record.
```

```
USE SOURCE  
LOCATE FOR SUIC = MSUIC  
IF FOUND()  
  CLEAR  
  SET FORMAT TO MODSOURC  
  EDIT  
  USE  
  CLEAR  
ELSE
```

```
* If not found, allows user to add a new record.
```

```
  CLEAR  
  @10,10 SAY 'SOURCE UIC WAS NOT FOUND. THIS IS NEW INFORMATION.'  
  ?  
  WAIT  
  CLEAR  
  SET FORMAT TO MODS01  
  APPEND  
  USE  
  CLEAR  
ENDIF  
RETURN
```

```
*EOF: Procedure MODS01
```

PROCEDURE EMPINFO

* Procedure EMPINFO

*

* This procedure is called by procedure MAINTAIN and is used to
* do a check to see if the employee's SSN is currently on
* file in data base file EMPLOYEE. If so, allows the user
* to edit.

CLEAR

* A check to see if SSN is on file.

MSSN = " "

@ 10,10 SAY 'ENTER EMPLOYEE SSN: ' GET MSSN

READ

USE EMPLOYEE

* If SSN is found, the data will be placed on the screen for
* verification and/or updating.

LOCATE FOR ESSN = MSSN

IF FOUND()

CLEAR

SET FORMAT TO EMPLOY

EDIT

USE

CLEAR

ELSE

* If SSN isn't found, state such, and return to procedure
* MAINTAIN.

@15,10 SAY 'EMPLOYEE SSN NOT FOUND.'

?

?

?

?

ENDIF

RETURN

* EOF: Procedure EMPINFO

PROCEDURE MOD1556

* Procedure MOD1556

*

* This procedure is called by procedure MAINTAIN and uses
* the data base file FORM1556. The procedure searches for
* the log number and if found allows user to edit. If
* it isn't found, states so and returns to procedure
* MAINTAIN.

CLEAR

* Asks for log number and reads same.

```

MLOGNUM = "      "
@10,10 SAY 'ENTER LOG NUMBER: ' GET MLOGNUM
READ

* Searches file FORM1556

USE FORM1556
LOCATE FOR LOGNUM = MLOGNUM

* If log number is found, places data of screen for editing.

IF FOUND()
  CLEAR
  SET FORMAT TO MOD1556
  EDIT
  USE
  CLEAR
ELSE

* If not found, states so and returns to procedure MAINTAIN.

  @ 15,10 SAY 'LOG NUMBER NOT FOUND.'
ENDIF
RETURN

*EOF: Procedure MOD1556

PROCEDURE MODCSE

* Procedure MODCSE
*
* This procedure is called by procedure MAINTAIN and allows user
* to edit data in the data base file COURSE. This procedure
* searches for a composite key (CNUMBER, CSTARTDT).

CLEAR

* Allows user to enter both course start date and course number.

MNUMBER = "      "
MSTARTDT = "      "
@ 10,10 SAY 'ENTER THE COURSE NUMBER: ' GET MNUMBER
@ 12,10 SAY 'ENTER THE COURSE START DATE (MM/DD/YY): ' GET MSTARTDT
READ

* Searches file COURSE for composite key. If found user can edit.

USE COURSE
LOCATE FOR CNUMBER = MNUMBER .AND. CSTARTDT = CTOD(MSTARTDT)
IF FOUND()
  CLEAR
  SET FORMAT TO COURSE
  EDIT
  USE
  CLEAR
ELSE

```

* If not found, states so and returns to procedure MAINTAIN.

@ 15,10 SAY 'COURSE NUMBER NOT FOUND.'

ENDIF

RELE MNUMBER, MSTARTDT

RETURN

*EOF: Procedure MODCSE.

PROCEDURE MODSOURC

* Procedure MODSOURC

*

* This procedure is called by procedure MAINTAIN and allows user

* to edit data in the data base file SOURCE.

CLEAR

* Allows user to enter source unit identification code.

MSUIC = " "

@ 10,10 SAY 'ENTER SOURCE UIC: ' GET MSUIC

READ

* Searches file SOURCE. If found, allows user to edit.

USE SOURCE

LOCATE FOR SUIC = MSUIC

IF FOUND()

CLEAR

SET FORMAT TO MODSOURC

EDIT

USE

CLEAR

ELSE

* If not found, states so and returns to procedure MAINTAIN.

@ 15,10 SAY 'SOURCE NOT FOUND.'

ENDIF

RETURN

*EOF: Procedure MODSOURC.

PROCEDURE REPORTS

*

* Procedure REPORTS

*

* This procedure produces the reports menu and allows

* the user to make a selection from three standardized reports.

*

SET TALK OFF

SET BELL OFF

```
SET STATUS ON
SET ESCAPE OFF
SET CONFIRM ON
```

```
DO WHILE .T.
```

```
    * ---Display menu options, centered on the screen.
```

```
    *   draw menu border and print heading
```

```
    CLEAR
```

```
    @ 2, 0 TO 13,79 DOUBLE
```

```
    @ 3,23 SAY (P R I N T   R E P O R T S   M E N U)
```

```
    @ 4,1 TO 4,78 DOUBLE
```

```
    * ---display detail lines
```

```
    @ 7,28 SAY (1. TRAINING LOG FOR 1556)
```

```
    @ 8,28 SAY (2. TRAINING COSTS BY CATEGORY)
```

```
    @ 9,28 SAY (3. LIST OF SOURCES)
```

```
    @ 11, 28 SAY '0. EXIT'
```

```
    * ---Initialize selectnum and read in user's menu selection
```

```
    STORE 0 TO selectnum
```

```
    @ 13,33 SAY " select      "
```

```
    @ 13,42 GET selectnum PICTURE "9" RANGE 0,3
```

```
    READ
```

```
    * ---Execute user's selection
```

```
    DO CASE
```

```
        CASE selectnum = 0
```

```
            SET TALK ON
```

```
            CLEAR ALL
```

```
            RETURN
```

```
        CASE selectnum = 1
```

```
        * DO TRAINING LOG FOR 1556
```

```
            DO LOG
```

```
            SET CONFIRM OFF
```

```
            STORE ' ' TO wait_subst
```

```
            @ 23,0 SAY 'Press any key to continue...' GET wait_subst
```

```
            READ
```

```
            SET CONFIRM ON
```

```
        CASE selectnum = 2
```

```
        * DO TRAINING COSTS BY CATEGORY
```

```
            DO COSTS
```

```
            SET CONFIRM OFF
```

```
            STORE ' ' TO wait_subst
```

```
            @ 23,0 SAY 'Press any key to continue...' GET wait_subst
```

```
            READ
```

```
            SET CONFIRM ON
```

```
        CASE selectnum = 3
```

```
        * DO LIST OF SOURCES
```

```
            DO SOURCES
```

```
            SET CONFIRM OFF
```

```
            STORE ' ' TO wait_subst
```

```
            @ 23,0 SAY 'Press any key to continue...' GET wait_subst
```

READ
SET CONFIRM ON
ENDCASE

ENDDO T
RETURN
* EOF: REPORTS.PRG

PROCEDURE LOG

* Procedure LOG
*
* This procedure generates a printed log of all FORM 1556'S where
* the training has not been completed. It is comprised of
* fields from FORM1556, EMPLOYEE, and COURSE joined via INTERSEC.

SET TALK OFF

* Direct user to set up printer.
*

CLEAR
@ 9,10 SAY 'PLEASE VERIFY THAT YOUR PRINTER IS READY.'
@ 10,10 SAY 'PRINTER MUST BE SET TO 132 CHARACTER WIDE PAPER.'
WAIT
@ 15,10 SAY 'PLEASE BE PATIENT. YOUR DATA IS BEING COMPILED.'

* Join only those records whose training is uncompleted.
*

SELECT 1
USE FORM1556
SELECT 2
USE INTERSEC
JOIN WITH FORM1556 TO TEMP FOR LOGNUM = FORM1556 -> LOGNUM .AND. .NOT.
FORM1556->LCOMPLETED FIELDS FORM1556 -> LTINDCOST, LOGNUM, ESSN,
CNUMBER, CSTARTDT

SELECT 3
USE EMPLOYEE
SELECT 4
USE TEMP
JOIN WITH C TO TEMP2 FOR ESSN = C->ESSN FIELDS C->ELASTNAME,
C->EFIRSTNAME, C->EMINITIAL, C->EDCODE, LTINDCOST, LOGNUM, ESSN,
CNUMBER, CSTARTDT

SELECT 5
USE COURSE
SELECT 6
USE TEMP2
JOIN WITH COURSE TO TEMP3 FOR CNUMBER = COURSE->CNUMBER .AND.
CSTARTDT = COURSE->CSTARTDT FIELDS COURSE->CTITLE, COURSE->CCITY,
COURSE->CSTATE, COURSE->CTDIRCOST, ELASTNAME, EFIRSTNAME,
EMINITIAL, EDCODE, LTINDCOST, CNUMBER, CSTARTDT, LOGNUM

* Print log and do housekeeping.
*
USE TEMP3

```
CLEAR
SET PRINT ON
REPORT FORM LOGS
SET PRINT OFF
CLOSE ALL
ERASE TEMP.DBF
ERASE TEMP2.DBF
ERASE TEMP3.DBF
CLEAR
RETURN
```

* EOF: LOG.PRG

PROCEDURE COSTS

* Procedure COSTS.PRG

*

* This procedure generates a report of expenditures for all
* requested training by categories that have not been completed.
* It is comprised of fields from FORM1556 and COURSE joined via
* INTERSEC with narrative training categories from TRNGCODE.

*

* Direct user to set up printer.

*

SET TALK OFF

CLEAR

@ 9,10 SAY 'PLEASE VERIFY THAT YOUR PRINTER IS READY.'

WAIT

@ 15,10 SAY 'PLEASE BE PATIENT. YOUR DATA IS BEING COMPILED.'

*

* Join only those records whose training is not completed.

*

SELECT 1

USE FORM1556

SELECT 2

USE INTERSEC

JOIN WITH FORM1556 TO TEMP FOR LOGNUM = FORM1556 -> LOGNUM .AND. .NOT.
FORM1556 -> LCOMPLETED FIELDS FORM1556 -> LTINDCOST, LOGNUM,
CNUMBER, CSTARTDT

SELECT 3

USE COURSE

SELECT 4

USE TEMP

JOIN WITH COURSE TO TEMP2 FOR CNUMBER = COURSE->CNUMBER .AND.
CSTARTDT = COURSE->CSTARTDT FIELDS COURSE->CTYPE, COURSE->CTDIRCOST,
LOGNUM, LTINDCOST

SELECT 5

USE TRNGCODE

SELECT 6

USE TEMP2

JOIN WITH TRNGCODE TO TEMP3 FOR CTYPE = TRNGCODE->CTYPE FIELDS
TRNGCODE->CATEGORY, CTDIRCOST, LOGNUM, LTINDCOST

* Print report and do housekeeping.

*

USE TEMP3
INDEX ON CATEGORY TO TEMP3NDX
USE TEMP3 INDEX TEMP3NDX
CLEAR
SET PRINT ON
REPORT FORM COSTS
SET PRINT OFF
CLOSE ALL
ERASE TEMP.DBF
ERASE TEMP2.DBF
ERASE TEMP3.DBF
ERASE TEMP3NDX.NDX
CLEAR
RETURN

* EOF: COSTS.PRG
PROCEDURE SOURCES

* Procedure SOURCES.PRG

*

* This procedure generates a report of all the different
* training sources that have been utilized for the current
* fiscal year. It is a dump of SOURCE.

CLEAR
SET TALK OFF

* Direct user to set up printer

*

@ 9, 10 SAY "PLEASE VERIFY THAT YOUR PRINTER IS READY"

WAIT

@ 15,10 SAY "PLEASE BE PATIENT. YOUR DATA IS BEING INDEXED."

* Index SOURCE, print report and do housekeeping.

*

USE SOURCE
INDEX ON SNAME TO SOURCNDX
USE SOURCE INDEX SOURCNDX
CLEAR
SET PRINT ON
REPORT FORM SOURCE
SET PRINT OFF
USE
ERASE SOURCNDX.NDX

CLEAR
RETURN

* EOF: SOURCES.PRG

PROCEDURE QUERIES

* Procedure QUERIES

*

* This procedure allows the user who has a good working
 * knowledge of dbase to perform ad hoc queries of the database.

```
CLEAR
SET BELL OFF
SET TALK OFF
DO WHILE .T.
  CLEAR
  @ 1,10 SAY "CAUTION:  UNLESS YOU HAVE A GOOD WORKING KNOWLEDGE "
  @ 2,10 SAY "OF DBASE, YOU SHOULD BE VERY CAREFUL USING THIS"
  @ 3,10 SAY "OPTION.  RECOMMEND BACKING UP YOUR DATA FILES BEFORE"
  @ 4,10 SAY "PROCEEDING."
  @ 5,10 SAY " "

  ACCEPT "ENTER THE COMMAND TO BE USED (N TO STOP):  " TO MCMD
  IF UPPER(MCMD) = 'N'
    CLEAR
    RETURN
  ENDIF
&MCMD
WAIT
ENDDO
RETURN
```

* EOF: QUERIES.PRG

PROCEDURE BANNER

* Procedure BANNER

*

* This procedure produces a banner/title page and tells
 * the user to ensure the caps key is in place and when
 * to use the escape key.

*

*

* Print the tile

*

```
@3,24 SAY '      CCCC      PPPP      0000'
@4,24 SAY '    CC  CC    PP  PP    00 00'
@5,24 SAY '    CC      PP  PP    00 00'
@6,24 SAY '    CC      PPPP      00 00'
@7,24 SAY '    CC      PP      00 00'
@8,24 SAY '    CC  CC    PP      00 00'
@9,24 SAY '      CCCC      PP      0000'
```

*

* Print remarks

*

```
@11,13 SAY 'PLEASE ENSURE THE "CAPS LOCK" IS ON.  THE WORD "Caps"'
@12,13 SAY 'WILL APPEAR ON THE RIGHT END OF COMMAND LINE.'
@14,13 SAY 'USE THE "ESC" KEY TO:  1.  SKIP A SCREEN.'
@15,13 SAY '                                2.  TO EXIT A SCREEN IF'
@16,13 SAY '                                INCORRECT DATA IS USED.'
```

*

* Wait command to allow user to read screen.

*

WAIT

CLEAR
RETURN

* INTERSEC. FMT

@ 4, 23 SAY "LOG INFORMATION FOR FORM 1556"
@ 7, 28 SAY "LOG NUMBER:"
@ 7, 41 GET INTERSEC->LOGNUM
@ 9, 25 SAY "EMPLOYEE SSN:"
@ 9, 40 GET INTERSEC->ESSN
@ 11, 24 SAY "COURSE NUMBER:"
@ 11, 40 GET INTERSEC->CNUMBER
@ 13, 21 SAY "COURSE STARTING DATE:"
@ 13, 44 GET INTERSEC->CSTARTDT
@ 2, 17 TO 15, 55 DOUBLE
@ 18, 2 SAY "TYPE IN THE REQUESTED INFORMATION CONTAINED ON YOUR FORM 1556."
@ 19, 2 SAY "HIT CTRL END OR CTRL W WHEN DONE."

* MOD1556. FMT

@ 3, 28 SAY "FORM 1556 LOG DATA"
@ 6, 4 SAY "LOG NUMBER:"
@ 6, 16 GET FORM1556->LOGNUM
@ 6, 24 SAY "DATE OF 1556:"
@ 6, 38 GET FORM1556->LD1556
@ 6, 48 SAY "APPROVAL DATE:"
@ 6, 63 GET FORM1556->LDAPP1556
@ 8, 4 SAY "TRAINING COMPLETED?"
@ 8, 24 GET FORM1556->LCOMPLETED
@ 8, 27 SAY "CODE FOR NONCOMPLETION:"
@ 8, 51 GET FORM1556->LNONCOMPL
@ 10, 4 SAY "ACCOUNTING CLASSIFICATION FOR DIRECT COSTS:"
@ 10, 48 GET FORM1556->LPAYMENT
@ 12, 4 SAY "TOTAL INDIRECT COSTS:"
@ 12, 26 GET FORM1556->LTINDCOST
@ 12, 36 SAY "FUNDING SOURCE:"
@ 12, 52 GET FORM1556->LFUNDSOURC
@ 12, 55 SAY "TANGO NO:"
@ 12, 65 GET FORM1556->LTANGONO
@ 14, 4 SAY "JOB ORDER NO:"
@ 14, 18 GET FORM1556->LJOBORDNO
@ 14, 26 SAY "TRAVEL ORDER NO:"
@ 14, 43 GET FORM1556->LTRAVORDNO
@ 16, 4 SAY "GOVT DEDUCTION:"
@ 16, 20 GET FORM1556->LGOVTDeduc
@ 16, 23 SAY "INVOICE:"
@ 16, 32 GET FORM1556->LINVOICE
@ 18, 2 TO 18, 76 DOUBLE
@ 1, 2 TO 18, 76 DOUBLE
@ 19, 2 SAY "INPUT THE NEW RECORD INFORMATION NOW. HIT CTRL END WHEN DONE."

* EMPLOYEE. FMT

@ 3, 32 SAY "EMPLOYEE FILE"
@ 6, 4 SAY "LAST NAME:"
@ 6, 15 GET EMPLOYEE->ELASTNAME

```

@ 6, 37 SAY "FIRST NAME: "
@ 6, 49 GET EMPLOYEE->EFIRSTNAME
@ 6, 61 SAY "MIDDLE INIT: "
@ 6, 74 GET EMPLOYEE->EMINITIAL
@ 8, 6 SAY "SSN: "
@ 8, 11 GET EMPLOYEE->ESSN
@ 8, 22 SAY "SEX: "
@ 8, 27 GET EMPLOYEE->ESEX
@ 8, 30 SAY "ORGANIZATION UIC: "
@ 8, 48 GET EMPLOYEE->EUIC
@ 8, 55 SAY "POSITION LEVEL: "
@ 8, 71 GET EMPLOYEE->ELEVEL
@ 10, 8 SAY "DEPT CODE: "
@ 10, 19 GET EMPLOYEE->EDCODE
@ 10, 26 SAY "FEDERAL SERVICE (YYMM): "
@ 10, 50 GET EMPLOYEE->ECONTSERV
@ 10, 56 SAY "HANDICAPPED?"
@ 10, 69 GET EMPLOYEE->EHANDICAP
@ 12, 11 SAY "POSITION TITLE/FUNCTION: "
@ 12, 36 GET EMPLOYEE->ETITLE
@ 14, 4 SAY "PAY PLAN: "
@ 14, 14 GET EMPLOYEE->EPAYPLAN
@ 14, 19 SAY "PAY SERIES: "
@ 14, 31 GET EMPLOYEE->ESERIES
@ 14, 37 SAY "PAYGRADE: "
@ 14, 47 GET EMPLOYEE->EGRADE
@ 14, 51 SAY "TYPE OF APPOINTMENT: "
@ 14, 72 GET EMPLOYEE->EPAPPT
@ 16, 7 SAY "PART TIME/FULL TIME (P/F): "
@ 16, 34 GET EMPLOYEE->ESKED
@ 16, 37 SAY "TRAINING PLAN: "
@ 16, 52 GET EMPLOYEE->ETRNGPLAN
@ 16, 55 SAY "ETHNIC GROUP: "
@ 16, 69 GET EMPLOYEE->ETHNICGRP
@ 2, 2 TO 17, 76 DOUBLE
@ 18,2 SAY"INPUT THE NEW RECORD INFORMATION NOW. HIT CTRL END OR CTRL W
      WHEN DONE."
@ 19,2 SAY"HIT ESCAPE (ESC) TO CANCEL INPUT AND RETURN TO PREVIOUS MENU."

* COURSE.FMT

@ 2, 29 SAY "COURSE INFORMATION"
@ 4, 12 SAY "COURSE TITLE: "
@ 4, 26 GET COURSE->CTITLE
@ 6, 8 SAY "COURSE NO: "
@ 6, 19 GET COURSE->CNUMBER
@ 6, 31 SAY "CITY: "
@ 6, 37 GET COURSE->CCITY
@ 6, 59 SAY "STATE: "
@ 6, 66 GET COURSE->CSTATE
@ 8, 14 SAY "STARTING DATE: "
@ 8, 29 GET COURSE->CSTARTDT
@ 8, 39 SAY "ENDING DATE: "
@ 8, 52 GET COURSE->CENDDT
@ 10, 10 SAY "COURSE HOURS PER PERSON - DURING DUTY: "
@ 10, 49 GET COURSE->CHRSDDUTY

```

```

@ 10, 53 SAY "NONDUTY: "
@ 10, 62 GET COURSE->CHRSNDUTY
@ 12, 5 SAY "PURPOSE OF COURSE: "
@ 12, 24 GET COURSE->CPURPOSE
@ 12, 28 SAY "TYPE: "
@ 12, 34 GET COURSE->CTYPE
@ 12, 37 SAY "PRIORITY: "
@ 12, 47 GET COURSE->CPRIORITY
@ 12, 50 SAY "METHOD OF EVALUATION: "
@ 12, 72 GET COURSE->CMETHOD
@ 14, 10 SAY "REASON FOR SELECTION: "
@ 14, 32 GET COURSE->CREASON
@ 14, 37 SAY "COURSE DIRECT COST: "
@ 14, 57 GET COURSE->CTDIRCOST
@ 16, 29 SAY "SOURCE UIC: "
@ 16, 41 GET COURSE->SUIC
@ 1, 3 TO 17, 75 DOUBLE
@ 19,2 SAY"INPUT THE NEW RECORD INFORMATION NOW. HIT CTRL END WHEN DONE."
@ 20,2 SAY"HIT ESCAPE (ESC) TO CANCEL INPUT AND RETURN TO PREVIOUS MENU."

```

* MODS01.FMT

```

@ 3, 22 SAY "SOURCE INFORMATION INPUT SCREEN"
@ 7, 10 SAY "SOURCE NAME: "
@ 7, 23 GET SOURCE->SNAME
@ 10, 10 SAY "SOURCE CITY: "
@ 10, 23 GET SOURCE->SCITY
@ 10, 35 SAY "SOURCE STATE: "
@ 10, 49 GET SOURCE->SSTATE
@ 13, 10 SAY "SOURCE UIC: "
@ 13, 22 GET SOURCE->SUIC
@ 13, 32 SAY "SOURCE TYPE: "
@ 13, 45 GET SOURCE->STYPE
@ 1, 6 TO 1, 68 DOUBLE
@ 16, 6 TO 16, 69 DOUBLE
@ 1, 6 TO 16, 6 DOUBLE
@ 1, 6 TO 1, 70 DOUBLE
@ 1, 6 TO 17, 70 DOUBLE
@ 19,2 SAY"INPUT THE NEW RECORD INFORMATION NOW. HIT CTRL END WHEN DONE."
@ 20,2 SAY"HIT ESCAPE (ESC) TO CANCEL INPUT AND RETURN TO PREVIOUS MENU."

```

* EMPLOY.FMT

```

@ 3, 32 SAY "EMPLOYEE FILE"
@ 6, 4 SAY "LAST NAME: "
@ 6, 15 GET EMPLOYEE->ELASTNAME
@ 6, 37 SAY "FIRST NAME: "
@ 6, 49 GET EMPLOYEE->EFIRSTNAME
@ 6, 61 SAY "MIDDLE INIT: "
@ 6, 74 GET EMPLOYEE->EMINITIAL
@ 8, 6 SAY "SSN: "
@ 8, 11 GET EMPLOYEE->ESSN
@ 8, 22 SAY "SEX: "
@ 8, 27 GET EMPLOYEE->ESEX
@ 8, 30 SAY "ORGANIZATION UIC: "
@ 3, 48 GET EMPLOYEE->EUIC

```

```

@ 8, 55 SAY "POSITION LEVEL: "
@ 8, 71 GET EMPLOYEE->ELEVEL
@ 10, 8 SAY "DEPT CODE: "
@ 10, 19 GET EMPLOYEE->EDCODE
@ 10, 26 SAY "FEDERAL SERVICE (YYMM): "
@ 10, 50 GET EMPLOYEE->ECONTSERV
@ 10, 56 SAY "HANDICAPPED?"
@ 10, 69 GET EMPLOYEE->EHANDICAP
@ 12, 11 SAY "POSITION TITLE/FUNCTION: "
@ 12, 36 GET EMPLOYEE->ETITLE
@ 14, 4 SAY "PAY PLAN: "
@ 14, 14 GET EMPLOYEE->EPAYPLAN
@ 14, 19 SAY "PAY SERIES: "
@ 14, 31 GET EMPLOYEE->ESERIES
@ 14, 37 SAY "PAYGRADE: "
@ 14, 47 GET EMPLOYEE->EGRADE
@ 14, 51 SAY "TYPE OF APPOINTMENT: "
@ 14, 72 GET EMPLOYEE->EPAPPT
@ 16, 7 SAY "PART TIME/FULL TIME (P/F): "
@ 16, 34 GET EMPLOYEE->ESKED
@ 16, 37 SAY "TRAINING PLAN: "
@ 16, 52 GET EMPLOYEE->ETRNGPLAN
@ 16, 55 SAY "ETHNIC GROUP: "
@ 16, 69 GET EMPLOYEE->ETHNICGRP
@ 2, 2 TO 17, 76 DOUBLE
@ 19, 2 SAY "MAKE CHANGES TO ANY FIELD NOW. WHEN DONE, HIT CTRL END."

```

* COURSE1.FMT

```

@ 2, 29 SAY "COURSE INFORMATION"
@ 4, 12 SAY "COURSE TITLE: "
@ 4, 26 GET COURSE->CTITLE
@ 6, 8 SAY "COURSE NO: "
@ 6, 19 GET COURSE->CNUMBER
@ 6, 31 SAY "CITY: "
@ 6, 37 GET COURSE->CCITY
@ 6, 59 SAY "STATE: "
@ 6, 66 GET COURSE->CSTATE
@ 8, 14 SAY "STARTING DATE: "
@ 8, 29 GET COURSE->CSTARTDT
@ 8, 39 SAY "ENDING DATE: "
@ 8, 52 GET COURSE->CENDDT
@ 10, 10 SAY "COURSE HOURS PER PERSON - DURING DUTY: "
@ 10, 49 GET COURSE->CHRSDDUTY
@ 10, 53 SAY "NONDUTY: "
@ 10, 62 GET COURSE->CHRSNDUTY
@ 12, 5 SAY "PURPOSE OF COURSE: "
@ 12, 24 GET COURSE->CPURPOSE
@ 12, 28 SAY "TYPE: "
@ 12, 34 GET COURSE->CTYPE
@ 12, 37 SAY "PRIORITY: "
@ 12, 47 GET COURSE->CPRIORITY
@ 12, 50 SAY "METHOD OF EVALUATION: "
@ 12, 72 GET COURSE->CMETHOD
@ 14, 10 SAY "REASON FOR SELECTION: "
@ 14, 32 GET COURSE->CREASON

```

```

@ 14, 37 SAY "COURSE DIRECT COST: "
@ 14, 57 GET COURSE->CTDIRCOST
@ 16, 29 SAY "SOURCE UIC: "
@ 16, 41 GET COURSE->SUIC
@ 1, 3 TO 17, 75 DOUBLE
@ 20, 2 SAY "MAKE CHANGES TO ANY FIELD NOW. IF NONE, HIT CTRL END."

```

* MODSOURC.FMT

```

@ 3, 22 SAY "SOURCE INFORMATION INPUT SCREEN"
@ 7, 10 SAY "SOURCE NAME: "
@ 7, 23 GET SOURCE->SNAME
@ 10, 10 SAY "SOURCE CITY: "
@ 10, 23 GET SOURCE->SCITY
@ 10, 35 SAY "SOURCE STATE: "
@ 10, 49 GET SOURCE->SSTATE
@ 13, 10 SAY "SOURCE UIC: "
@ 13, 22 GET SOURCE->SUIC
@ 13, 32 SAY "SOURCE TYPE: "
@ 13, 45 GET SOURCE->STYPE
@ 1, 6 TO 1, 68 DOUBLE
@ 16, 6 TO 16, 69 DOUBLE
@ 1, 6 TO 16, 6 DOUBLE
@ 1, 6 TO 1, 70 DOUBLE
@ 1, 6 TO 17, 70 DOUBLE
@ 19, 2 SAY "MAKE CHANGES TO ANY FIELD NOW. WHEN DONE, HIT CTRL END."

```

APPENDIX E. PROTOTYPE: SECOND ITERATION

```
* Program...: CPO.PRG version 2
* Author...: LCDR SHARON SLOMINSKI, USN
*           LT IVON YOUNG, SC, USN
* Date.....: 08 FEB 88
*
* The purpose of this program is to oversee the maintenance
* of the database system for civilian personnel training.
*
SET COLOR TO W+/B+,W+/R,R+
CLEAR
SET TALK OFF
SET BELL OFF
SET STATUS ON
SET ESCAPE OFF
SET CONFIRM ON

SET PROCEDURE TO CALLS

* Call procedure BANNER to give title page and instructions
* to user.

DO BANNER

DO WHILE .T.

    * ---Display menu options, centered on the screen.
    * draw menu border and print heading
    CLEAR
    @ 2, 0 TO 15,79 DOUBLE
    @ 3,11 SAY (C P O   T R A I N I N G   T R A C K I N G   S Y S T E M)
    @ 4,1 TO 4,78 DOUBLE
    * ---display detail lines
    @ 7,30 SAY (1. Add a Record)
    @ 8,30 SAY (2. Change a Record)
    @ 9,30 SAY (3. Delete a Record)
    @ 10,30 SAY (4. Ad Hoc Queries)
    @ 11,30 SAY (5. Print Reports)
    @ 13,30 SAY '0. Exit'
    STORE 0 TO selectnum
    @ 15,33 SAY " select      "
    @ 15,42 GET selectnum PICTURE "9" RANGE 0,5
    READ
    * Case statement. Depending on user's selection the correct
    * procedure will be called and executed.

    DO CASE

    * User is exiting the program.
      CASE selectnum = 0
        SET TALK ON
        USE F1556
```

```

        PACK
        USE
        CLEAR ALL
        CLEAR
        RETURN

*      Do Add a Record
      CASE selectnum = 1
        DO ADDMENU
        SET CONFIRM OFF
        STORE ' ' TO wait_subst
        @ 23,0 SAY 'Press any key to continue...' GET wait_subst
        READ
        SET CONFIRM ON

*      Do Change a Record
      CASE selectnum = 2
        DO CHGMENU
        SET CONFIRM OFF
        STORE ' ' TO wait_subst
        @ 23,0 SAY 'Press any key to continue...' GET wait_subst
        READ
        SET CONFIRM ON

*      Do Delete a Record
      CASE selectnum = 3
        DO DELERCD
        SET CONFIRM OFF
        STORE ' ' TO wait_subst
        @ 23,0 SAY 'Press any key to continue...' GET wait_subst
        READ
        SET CONFIRM ON

*      Do Ad Hoc Queries
      CASE selectnum = 4
        DO QUERIES
        SET CONFIRM OFF
        STORE ' ' TO wait_subst
        @ 23,0 SAY 'Press any key to continue...' GET wait_subst
        READ
        SET CONFIRM ON

*      Do Print Reports
      CASE selectnum = 5
        DO RPTMENU
        SET CONFIRM OFF
        STORE ' ' TO wait_subst
        @ 23,0 SAY 'Press any key to continue...' GET wait_subst
        READ
        SET CONFIRM ON

ENDCASE

ENDDO T
CLEAR

```

RETURN

* EOF: CPO. PRG

PROCEDURE BANNER

* Procedure BANNER

*

* This procedure is called from CPO and produces a banner/title
* page and tells to the user ensure the caps key is in place and
* when to use the escape key to continue.

*

* Print the tile

*

@3,24 SAY '	CCCC	PPPP	0000'
@4,24 SAY '	CC CC	PP PP	00 00'
@5,24 SAY '	CC	PP PP	00 00'
@6,24 SAY '	CC	PPPP	00 00'
@7,24 SAY '	CC	PP	00 00'
@8,24 SAY '	CC CC	PP	00 00'
@9,24 SAY '	CCCC	PP	0000'

*

* Print remarks

*

@11,13 SAY 'Please Ensure The "CAPS LOCK" Is On. The Word "Caps"
@12,13 SAY 'Will Appear In The Lower Right End Of The Command Line.'
@14,13 SAY 'Use The "ESC" Key To: 1. Skip A Screen.'
@15,13 SAY '2. To Exit A Screen If'
@16,13 SAY 'Incorrect Data Is Used.'

*

* Wait command to allow user to read screen.

*

WAIT

CLEAR

RETURN

* EOF: BANNER. PRG

PROCEDURE ADDMENU

* Procedure ADDMENU. PRG

*

* This procedure is called by CPO and is used to add new 1556 records
* or invoice information to existing 1556 records to the database. This
* procedure is a menu which calls the procedures ADD1556 or ADDINVOICE.

SET TALK OFF

SET BELL OFF

SET STATUS ON

SET ESCAPE OFF

SET CONFIRM ON

USE F1556

DO WHILE .T.

* ---Display menu options, centered on the screen.

* draw menu border and print heading

CLEAR

@ 2, 0 TO 12,79 DOUBLE

```

@ 3,24 SAY (A D D   A   R E C O R D   M E N U)
@ 4,1 TO 4,78 DOUBLE
* ---display detail lines
@ 7,30 SAY (1. Add A New 1556 Record)
@ 8,30 SAY (2. Add Invoice Information)
@ 10,30 SAY '0. Exit'
STORE 0 TO selectnum
@ 12,33 SAY " select          "
@ 12,42 GET selectnum PICTURE "9" RANGE 0,2
READ

DO CASE

*   Return to main menu.
CASE selectnum = 0
    SET TALK ON
    CLEAR ALL
    RETURN

*   Do Add a new 1556 record
CASE selectnum = 1
    DO ADD1556
    SET CONFIRM OFF
    STORE ' ' TO wait_subst
    @ 23,0 SAY 'Press any key to continue...' GET wait_subst
    READ
    SET CONFIRM ON

*   Do Add invoice information to existing 1556 record.
CASE selectnum = 2
    DO ADDINVOICE
    SET CONFIRM OFF
    STORE ' ' TO wait_subst
    @ 23,0 SAY 'Press any key to continue...' GET wait_subst
    READ
    SET CONFIRM ON

ENDCASE

ENDDO T
RETURN
* EOF: ADDMENU.PRG

PROCEDURE ADD1556

*   Procedure ADD1556.PRG
*
*   This procedure is called by ADDMENU and allows the user to add new
*   1556 records to the database after checking for unique log number.

SET TALK OFF
SET BELL OFF
SET STATUS ON
SET ESCAPE OFF
SET CONFIRM ON

```

* Verify that the log number to the new 1556 record has not been
* used before.

```
MDUP = 'FALSE'
DO WHILE MDUP = 'FALSE'
  CLEAR
  mLOGNUM = ' '
  @ 10, 20 SAY 'Enter Log Number: ' GET mLOGNUM
  READ
  USE F1556
  LOCATE FOR LOGNUM=mLOGNUM
  IF FOUND ( )
    @ 12,10 SAY 'Log Number Already Exist. Enter A New Log Number.'
    WAIT
  ELSE
    MDUP = 'TRUE'
  ENDIF
ENDDO
```

* Add a new record to the database.

```
CLEAR
APPEND BLANK RECORD
```

* Add log information
SET FORMAT TO LOG
EDIT

* Add trainee information
SET FORMAT TO TRAINEE
EDIT

* Add training course information
SET FORMAT TO TRNCOURS
EDIT

* Add costs and billing information
SET FORMAT TO BILLINGS
EDIT

```
CLOSE ALL
```

```
CLEAR
```

```
RELE mLOGNUM
```

```
RETURN
```

* EOF ADD1556.PRG

```
PROCEDURE ADDINVOI
```

* Procedure ADDINVOI.PRG

*

* This procedure is called by ADDMENU and allows the user to enter
* invoice information to appropriate 1556 record.

```
SET TALK OFF
SET BELL OFF
SET STATUS ON
SET ESCAPE OFF
SET CONFIRM ON
```

* Locate appropriate 1556 record by log number.

```

MDUP = 'FALSE'
DO WHILE MDUP = 'FALSE'
  CLEAR
  mLOGNUM = ' '
  @ 10, 20 SAY 'Enter Log Number: ' GET mLOGNUM
  READ
  USE F1556
  LOCATE FOR LOGNUM=mLOGNUM
  IF .NOT. FOUND ()
    @ 12,10 SAY 'Log Number Does Not Exist. Enter A New Log Number.'
    WAIT
  ELSE
    MDUP = 'TRUE'
  ENDIF
ENDDO

```

* Add invoice information to existing 1556 record.

```

CLEAR
SET FORMAT TO INVOICE
EDIT
USE
CLEAR
RELE mLOGNUM
RETURN
* EOF ADDINVOI.PRG

```

PROCEDURE CHGMENU

```

* Procedure CHGMENU.PRG
*
* This procedure is called by CPO and allows the user to change data
* either by individual sections of the 1556 or as an entire record.
* This procedure calls the following procedures: CHGLOG, CHGTRAIN,
* CHGCOURS, CHGBILL, CHGINVOI, and CHG1556.

```

```

SET TALK OFF
SET BELL OFF
SET STATUS ON
SET ESCAPE OFF
SET CONFIRM ON

```

```
DO WHILE .T.
```

```

  * ---Display menu options, centered on the screen.
  * draw menu border and print heading
  CLEAR
  @ 2, 0 TO 16,79 DOUBLE
  @ 3,11SAY(C H A N G E 1 5 5 6 R E C O R D I N F O R M A T I O N)
  @ 4,1 TO 4,78 DOUBLE
  * ---display detail lines
  @ 7,27 SAY (1. Change Log Information)
  @ 8,27 SAY (2. Change Trainee Information)
  @ 9,27 SAY (3. Change Training Course Data)
  @ 10,27 SAY (4. Change Costs and Billing Information)

```

```
@ 11,27 SAY (5. Change Invoice Information)
@ 12,27 SAY (6. Change an Entire 1556 Record)
@ 14,27 SAY '0. Exit'
STORE 0 TO selectnum
@ 16,33 SAY " select      "
@ 16,42 GET selectnum PICTURE "9" RANGE 0,6
READ
```

```
DO CASE
```

- * Return user to main menu
CASE selectnum = 0
SET TALK ON
CLEAR ALL
RETURN
- * Do change log information
CASE selectnum = 1
DO CHGLOG
SET CONFIRM OFF
STORE ' ' TO wait_subst
@ 23,0 SAY 'Press any key to continue...' GET wait_subst
READ
SET CONFIRM ON
- * Do change trainee information
CASE selectnum = 2
DO CHGTRAINEE
SET CONFIRM OFF
STORE ' ' TO wait_subst
@ 23,0 SAY 'Press any key to continue...' GET wait_subst
READ
SET CONFIRM ON
- * Do change training course data
CASE selectnum = 3
DO CHGCOURSE
SET CONFIRM OFF
STORE ' ' TO wait_subst
@ 23,0 SAY 'Press any key to continue...' GET wait_subst
READ
SET CONFIRM ON
- * Do change costs and billing information
CASE selectnum = 4
DO CHGBILLINGS
SET CONFIRM OFF
STORE ' ' TO wait_subst
@ 23,0 SAY 'Press any key to continue...' GET wait_subst
READ
SET CONFIRM ON
- * Do change invoice information
CASE selectnum = 5
DO CHGINVOICE
SET CONFIRM OFF
STORE ' ' TO wait_subst

```

        @ 23,0 SAY 'Press any key to continue...' GET wait_subst
        READ
        SET CONFIRM ON

*   Do change an entire 1556 record
    CASE selectnum = 6
        DO CHG1556
        SET CONFIRM OFF
        STORE ' ' TO wait_subst
        @ 23,0 SAY 'Press any key to continue...' GET wait_subst
        READ
        SET CONFIRM ON

    ENDCASE

    ENDDO T
    RETURN
* EOF: CHGMENU.PRG

PROCEDURE CHGLOG

*   Procedure CHGLOG.PRG
*
*   This procedure is called by CHGMENU and allows the user to change
*   data in the log section for a particular 1556 record. The appropriate
*   record is located by log number.

    SET TALK OFF
    SET BELL OFF
    SET STATUS ON
    SET ESCAPE OFF
    SET CONFIRM ON

*   Locate the appropriate 1556 record by log number.

    CLEAR
    mLOGNUM = ' '
    @ 10,20 SAY 'Enter Log Number: ' GET mLOGNUM
    READ
    USE F1556
    LOCATE FOR LOGNUM=mLOGNUM
    IF FOUND ()
        * CHANGE LOG INFORMATION

        CLEAR
        SET FORMAT TO CHGLOG
        EDIT
        USE
        CLEAR

    ELSE
        @ 12,20 SAY 'Log Number Not Found.'
    ENDIF

```

```
RELE mLOGNUM
RETURN
* EOF CHGLOG. PRG
```

PROCEDURE CHGTRAIN

```
* Procedure CHGTRAIN. PRG
*
* This procedure is called by CHGMENU and allows the user to change
* data in the trainee section for a particular 1556 record. This
* proper 1556 record is located by log number.
```

```
SET TALK OFF
SET BELL OFF
SET STATUS ON
SET ESCAPE OFF
SET CONFIRM ON
```

```
* Locate the appropriate 1556 record by log number.
```

```
CLEAR
mLOGNUM = '
@ 10, 20 SAY 'Enter Log Number: ' GET mLOGNUM
READ
USE F1556
LOCATE FOR LOGNUM=mLOGNUM
IF FOUND ()
```

```
  * CHANGE TRAINEE INFORMATION
```

```
  CLEAR
  SET FORMAT TO CHGTRAIN
  EDIT
  USE
  CLEAR
```

```
ELSE
  @ 12, 20 SAY 'Log Number Not Found.'
ENDIF
```

```
RELE mLOGNUM
RETURN
* EOF CHGTRAIN. PRG
```

PROCEDURE CHGCOURS

```
* Procedure CHGCOURS. PRG
*
* This procedure is called from CHGMENU and allows the user to change
* data in the training course section of a particular 1556 record. The
* proper 1556 record is located by log number.
```

```
SET TALK OFF
SET BELL OFF
SET STATUS ON
SET ESCAPE OFF
SET CONFIRM ON
```

* Locate the particular 1556 record.

```
CLEAR
mLOGNUM = '
@ 10, 20 SAY 'Enter Log Number: ' GET mLOGNUM
READ
USE F1556
LOCATE FOR LOGNUM=mLOGNUM
IF FOUND ()
```

* CHANGE TRAINING COURSE DATA

```
CLEAR
SET FORMAT TO CHGCOURS
EDIT
USE
CLEAR
```

```
ELSE
@ 12, 20 SAY 'Log Number Not Found.'
ENDIF
```

```
RELE mLOGNUM
RETURN
* EOF CHGCOURS.PRG
```

PROCEDURE CHGBILLI

* Procedure CHGBILLI.PRG

*
* This procedure is called from CHGMENU and allows the user to change
* data in the costs and billing section for a particular 1556 record.
* The proper 1556 record is located by log number.

```
SET TALK OFF
SET BELL OFF
SET STATUS ON
SET ESCAPE OFF
SET CONFIRM ON
```

* Locate the particular 1556 record by log number.

```
CLEAR
mLOGNUM = '
@ 10, 20 SAY 'Enter Log Number: ' GET mLOGNUM
READ
USE F1556
LOCATE FOR LOGNUM=mLOGNUM
IF FOUND ()
```

* CHANGE COSTS AND BILLING INFORMATION

```
CLEAR
SET FORMAT TO CHGBILL
EDIT
USE
CLEAR
```

```

ELSE
  @ 12, 20 SAY 'Log Number Not Found.'
ENDIF

RELE mLOGNUM
RETURN
* EOF CHGBILLI.PRG

PROCEDURE CHGINVOI

* Procedure CHGINVOI.PRG
*
* This procedure is called from CHGMENU and allows the user to change
* data in the invoice section for a particular 1556 record. The
* proper 1556 record is located by log number.

SET TALK OFF
SET BELL OFF
SET STATUS ON
SET ESCAPE OFF
SET CONFIRM ON

* Locate the particular 1556 record by log number.

CLEAR
mLOGNUM = ' '
@ 10, 20 SAY 'Enter Log Number: ' GET mLOGNUM
READ
USE F1556
LOCATE FOR LOGNUM=mLOGNUM
IF FOUND ( )

  * CHANGE INVOICE INFORMATION
  CLEAR
  SET FORMAT TO CHGINVOI
  EDIT
  USE
  CLEAR

ELSE
  @ 12, 20 SAY 'Log Number Not Found.'
ENDIF

RELE mLOGNUM
RETURN
* EOF CHGINVOICE.PRG

PROCEDURE CHG1556

* Procedure CHG1556.PRG
*
* This procedure is called from CHGMENU and allows the user to change
* data from an entire 1556 record by sections at one time. The proper
* 1556 record is located by log number.

```

SET TALK OFF
SET BELL OFF
SET STATUS ON
SET ESCAPE OFF
SET CONFIRM ON

* Locate the particular 1556 record by log number.

CLEAR
mLOGNUM = ' '
@ 10, 20 SAY 'Enter Log Number: ' GET mLOGNUM
READ
USE F1556
LOCATE FOR LOGNUM=mLOGNUM
IF FOUND ()

* CHANGE ENTIRE 1556 RECORD

CLEAR
SET FORMAT TO CHGLOG
EDIT
SET FORMAT TO CHGTRAIN
EDIT
SET FORMAT TO CHGCOURS
EDIT
SET FORMAT TO CHGBILL
EDIT
SET FORMAT TO CHGINVOI
EDIT
USE
CLEAR

ELSE
@ 12, 20 SAY 'Log Number Not Found.'
ENDIF

RELE mLOGNUM
RETURN
* EOF CHG1556.PRG

PROCEDURE DELERCD

* Procedure DELERCD.PRG

*
* This procedure is called from CPO and allows the user to
* delete individual 1556 records from the database. The proper
* records are located by log number and displayed to permit the
* user to verify that it is the correct record to be deleted.

SET TALK OFF
SET BELL OFF
SET STATUS ON
SET ESCAPE OFF
SET CONFIRM ON

```

* Continue until correct record is located.
TEST = 'FALSE'
DO WHILE TEST = 'FALSE'
  mANS = ' '
  CLEAR
  mLOGNUM = ' '
  @ 10,20 SAY 'Enter Log Number: ' GET mLOGNUM
  READ
  USE F1556
  LOCATE FOR LOGNUM=mLOGNUM
  IF FOUND ()
    * Load record for display

    STORE ELASTNAME TO mLAST
    STORE EFIRSTNAME TO mFIRST
    STORE EMINITIAL TO mINITIAL
    STORE ESSN TO mSSN
    STORE EUIC TO mUIC
    STORE CSTARTDT TO mSTART
    STORE CTITLE TO mTITLE

    * Display the record.
    CLEAR
    @ 2, 23 SAY "VERIFICATION OF 1556 TO BE DELETED"
    @ 5, 18 SAY "LOG NUMBER: "
    @ 5, 30 GET mLOGNUM
    @ 7, 18 SAY "NAME: "
    @ 7, 24 GET mLAST
    @ 7, 44 SAY " "
    @ 7, 46 GET mFIRST
    @ 7, 57 GET mINITIAL
    @ 7, 58 SAY " "
    @ 9, 18 SAY "SSN: "
    @ 9, 23 GET mSSN
    @ 11, 18 SAY "UIC: "
    @ 11, 23 GET mUIC
    @ 13, 18 SAY "COURSE START DATE: "
    @ 13, 37 GET mSTART
    @ 15, 18 SAY "COURSE TITLE: "
    @ 15, 32 GET mTITLE
    @ 1, 13 TO 17, 66 DOUBLE
    @ 3, 14 TO 3, 65
    CLEAR GETS

    @ 19,23 SAY 'Is This The Correct Record? Y/N: ' GET mANS
    READ

    * Delete the record
    IF UPPER(mANS) = 'Y'
      DELETE
      TEST = 'TRUE'
    ENDIF

  ELSE
    @ 12,20 SAY 'Log Number Not Found.'
    WAIT
  ENDIF
ENDWHILE

```

```

ENDIF

ENDDO TEST
USE
RELE mLOGNUM, mLAST, mFIRST, mINITIAL, mSSN, mUIC, mSTART, mTITLE
RELE mANS, TEST
RETURN
* EOF DELERCD.PRG

PROCEDURE QUERIES

* Procedure QUERIES
*
* This procedure is called from CPO and allows the user who has
* a good working knowledge of dBase to perform ad hoc queries
* of the database.

CLEAR
SET BELL OFF
SET TALK OFF

DO WHILE .T.
    CLEAR
    @ 1,10 SAY "CAUTION: Unless You Have A Good Working Knowledge "
    @ 2,10 SAY " Of dBASE, You Should Be Very Careful Using This"
    @ 3,10 SAY " Option. Recommed Backing Up Your Data Files"
    @ 4,10 SAY " Before Proceeding."
    @ 5,10 SAY " "

    ACCEPT " Enter The Command To Be Used (N TO STOP): " TO mCMD
    IF UPPER(mCMD) = 'N'
        CLEAR
        RETURN
    ENDIF
    &mCMD
    WAIT
ENDDO
RETURN
* EOF: QUERIES.PRG

PROCEDURE RPTMENU

* Procedure RPTMENU.PRG
*
* This procedure produces the reports menu and allows
* the user to make a selection from FOUR standardized reports.
* This procedure calls NEWLOG, NONCOMP, and DCOSTS.

SET TALK OFF
SET BELL OFF
SET STATUS ON
SET ESCAPE OFF
SET CONFIRM ON

DO WHILE .T.

```

```

* ---Display menu options, centered on the screen.
*   draw menu border and print heading
CLEAR
@ 2, 0 TO 13,79 DOUBLE
@ 3,23 SAY (P R I N T   R E P O R T S   M E N U)
@ 4,1 TO 4,78 DOUBLE
* ---display detail lines
@ 7,28 SAY (1. 1556 Training Log)
@ 8,28 SAY (2. Noncompleted Training Log)
@ 9,28 SAY (3. Training Costs By Department)
@ 11, 28 SAY '0. Exit'

* ---Initialize selectnum and read in user's menu selection
STORE 0 TO selectnum
@ 13,33 SAY " select      "
@ 13,42 GET selectnum PICTURE "9" RANGE 0,3
READ

* ---Execute user's selection
DO CASE

*   Return to main menu
CASE selectnum = 0
    SET TALK ON
    CLEAR ALL
    RETURN

*   Produce 1556 training log
CASE selectnum = 1
    DO NEWLOG
    SET CONFIRM OFF
    STORE ' ' TO wait_subst
    @ 23,0 SAY 'Press any key to continue...' GET wait_subst
    READ
    SET CONFIRM ON

*   Produce noncompleted training log
CASE selectnum = 2
    DO NONCOMP
    SET CONFIRM OFF
    STORE ' ' TO wait_subst
    @ 23,0 SAY 'Press any key to continue...' GET wait_subst
    READ
    SET CONFIRM ON

*   Produce costs by department
CASE selectnum = 3
    DO DCOSTS
    SET CONFIRM OFF
    STORE ' ' TO wait_subst
    @ 23,0 SAY 'Press any key to continue...' GET wait_subst
    READ
    SET CONFIRM ON

ENDCASE

```

ENDDO T
RETURN
* EOF: RPTMENU.PRG

PROCEDURE NEWLOG

* Procedure NEWLOG.PRG
*
* This procedure generates a printed log of FORM 1556'S.
* It is comprised of fields LOGNUM, EDEPTCODE, ELASTNAME,
* EFIRSTNAME, EMINITIAL, CTITLE, CNUMBER, DIRCOSTS, and
* INDIRCOSTS from F1556.

SET TALK OFF

* Direct user to set up printer.

CLEAR
@ 9,10 SAY 'Please Verify That Your Printer Is Ready.'
@ 10,10 SAY 'Printer Must Be Set To 132 Character Wide Paper.'
WAIT
@ 15,10 SAY 'Please Be Patient. Your Data Is Being Compiled.'

USE F1556
INDEX ON LOGNUM TO TEMPNDX
USE F1556 INDEX TEMPNDX

* Print log and do housekeeping.

CLEAR
SET PRINT ON
REPORT FORM RNEWLOG
SET PRINT OFF
CLOSE ALL
ERASE TEMPNDX.NDX
CLEAR
RETURN
* EOF: NEWLOG.PRG

PROCEDURE NONCOMP

* Procedure NONCOMP.PRG
*
* This procedure generates a printed log of all FORM 1556'S where
* the training has not been completed. It is comprised of
* fields from LOGNUM, EDEPTCODE, ELASTNAME, EFIRSTNAME,
* EMINITIAL, CTITLE, and CNUMBER from F1556.

SET TALK OFF

* Direct user to set up printer.

CLEAR
@ 9,10 SAY 'Please Verify That Your Printer Is Ready.'
@ 10,10 SAY 'Printer Must Be Set To 132 Character Wide Paper.'

WAIT
@ 15,10 SAY 'Please Be Patient. Your Data Is Being Compiled.'

USE F1556
SORT ON EDEPTCODE FOR .NOT. COMPLETED TO TEMP
USE TEMP
CLEAR
SET PRINT ON
REPORT FORM RNONCOMP
SET PRINT OFF
CLOSE ALL
ERASE TEMP.DBF
CLEAR
RETURN
* EOF: NONCOMP.PRG

PROCEDURE DCOSTS

* Procedure DCOSTS.PRG
*
* This procedure generates a report of expenditures for all
* requested training by departments. It is comprised of fields
* EDEPTCODE, DIRCOSTS, and INDIRCOSTS from F1556.

* Direct user to set up printer.

SET TALK OFF
CLEAR
@ 9,10 SAY 'Please Verify That Your Printer Is Ready.'
@ 10,10 Say 'Printer Must Be Set To 132 Character Wide Paper.'
WAIT
@ 15,10 SAY 'Please Be Patient. Your Data Is Being Compiled.'

* Print report and do housekeeping.

USE F1556
INDEX ON EDEPTCODE TO TEMPNDX
USE F1556 INDEX TEMPNDX
CLEAR
SET PRINT ON
REPORT FORM RDCOSTS
SET PRINT OFF
CLOSE ALL
ERASE TEMPNDX.NDX
CLEAR
RETURN
* EOF: DCOSTS.PRG

* LOG.FMT

@ 1, 9 SAY "Log Number:"
@ 1, 22 GET F1556->LOGNUM
@ 1, 35 SAY "Fiscal Year of Course:"
@ 1, 59 GET F1556->FYCRS
@ 3, 9 SAY "Multiple Employees:"
@ 3, 29 GET F1556->MULEMPS

```

@ 3, 35 SAY "Course completed (Y/N):"
@ 3, 59 GET F1556->COMPLETED
@ 5, 9 SAY "Reason for Noncompletion (F,C,W,T,I):"
@ 5, 48 GET F1556->NCOMPLCODE
@ 7, 9 SAY "Dates: 1556 Received....."
@ 7, 51 GET F1556->DTREC1556
@ 9, 17 SAY "Letter of Agreement Sent....."
@ 9, 51 GET F1556->DTLTRAGR
@ 11, 17 SAY "Copy of 1556 Sent to 002....."
@ 11, 51 GET F1556->DTCOPY002
@ 13, 17 SAY "Copy of 1556 Sent to Souces....."
@ 13, 51 GET F1556->DTCOPYSCR
@ 15, 17 SAY "Copy of 1556 Sent to Department.."
@ 15, 51 GET F1556->DTCOPYDEPT
@ 17, 17 SAY "Evaluation Sent to Department...."
@ 17, 51 GET F1556->DTEVALDEPT
@ 0, 5 TO 18, 71 DOUBLE

```

* TRAINEE.FMT

```

@ 1, 22 SAY "SECTION A -- TRAINEE INFORMATION"
@ 4, 3 SAY "Last Name:"
@ 4, 14 GET F1556->ELASTNAME
@ 4, 35 SAY "First Name:"
@ 4, 47 GET F1556->EFIRSTNAME
@ 4, 58 SAY "Middle Initial:"
@ 4, 74 GET F1556->EMINITIAL
@ 6, 3 SAY "SSN:"
@ 6, 8 GET F1556->ESSN
@ 6, 23 SAY "UIC:"
@ 6, 28 GET F1556->EUIC
@ 6, 39 SAY "Supervisory Code (O,S,M,X):"
@ 6, 67 GET F1556->ESUPVCODE
@ 8, 3 SAY "Department Code:"
@ 8, 20 GET F1556->EDEPTCODE
@ 8, 30 SAY "Continuous Federal Service (YYMM):"
@ 8, 65 GET F1556->EFEDSERV
@ 10, 3 SAY "Position Title:"
@ 10, 19 GET F1556->ETITLE
@ 10, 54 SAY "Handicapped (Y/N):"
@ 10, 73 GET F1556->EHANDICAP
@ 12, 3 SAY "Pay Plan (SA,ST,GS,GM,AD,WG,WS):"
@ 12, 36 GET F1556->EPAYPLAN
@ 12, 42 SAY "Pay Series:"
@ 12, 54 GET F1556->ESERIES
@ 14, 3 SAY "Pay Grade:"
@ 14, 14 GET F1556->EPAYGRADE
@ 14, 21 SAY "Type of Appointment (T1,T2,C1,C2):"
@ 14, 56 GET F1556->EAPPTMT
@ 16, 3 SAY "Schedule (F,P,I):"
@ 16, 21 GET F1556->ESCHEDULE
@ 16, 26 SAY "Sex (M,F):"
@ 16, 37 GET F1556->ESEX
@ 16, 42 SAY "Ethnic Code (C,B,N,A,H):"
@ 16, 67 GET F1556->ETHNICGRP

```

@ 0, 1 TO 17, 77 DOUBLE
@ 2, 2 TO 2, 76

* TRNCOURS. FMT

@ 1, 22 SAY "SECTION B -- TRAINING COURSE DATA"
@ 3, 4 SAY "Source Uic:"
@ 3, 16 GET F1556->SUIC
@ 3, 23 SAY "Source Name:"
@ 3, 36 GET F1556->SNAME
@ 5, 4 SAY "Source City:"
@ 5, 17 GET F1556->SCITY
@ 5, 42 SAY "Source State:"
@ 5, 56 GET F1556->SSTATE
@ 7, 4 SAY "Course City:"
@ 7, 17 GET F1556->CCITY
@ 7, 42 SAY "Course State:"
@ 7, 56 GET F1556->CSTATE
@ 9, 4 SAY "Training Type (U,V,M,N,S,A,C,I,E,X,O):"
@ 9, 43 GET F1556->TRNGTYPE
@ 11, 4 SAY "Course Title:"
@ 11, 18 GET F1556->CTITLE
@ 11, 50 SAY "Course Number:"
@ 11, 65 GET F1556->CNUMBER
@ 13, 4 SAY "Dates: (Start:"
@ 13, 19 GET F1556->CSTARTDT
@ 13, 29 SAY "End:"
@ 13, 34 GET F1556->CENDDT
@ 13, 42 SAY ") Hours: (Duty:"
@ 13, 58 GET F1556->CDUTYHRS
@ 13, 63 SAY "Nonduty:"
@ 13, 72 GET F1556->CNONDUTYHR
@ 13, 75 SAY ")"
@ 15, 4 SAY "Purpose (1,2,3):"
@ 15, 21 GET F1556->CPURPOSE
@ 15, 27 SAY "Priority (1,2,3):"
@ 15, 45 GET F1556->CPRIORITY
@ 15, 52 SAY "Method (C,R,F):"
@ 15, 68 GET F1556->CMETHOD
@ 17, 4 SAY "Source (GA,GN,GC,GF,GM,GG,GO,GI,GS,GC,LN,LP,OR,OT):"
@ 17, 56 GET F1556->STYPE
@ 19, 4 SAY "Training Program (V,U,M,S):"
@ 19, 32 GET F1556->ETRNGPLAN
@ 19, 38 SAY "Reason for Selection (C,T,O,Q):"
@ 19, 70 GET F1556->CREASON
@ 0, 2 TO 20, 77 DOUBLE
@ 2, 3 TO 2, 76

* BILLINGS. FMT

@ 2, 18 SAY "SECTION C -- COSTS AND BILLING INFORMATION"
@ 5, 5 SAY "Payment Method (B,T,R):"
@ 5, 29 GET F1556->PAYMETHOD
@ 5, 37 SAY "Total Direct Costs:"
@ 5, 57 GET F1556->DIRCOSTS
@ 7, 5 SAY "Funding Source (R,O):"

```

@ 7, 27 GET F1556->FUNDSOURCE
@ 7, 30 SAY "TANGO Number:"
@ 7, 44 GET F1556->TANGONO
@ 7, 53 SAY "Job Order Number:"
@ 7, 71 GET F1556->JOBORDNO
@ 9, 5 SAY "Total Indirect Costs:"
@ 9, 27 GET F1556->INDIRCOSTS
@ 9, 42 SAY "Travel Order Number:"
@ 9, 63 GET F1556->TRAVORDNO
@ 11, 5 SAY "Government Deduction (Y,N):"
@ 11, 33 GET F1556->GOVDEDUC
@ 1, 2 TO 13, 77 DOUBLE
@ 3, 3 TO 3, 76

```

* INVOICE.FMT

```

@ 1, 31 SAY "INVOICE TRACKING"
@ 5, 21 SAY "Log Number:"
@ 5, 34 GET F1556->LOGNUM
@ 7, 21 SAY "Invoice Number:"
@ 7, 38 GET F1556->INVOICENO
@ 9, 21 SAY "Date Invoice Received:"
@ 9, 45 GET F1556->DTINVOIRCV
@ 11, 21 SAY "Date Invoice Forwarded to 002:"
@ 11, 53 GET F1556->DTINVOIFWD
@ 0, 13 TO 14, 69 DOUBLE
@ 2, 14 TO 2, 68

```

* CHGLOG.FMT

```

@ 1, 9 SAY "Log Number:"
@ 1, 22 GET F1556->LOGNUM
@ 1, 35 SAY "Fiscal Year of Course:"
@ 1, 59 GET F1556->FYCRS
@ 3, 9 SAY "Multiple Employees:"
@ 3, 29 GET F1556->MULEMPS
@ 3, 35 SAY "Course completed (Y/N):"
@ 3, 59 GET F1556->COMPLETED
@ 5, 9 SAY "Reason for Noncompletion (F,C,W,T,I):"
@ 5, 48 GET F1556->NCOMPLCODE
@ 7, 9 SAY "Dates: 1556 Received....."
@ 7, 51 GET F1556->DTREC1556
@ 9, 17 SAY "Letter of Agreement Sent....."
@ 9, 51 GET F1556->DTLTRAGR
@ 11, 17 SAY "Copy of 1556 Sent to 002....."
@ 11, 51 GET F1556->DTCOPY002
@ 13, 17 SAY "Copy of 1556 Sent to Souces....."
@ 13, 51 GET F1556->DTCOPYSCR
@ 15, 17 SAY "Copy of 1556 Sent to Department.."
@ 15, 51 GET F1556->DTCOPYDEPT
@ 17, 17 SAY "Evaluation Sent to Department...."
@ 17, 51 GET F1556->DTEVALDEPT
@ 0, 5 TO 18, 71 DOUBLE
@ 20, 10 SAY 'Input New Information Now. Press CTRL-W When Done.'

```

* CHGTRAIN.FMT

```

@ 1, 22 SAY "SECTION A -- TRAINEE INFORMATION"
@ 4, 3 SAY "Last Name: "
@ 4, 14 GET F1556->ELASTNAME
@ 4, 35 SAY "First Name: "
@ 4, 47 GET F1556->EFIRSTNAME
@ 4, 58 SAY "Middle Initial: "
@ 4, 74 GET F1556->EMINITIAL
@ 6, 3 SAY "SSN: "
@ 6, 8 GET F1556->ESSN
@ 6, 23 SAY "UIC: "
@ 6, 28 GET F1556->EUIC
@ 6, 39 SAY "Supervisory Code (O,S,M,X): "
@ 6, 67 GET F1556->ESUPVCODE
@ 8, 3 SAY "Department Code: "
@ 8, 20 GET F1556->EDEPTCODE
@ 8, 30 SAY "Continuous Federal Service (YYMM): "
@ 8, 65 GET F1556->EFEDSERV
@ 10, 3 SAY "Position Title: "
@ 10, 19 GET F1556->ETITLE
@ 10, 54 SAY "Handicapped (Y/N): "
@ 10, 73 GET F1556->EHANDICAP
@ 12, 3 SAY "Pay Plan (SA,ST,GS,GM,AD,WG,WS): "
@ 12, 36 GET F1556->EPAYPLAN
@ 12, 42 SAY "Pay Series: "
@ 12, 54 GET F1556->ESERIES
@ 14, 3 SAY "Pay Grade: "
@ 14, 14 GET F1556->EPAYGRADE
@ 14, 21 SAY "Type of Appointment (T1,T2,C1,C2): "
@ 14, 56 GET F1556->EAPPTMT
@ 16, 3 SAY "Schedule (F,P,I): "
@ 16, 21 GET F1556->ESCHEDULE
@ 16, 26 SAY "Sex (M,F): "
@ 16, 37 GET F1556->ESEX
@ 16, 42 SAY "Ethnic Code (C,B,N,A,H): "
@ 16, 67 GET F1556->ETHNICGRP
@ 0, 1 TO 17, 77 DOUBLE
@ 2, 2 TO 2, 76
@ 19,10 SAY 'Input New Information Now. Press CTRL-W When Done.'

```

* CHGCOURS.FMT

```

@ 1, 22 SAY "SECTION B -- TRAINING COURSE DATA"
@ 3, 4 SAY "Source Uic: "
@ 3, 16 GET F1556->SUIC
@ 3, 23 SAY "Source Name: "
@ 3, 36 GET F1556->SNAME
@ 5, 4 SAY "Source City: "
@ 5, 17 GET F1556->SCITY
@ 5, 42 SAY "Source State: "
@ 5, 56 GET F1556->SSTATE
@ 7, 4 SAY "Course City: "
@ 7, 17 GET F1556->CCITY
@ 7, 42 SAY "Course State: "
@ 7, 56 GET F1556->CSTATE
@ 9, 4 SAY "Training Type (U,V,M,N,S,A,C,I,E,X,O): "

```

```

@ 9, 43 GET F1556->TRNGTYPE
@ 11, 4 SAY "Course Title:"
@ 11, 18 GET F1556->>CTITLE
@ 11, 50 SAY "Course Number:"
@ 11, 65 GET F1556->>CNUMBER
@ 13, 4 SAY "Dates: (Start:"
@ 13, 19 GET F1556->CSTARTDT
@ 13, 29 SAY "End:"
@ 13, 34 GET F1556->CENDDT
@ 13, 42 SAY ") Hours: (Duty:"
@ 13, 58 GET F1556->CDUTYHRS
@ 13, 63 SAY "Nonduty:"
@ 13, 72 GET F1556->CNONDUTYHR
@ 13, 75 SAY ")"
@ 15, 4 SAY "Purpose (1,2,3):"
@ 15, 21 GET F1556->CPURPOSE
@ 15, 27 SAY "Priority (1,2,3):"
@ 15, 45 GET F1556->CPRIORITY
@ 15, 52 SAY "Method (C,R,F):"
@ 15, 68 GET F1556->CMETHOD
@ 17, 4 SAY "Source (GA,GN,GC,GF,GM,GG,GO,GI,GS,GC,LN,LP,OR,OT):"
@ 17, 56 GET F1556->>STYPE
@ 19, 4 SAY "Training Program (V,U,M,S):"
@ 19, 32 GET F1556->ETRNGPLAN
@ 19, 38 SAY "Reason for Selection (C,T,O,Q):"
@ 19, 70 GET F1556->CREASON
@ 0, 2 TO 20, 77 DOUBLE
@ 2, 3 TO 2, 76
@ 21,10 SAY 'Input New Information Now. Press CTRL-W When Done.'

```

* CHGBILL.FMT

```

@ 2, 18 SAY "SECTION C -- COSTS AND BILLING INFORMATION"
@ 5, 5 SAY "Payment Method (B,T,R):"
@ 5, 29 GET F1556->PAYMETHOD
@ 5, 37 SAY "Total Direct Costs:"
@ 5, 57 GET F1556->DIRCOSTS
@ 7, 5 SAY "Funding Source (R,O):"
@ 7, 27 GET F1556->FUNDSOURCE
@ 7, 30 SAY "TANGO Number:"
@ 7, 44 GET F1556->TANGONO
@ 7, 53 SAY "Job Order Number:"
@ 7, 71 GET F1556->JOBORDNO
@ 9, 5 SAY "Total Indirect Costs:"
@ 9, 27 GET F1556->INDIRCOSTS
@ 9, 42 SAY "Travel Order Number:"
@ 9, 63 GET F1556->TRAVORDNO
@ 11, 5 SAY "Government Deduction (Y,N):"
@ 11, 33 GET F1556->GOVDEDUC
@ 1, 2 TO 13, 77 DOUBLE
@ 3, 3 TO 3, 76
@ 15,10 SAY 'Input New Information Now. Press CTRL-W When Done.'

```

* CHGINVOI.FMT

@ 1, 31 SAY "INVOICE TRACKING"
@ 5, 21 SAY "Log Number: "
@ 5, 34 GET F1556->LOGNUM
@ 7, 21 SAY "Invoice Number: "
@ 7, 38 GET F1556->INVOICENO
@ 9, 21 SAY "Date Invoice Received: "
@ 9, 45 GET F1556->DTINVOIRCV
@ 11, 21 SAY "Date Invoice Forwarded to 002:"
@ 11, 53 GET F1556->DTINVOIFWD
@ 0, 13 TO 14, 69 DOUBLE
@ 2, 14 TO 2, 68
@ 16,14 SAY 'Input New Information Now. Press CTRL-W When Done.'

LIST OF REFERENCES

1. Dehning, W., Essig, H., and Maass, S., "The Adaptation of Virtual Man-Computer Interfaces to User Requirements in Dialogs," *Lecture Notes in Computer Science*, vol. 10, 1981.
2. Stahl, B., "Friendly Mainframe Software Guides Users Toward Productivity," *Computerworld*, vol. 20, no. 5, February 3, 1986, pp. 53-66.
3. Carter, J.A., Jr., "A Taxonomy of User-Oriented Functions," *International Journal of Man-Machine Studies*, vol. 24, no. 3, March 1986, pp. 195-292.
4. Trenner, L., "How to Win Friends and Influence People: Definitions of User-Friendliness in Interactive Computer Systems," *Journal of Information Science*, vol. 13, 1987, pp. 99-107.
5. Benbaset, I., "The Impact of Cognitive Styles on Information System Design," *MIS Quarterly*, vol. 2, 1978, pp. 43-54.
6. Callahan, E., "User-Friendly Software: The Role of the User," *Proceedings, Fifth Interservice Industry Training Equipment Conference*, November 14-16, 1983, pp. 312-319.
7. Galitz, W., "Human Engineering in Screen Design," *Journal of Systems Management*, May 1983, pp. 6-11.
8. Houghton, R.C., Jr., "Designing User Interfaces: A Key to System Success," *Journal of Information Systems Management*, vol. 3, no. 3, Summer 1986, pp. 56-62.
9. Hatcher, M.E. and Diebert, T.R., "Predicting End-User Acceptance of Microcomputers in the Workplace," *International Journal of Man-Machine Studies*, vol. 26, no. 6, June 1987, pp. 695-705.

10. Lippert, F.G., "Introducing Change," *Supervision*. August 1981, pp. 18-19.
11. Jones, C.G., "Computer: Job Gobbler or Helpmate?," *Journal of Systems Management*, November 1981, p. 27.
12. Gross, P.N., "Flushing Out the Fear of Computing," *Data Management*, June 1983, p. 34.
13. Raymond, L., "Validating and Applying User Satisfaction as a Measure of MIS Success in Small Organizations," *Information and Management*, vol. 12, 1987, pp. 173-179.
14. Fryberger, R., "Developing an Online Interactive System for the Naive User," *1981 Western Educational Computing Conferene*, California Educational Computing Consortium, November 1981.
15. Nesdore, P., "Friendly to All or None," *Computerworld: In Depth*, October 17, 1981.
16. Shneiderman, B., "Human Factors Experiments in Designing Interactive Systems," *Computer*, December 1983, pp. 9-20.
17. Martin, M.P. and Fuerst, W.L., "Using Computer Knowledge in the Design of Interactive Systems," *Internation Journal of Man-Machine Studiès*, vol. 26, no. 3, March 1987, pp. 333-342.
18. Fried, L., "Nine Principles for Ergonomic Software," *Datamation*, vol. 28, no. 12, November 1982, pp. 9-17.
19. Gaines, B.R., "The Technology of Interaction--Dialogue Programming Rules," *International Journal of Man-Machine Studies*, vol. 14, no. 1, January 1981, pp. 136-146.

20. Knittle, D.L., Ruth, S., and Gardner, E.P., "Establishing User-Centered Criteria for Information Systems: A Software Ergonomics Perspective," *Information and Management*, vol. 11, 1987, pp. 163-172.
21. Schmidt, P., "What's New in Computer Psychology," *New York Times*, vol. 134, section F, January 20, 1985, p. 15.
22. Morland, D.J., "Human Factors Guidelines for Terminal Interface Design," *Communications of the ACM*, vol. 26, no. 7, July 1983, pp. 484-494.
23. Warner, J., "Designing User-Friendly Graphics Software," *Hardcopy*, July 1984, p. 180.
24. Chafin, R.L., "Human Factors in Applications Programming," *Infosystems*, vol. 30, June 1983, pp. 102-104.
25. Dean, M., "How a Computer Should Talk to People," *IBM Systems Journal*, vol. 21, no. 4, 1982, pp. 424-453.
26. Dunsmore, H.E., "Using Formal Grammar to Predict the Most Useful Characteristics of Interactive Systems," *Office Automation Conference Digest*, San Francisco, April 5-7, 1982, pp. 53-56.
27. Arthur, J.D., "Toward a Formal Specification of Menu-Based Systems," *Journal of Systems and Software*, vol. 7, 1987, pp. 73-82.
28. Shneiderman, B., *Designing the User Interface: Strategies for Effective Human-Computer Interaction*, Addison-Wesley Publishing Company, 1987.
29. Page-Jones, M., *The Practical Guide to Structured Systems Design*, Yourdon Press, 1980.
30. McMenamin, S.M., and Palmer, J.F., *Essential Systems Analysis*, Yourdon Press, 1984.

31. Stevens, W.P., Myers, G.J., and Constantina, L.L., "Structured Design," *IBM Systems Journal*, vol. 13, no. 2, 1974, pp. 115-139.
32. Davis, W.S., *Systems Analysis and Design: A Structured Approach*, Addison-Wesley Publishing Company, 1983.
33. Kerlinger, F., *Foundations of Behavior Research*, Holt, Rinehart, and Winston, 1973.
34. Shneiderman, B., *Designing the User Interface: Strategies for Effective Human-Computer Interaction*, Addison-Wesley Publishing Company, 1987.
35. Babbie, E.R., *The Practice of Social Research*, 4th edition, Wadsworth Publishing Co., 1987.
36. Flower, F.J., *Survey Research Methods*, Sage Publications, Inc., 1987.
37. Buzzell, R.E., Cox, D.F., and Brown, R.V., *Marketing Research and Information Systems*, McGraw-Hill, Inc., 1969.
38. Ogden, W.C., and Boyle, J.M., "Evaluating Human Computer Dialog Styles: Command vs. Form Fill-In for Report Modification," *Proceedings of the Human Factors Society*, 26th Annual Meeting, Human Factors Society, 1982, pp. 542-545.

INITIAL DISTRIBUTION LIST

	No. Copies
1. Defense Technical Information Center Cameron Station Alexandria, VA 22304-6145	2
2. Library, Code 0142 Naval Postgraduate School Monterey, CA 93943-5002	2
3. Director, Information Systems (OP-945) Office of the Chief of Naval Operations Navy Department Washington, DC 20350-2000	1
4. Superintendent, Naval Postraduate School Computer Technology Programs, Code 37 Monterey, CA 93943-5000	1
5. Dr. T. R. Sivasankaran, Department of MIS and Accounting California State University 18111 Nordhoff Street Northridge, CA 91330	1
6. LTC Gerald L. Pauler, USA, Code 55PA Naval Postgraduate School Monterey, CA 93943-5000	1
7. LCDR Sharon Slominski, USN c o Mr. John W. Ranes 1842 Glen Ridge Road Baltimore, MD 21234	2
8. LT Ivon Young, Sr., SC, USN c o Mr. F. Marcel Young 71 North 705 West Orem, UT 84057	2