Research Note 81-22

EARLY TRAINING ESTIMATION SYSTEM: FINAL YEARLY REPORT NUMBER 1

Lawrence H. O'Brien, Gavin H. Livingston
DYNAMICS RESEARCH CORPORATION

ARI FIELD UNIT AT FORT BLISS, TEXAS

U. S. Army
Research Institute for the Behavioral and Social Sciences

August 1980

Approved for public release; distribution unlimited.

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: Research Note 81-22	2. GOVT ACCESSION NO.: AD-A126365
3. RECIPIENT'S CATALOG NUMBER	
4. TITLE (and Subtitle): EARLY TRAINING ESTIMATION SYSTEM: FINAL YEARLY REPORT NUMBER 1	5. TYPE OF REPORT & PERIOD COVERED
	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s): Lawrence H. O'Brien, Gavin H. Livingston	8. CONTRACT OR GRANT NUMBER(s): MDA-903-80-C-0525
9. PERFORMING ORGANIZATION NAME AND ADDRESS: DYNAMICS RESEARCH CORPORATION, 60 Concord Street, Wilmington, MA 01887	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: 2Q162722A791
11. CONTROLLING OFFICE NAME AND ADDRESS: US Army Research Institute for the Behavioral and Social Sciences Field Unit, Fort Bliss, Texas 79916	12. REPORT DATE: August 1980
	13. NUMBER OF PAGES: 277
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)	15. SECURITY CLASS. (of this report): Unclassified
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited

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

18. SUPPLEMENTARY NOTES

This contract was monitored technically by Dr. Charles Jorgensen, ARI Field Unit Fort Bliss, Texas.

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

Training
Instructional system development
Data base management
Task analysis
Training estimation
Simulation

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

This report describes the major products developed under the first year of the Early Training Estimation System (ETES) development project.

The Early Training Estimation System (ETES) is designed to deal with two major deficiencies in existing technologies: (1) the lack of a systematic tool for describing, storing, and updating system concepts and for transmitting this information to all of the various participants in

DD FORM 1473, 1 JAN 73 — EDITION OF 1 NOV 65 IS OBSOLETE

Item 20 (Cont'd)

the development/acquisition process and (2) the lack of a comprehensive set of training analysis tools which are appropriate for the early phases of design. The ETES will have four major components; a System Description Technology, training estimation aids and procedures, human performance simulation models, and a users guide.

The System Description Technology (SDT) will be an automated tool for describing actual and projected system elements, including functional requirements, design concepts, tasks, skills, training program elements and their associated resources; for storing the above information; for changing and updating this information; and for transmitting the information among all of the participants in the acquisition process.

The training estimation aids and procedures will be specifically designed for early training estimation. They will include procedures (automated whenever possible) for (1) identifying comparable equipments, (2) generating and modifying tasks, (3) generating and modifying courses, (4) selecting and assigning tasks to training settings and methods, (5) determining the number of personnel to be trained, (6) determining training resources, and (7) developing training cost measures.

The human performance - system performance simulation models will be used to relate human task performance to system performance. The simulation models will provide the capability for trading off training-related system elements with other system elements.

The User's Guide will provide a detailed, step-by-step- handbook describing the use of the other three tools to assess early training requirements.

The first year of the study concentrated on the development of the SDT, the most important component of ETES. The SDT will provide a data base management tool which will be capable of describing most of the major elements of an emerging system. As such, the SDT will provide an important data base management capability that has wide ranging applicability, far beyond training-related issues.

This yearly report outlines specifications for the SDT development, provides a description of the physical and operational features of a prototype SDT concept, and describes the analytical procedures underlying the development of this concept.

Research Note 81-22

EARLY TRAINING ESTIMATION SYSTEM: FINAL YEARLY REPORT NUMBER 1

Lawrence H. O'Brien, Gavin H. Livingston
DYNAMICS RESEARCH CORPORATION

ARI FIELD UNIT AT FORT BLISS, TEXAS

U.S. ARMY RESEARCH INSTITUTE FOR THE BEHAVIORAL AND SOCIAL SCIENCES
5001 Eisenhower Avenue, Alexandria, Virginia 22333

Office, Deputy Chief of Staff for Personnel
Department of the Army

August 1980

Army Project Number
2Q162722A791

Manpower, Personnel
and Training

TABLE OF CONTENTS

		Page
PREFACE		xi
1. SUMMARY		1-1
1.1	Background	1-1
1.2	Current Problems Surrounding Early Training Estimation	1-2
1.3	ETES Components	1-5
	1.3.1 System Description Technology	1-5
	1.3.2 Training Estimation Aids and Procedures	1-14
	1.3.3 Human Simulation Models	1-15
	1.3.4 User's Guide	1-16
1.4	ETES Study Tasks	1-16
1.5	Progress on Study Tasks	1-16
	1.5.1 Task 1: Review of Existing Procedures	1-16
	1.5.2 Task 2: Develop SDT	1-20
	1.5.3 Tasks 3, 4 and 5	1-21
2. SPECIFICATIONS FOR SDT		2-1
2.1	Overview of SDT Requirements	2-2
	2.1.1 Role of SDT in the Acquisition Process	2-5
	2.1.2 A Basic Data Problem in Early Training Estimation	2-6
2.2	Overview of SDT Functions	2-10
	2.2.1 Functional Requirements	2-12
	2.2.2 Design Concepts	2-17
	2.2.3 Equipment-Task Interface	2-20
	2.2.4 Behavioral Task Elements and Features	2-25
	2.2.5 Skills and Knowledges	2-28
	2.2.6 Training Program Elements	2-28

TABLE OF CONTENTS (continued)

			Page
	2.3	General Guidelines for Input/Output Mechanisms	2-28
	2.4	Sequence of SDT Applications Through the Acquisition Process	2-35
		2.4.1 Period 1: Initial Functional Requirement Analysis	2-36
		2.4.2 Period 2: Initial Training Estimation -- Contractor Design Alternative Not Specified	2-38
		2.4.3 Period 3: Training Estimation for Identified Design Concepts	2-40
		2.4.4 Period 4: Training Estimation for Identified Tasks	2-41
		2.4.5 Period 5: Training Development for Selected System	2-42
	2.5	Past Efforts in Developing System-Specific Data Bases	2-43
		2.5.1 Logistics Support Analysis Record	2-43
		2.5.2 Air Force Human Resource Lab Unified Data Base	2-48
		2.5.3 Consolidated Data Base of HARDMAN Methodology	2-52
		2.5.4 SAT Program for the B-1 Bomber	2-54
		2.5.5 Navy Enlisted Professional Development Information Support System (NEPDISS)	2-56
		2.5.6 Other Data Bases	2-57
3.	SELECTION OF AN AUTOMATED TOOL FOR SDT DEVELOPMENT		3-1
	3.1	Overview of Automated Tools	3-1
	3.2	Review of Requirements Analysis Tools	3-2
	3.3	Review of Data Base Management Systems	3-9
		3.3.1 Overview of Data Base Management Systems	3-10
		3.3.2 Types of Data Base Management Systems	3-14
	3.4	The Application of DBMS Technology to the SDT	3-19

TABLE OF CONTENTS (continued)

	Page
3.5 Restructuring of the SDT into a Relational Framework	3-21
3.5.1 Discussion of the SDT from an Entity-Attribute-Relationship Perspective	3-21
3.6 Selection of a DBMS for the SDT	3-26
3.6.1 Determination of the SDT Requirements that Apply to the Selection of a DBMS	3-26
3.6.2 Selection of DBMSs that Fulfill the Requirements of the SDT	3-28
3.6.3 Comparison of the Selected DBMS Alternatives	3-33
3.6.4 Review of the Applicable DBMSs	3-37
3.6.5 Alternatives for Developing and Operating the SDT	3-39
3.6.6 Specific Recommendation for Developing the SDT	3-40
4. SDT DESCRIPTION	4-1
4.1 Overview of SDT Features and Relationship to Other Sections	4-2
4.2 Users of SDT	4-2
4.2.1 Primary Users	4-3
4.2.2 Data Base Directors (DBDs)	4-5
4.2.3 SDT Management Group	4-6
4.3 Physical Description of SDT	4-7
4.3.1 Physical Description of Primary User Hardware	4-7
4.3.2 DBDs Physical Equipment Description	4-10
4.3.3 SDT Management Group Physical Equipment Description	4-10

TABLE OF CONTENTS (continued)

			Page
4.4	Overview of SDT Processes		4-10
4.5	Modes of Operation for Primary Users and Data Base Directors		4-12
	4.5.1	Sign-On/System-Status Check	4-13
	4.5.2	System Examination	4-14
	4.5.3	Input Mode	4-14
	4.5.4	Update/Modify Mode	4-15
	4.5.5	Output Mode	4-16
4.6	Overview of SDT Operation		4-17
4.7	Example Interactions		4-17
4.8	Initial Version of SDT		4-61
	4.8.1	Characteristics of Initial SDT	4-61
	4.8.2	Expected Users	4-61
	4.8.3	Input Capabilities of SDT	4-61
	4.8.4	Output Capabilities of SDT	4-62
	4.8.5	System Elements Described in SDT	4-62
	4.8.6	Other Characteristics of the Initial SDT	4-63
REFERENCES			R-1
Appendix A:	Review of Army Documents and Procedures Related to ETES		A-1
Appendix B:	Example Outputs for Functional Requirements and Design Concepts		B-1
Appendix C:	Review of Psychological Research Related to Design		C-1
Appendix D:	Review of Research Related to Human Computer Interactions		D-1

LIST OF FIGURES

		Page
1-1	System Definition as a Function of Time	1-3
1-2	System Development Process for SDT	1-8
1-3	Overview of SDT Physical Structure	1-12
1-4	Overview of SDT Input/Output Operation	1-13
1-5	ETES Study Tasks	1-17
2-1	Etes Components	2-3
2-2	SDT Application	2-4
2-3	Role of SDT During Acquisition Process	2-7
2-4	System Development	2-9
2-5	SDT	2-11
2-6	Functional Requirements	2-13
2-7	High Level Functional Breakdown for Weapon Systems	2-16
2-8	Design Concepts	2-18
2-9	Sequence for Identifying Design Alternatives	2-22
2-10	Equipment-Task Interface	2-23
2-11	Tasks	2-26
2-12	Skills and Knowledge	2-29
2-13	Training Program Elements	2-31
2-14	Periods in the Application of SDT	2-37
3-1	An Architecture for a Database System	3-13
3-2	Relational DBMS Form	3-15
3-3	Network DBMS Form	3-17
3-4	Hierarchical DBMS Form	3-18
4-1	Overview of SDT Physical Characteristics	4-8
4-2	Overview of SDT Processes	4-11
4-3	Schematic Representation of SDT Frames	4-19

LIST OF TABLES

		Page
1-1	ETES Study Activities and Related Sections of the Report	1-18
2-1	Outputs Related to Functional Requirements	2-14
2-2	Outputs Related to Design Concepts	2-19
2-3	Levels of Design Concept Development	2-21
2-4	Outputs Related to Equipment-Task Interface	2-24
2-5	Outputs Related to Task Information	2-27
2-6	Outputs Related to Skills and Knowledge (S&K)	2-30
2-7	Outputs Related to Training Program Elements	2-32
2-8	Past Efforts at Human Resource Data Base Development	2-44
2-9	Overview of LSAR and Its Major Weaknesses	2-45
2-10	Limitations of the UDB	2-50
2-11	Data Elements Contained in CDB	2-53
2-12	SAT Data Elements and Limitations	2-55
2-13	NEPDISS Data Limitations	2-58
3-1	Requirements Methodologies and Results of Preliminary Assessment	3-5
3-2	Implicit Entity and Attribute Classes for Developing Weapon Systems	3-23
3-3	Types of Relationships Required by SDT	3-24
3-4	Implicit Relationships in SDT	3-25
3-5	Characteristics of Commercially-Available DBMS	3-29
3-6	Evaluation of Selected DBMS Alternatives for the SDT	3-35
4-1	Primary Users of SDT	4-4

PREFACE

This paper is the first yearly report for the Early Training Estimation System (ETES) development project (Contract No. MDA-903-80-C-0525). The report is divided into four sections. Section 1 provides an overview of the report, the ETES study components, study tasks, and the major activities that were conducted under these tasks during the first year of the study. The next three chapters describe the System Description Technology, the most important component of ETES. Section 2 presents a set of detailed specifications for the information elements which must be described by the SDT. Section 3 describes the results of an evaluation of current automated tools which were considered for application in the SDT. Section 4 presents a detailed description of the physical and operational characteristics of the SDT.

A number of different analyses and reviews were conducted during the first year of the study in support of the SDT development. These analyses are described in a series of appendices. Appendix A presents the results of a detailed review of existing Army acquisition procedures and practices and their implications for ETES. Appendix B describes some examples of the types of information which are likely to be output from the SDT. Appendix C presents the results of a review of psychological research related to design and its implications for the SDT. Appendix D reviews psychological research related to human computer interaction, an area of research closely related to the automated SDT.

The contract monitor for the study was Dr. Charles Jorgensen of the Army Research Institute Field Unit at Fort Bliss, Texas. The DRC program manager for the study was Dr. Lawrence H. O'Brien. The key contributors to DRC's first year efforts in ETES development were Cecil Wakelin, Gavin Livingstone, Ray Walsh, Peter Weddle, David Herlihy, Laurel Brown, Drs. Paul Ronco and Jack Hansen, and Gen. (Ret.) Paul Phillips.

SECTION 1 - SUMMARY

This section summarizes the activities and analyses conducted during the first year of the Early Training Estimation System (ETES) development project. The section is divided into five subsections. Subsection 1.1 reviews the general trends which are placing heavier and heavier demands on training development. Subsection 1.2 describes the specific problems and deficiencies in existing Army practices which led to the initiation of the ETES project. Subsection 1.3 presents an overview of the four components of ETES. Subsection 1.4 describes the major tasks in the ETES development project. Subsection 1.5 presents a detailed description of the progress achieved under each of these tasks during the first year of the study.

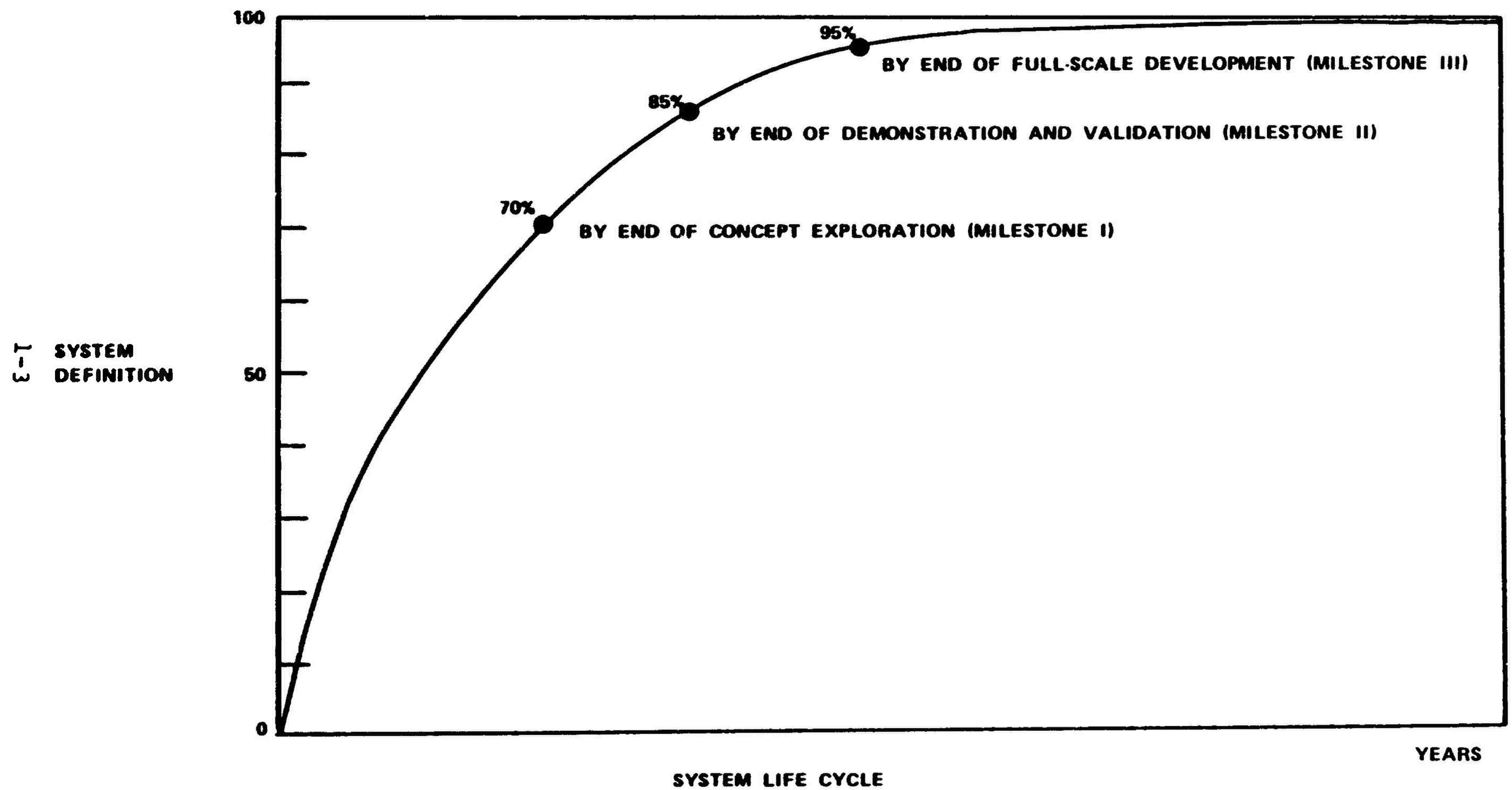
1.1 BACKGROUND

The Early Training Estimation System will provide a capability for systematically estimating training requirements during the earliest phases of the acquisition process (mission area analysis, concept exploration - Phase I, and validation and demonstration - Phase II). There are two major reasons why such early estimates of training requirements are needed. First, by developing earlier and more accurate estimates of training requirements, the training planning process can begin earlier, and thus the training products associated with a system, many of which require a long lead time, are more likely to be available when the system is fielded. Second, by developing estimates of training requirements for the various alternatives which are likely to exist during the early phases of the acquisition process, the training developer can provide the information needed to effectively influence design with training-related considerations. The importance of the latter utilization of early training projections cannot be overestimated. Most of the major design decisions related to a new system are made during the early phases of the acquisition process (see Figure 1-1). Thus, if training is to influence design, it must impact these early design decisions. And there is good reason for insuring that training-related considerations do, in fact, impact design. Studies have shown that, in most weapon systems, operation and support costs comprise 50 to 80 percent of total life cycle cost. Further, over 60 percent of these operation and support costs are related to manpower, including the cost of training. Because these costs are the result of demands generated by the design characteristics of a system, acquisition policies have been established in the Federal Government to insure that support requirements are accurately determined and evaluated in conjunction with system development (e.g., DoDD 5000.1, DODI 5000.2, and DODD 5000.39). ETES is specifically designed to provide the Army with the capability for meeting the training-related requirements in these new acquisition policies.

1.2 CURRENT PROBLEMS SURROUNDING EARLY TRAINING ESTIMATION

Given the clear needs for early training estimation which were outlined above, one might wonder why a systematic early training estimation tool has not yet been developed. There are two reasons for this current gap. First, the needs described in Section 1.1 have only recently been identified. Second, and most important, current procedures and practices have three major deficiencies which limit, and

FIGURE 1-1 SYSTEM DEFINITION AS A FUNCTION OF TIME



in most cases prohibit, the development of early estimates of training requirements. These deficiencies are:

(1) Lack of a Systematic Flow of Information Between Training Developers and Other Participants in the Acquisition Process - To develop estimates of training requirements, training developers must have information on actual or estimated system functional requirements and design concepts as soon as they are generated and, to maintain the accuracy of these estimates, these same training developers must be quickly informed of design changes and updates. Unfortunately, under current practices and procedures, training developers do not receive information on system functional requirements and design concepts in any systematic format, nor is there any formal mechanism through which they can obtain information on system updates.

(2) Lack of Estimation Procedures/Aids Appropriate to the Design Process - Even if training developers were receiving accurate and timely information on early system concepts, systematic estimates of training resources could not be developed because of the deficiencies in the current state of the art in training estimation procedures and aids. Current training technologies are geared to deal with the type of detailed data and the types of analytical questions which are relevant to later phases of the acquisition process. These technologies cannot deal with the special requirements of the early phases such as the

identification of comparable existing equipment, the generation of tasks for systems whose hardware has not yet been built, the rapid assignment of tasks, and the rapid estimation of training resources and costs.

(3) Lack of Simulation Models and Other Evaluative Technologies which Incorporate Human Performance. Currently, there is not an adequate set of simulation models which can be used to relate human task performance to overall system performance. Without such models, it is difficult to estimate some of the key interdisciplinary tradeoffs (e.g., training versus hardware) which must be made during the early phases of the acquisition process.

1.3 ETES COMPONENTS

To deal with the deficiencies described above and to develop a comprehensive set of early training estimation tools, the Army Research Institute (ARI) initiated a three-year effort to develop an Early Training Estimation System (ETES). The ETES will have four major components: a System Description Technology (SDT), Training Estimation Aids and Procedures, Human Performance Simulation Models, and a User's Guide.

1.3.1 System Description Technology (SDT)

The SDT will be an automated tool for describing actual and projected system elements, including functional requirements, design concepts, tasks, skills, training program elements, and their associated resources; for storing the above information; for changing and updating this information; and for transmitting the information among all of the participants in the acquisition process.

The SDT is clearly the most important component of ETES and will be given the greatest amount of attention and resources during development. In fact, the primary focus of the first year of the study efforts has been on the development of specifications for the SDT.

It should be noted that even though the SDT is being developed under the auspices of an early training estimation project, the SDT will provide a data base management tool which will be capable of describing most of the major elements of a system (e.g., functional requirements, design, tasks, skills, and training program elements). As such, the SDT will provide an important data base management capability that has wide ranging applicability beyond training related issues.

To provide an effective communication vehicle for training developers and other participants in the acquisition process, the SDT will describe (a) training programs and their associated resources, (b) the tasks which drive these training programs, (c) the personnel who will be required to perform the tasks, (d) the system designs which generate the task requirements, and (e) the functional requirements for which the system designs have been developed.

In order to provide a capability for early training requirements estimation, the SDT will describe these system elements during the earliest phases of the acquisition process. To systematically generate data during the early phases of the acquisition process, comparability analysis procedures will be employed.

More specifically, during the early phases of the acquisition process when only information on functional requirements is available, a systematic comparability analysis can be conducted to identify existing subsystems, and historical data for these subsystems can be modified to meet the differential requirements of the new system. By utilizing design and task data from comparable existing systems, systematic estimations of early training requirements can be made when only functional information on the projected system is available (see Figure 1-2). Later, as actual design concepts are developed, the comparability analyses can be used to develop estimates of tasks and training program elements. Still later, when the actual system tasks are available, only the training program elements must be estimated.

The SDT will thus be capable not only of describing the current state of the system during the earliest phases of the acquisition process, but also of (1) detailing projected system elements and alternative system concepts, (2) relating alternative system concepts to a common framework so that meaningful comparisons can be made, and (3) refining system information as more accurate and more detailed data are developed.

1.3.1.1 SDT as a Data Base Management Tool

An extensive review of automated tools was conducted during the first year of the ETES study to identify an extant technique or approach which would provide the best vehicle

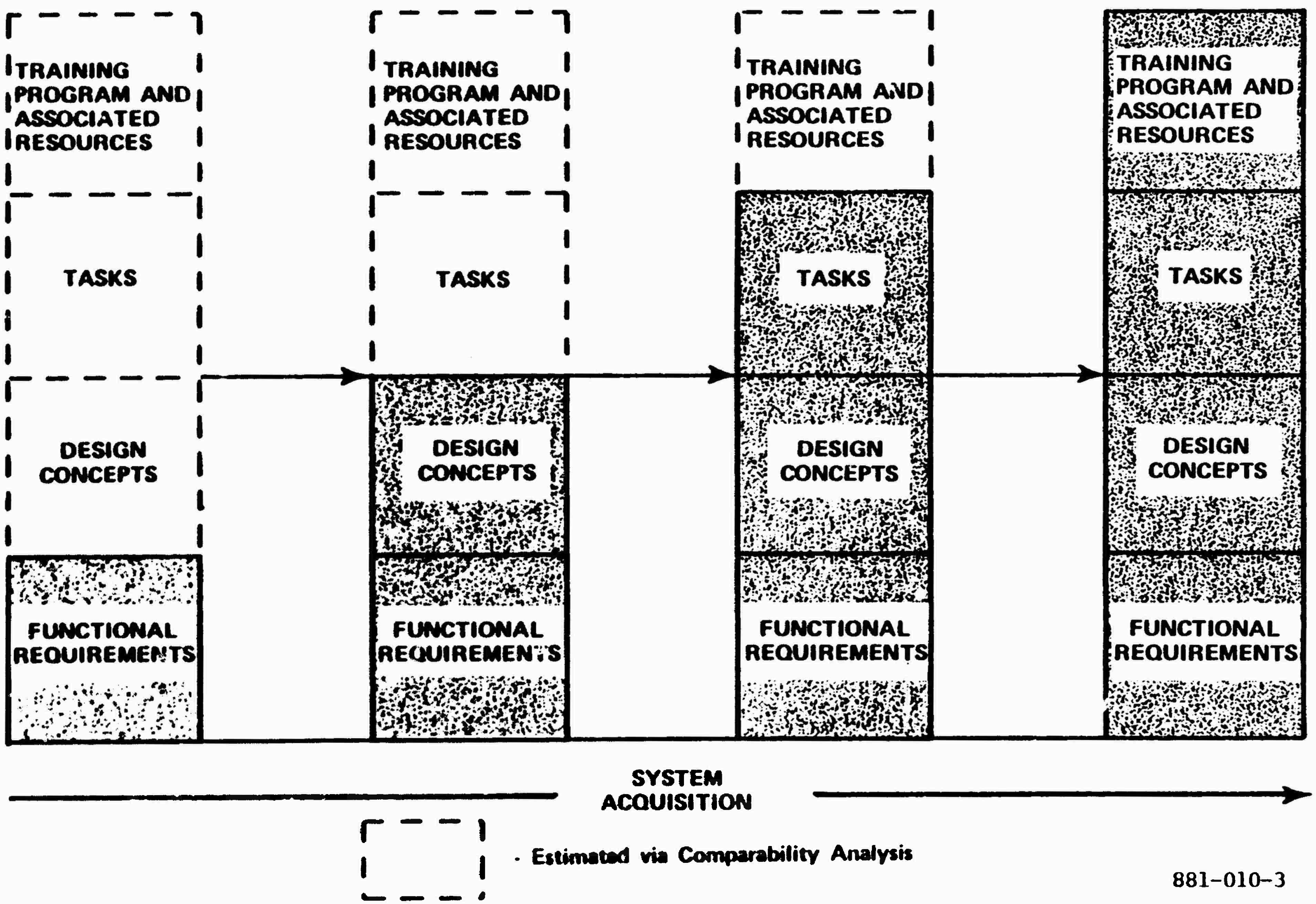


FIGURE 1-2 SYSTEM DEVELOPMENT PROCESS FOR SDT

for STD development. The results of this review indicated that a Data Base Management System (DBMS) could best fill the SDT requirements. The Data Base Management System concept has a number of advantages over other automated tools. First, DBMSs are specifically designed to deal with the types of issues which are central to the major problem facing the SDT - namely, the description, update, expansion and retrieval of data on an emerging system and the transmission of this information to a wide range of users. Second, DBMSs have the capability to be fitted with input/output mechanisms which are specifically geared for use by uninitiated users. Third, DBMSs can incorporate information on the implicit relationships and classes of information which are applicable to all weapon systems and these stored relationships can be used to reduce the input load on the user. Fourth, DBMSs can maintain a consistent internal data base while at the same time allowing different users to have different "views" of the stored data and different input and output requirements.

The centralized control provided by a DBMS can, in turn, (1) reduce redundancy in stored data, (2) avoid inconsistency in stored data, (3) allow for greater sharing of data, (4) permit standards to be enforced, (5) permit security restrictions to be applied, (6) permit a greater capability for checking and maintaining data, and (7) provide a capability for "data independence". Data independence is achieved by maintaining an internal structure of the data which is independent of the individual applications of the data and individual user viewpoints. This data independence may be contrasted with data dependent systems in which the data are stored and accessed in a manner which is dictated by the structure of the applications.

1.3.1.2 Users of SDT

Some of the organizations which are likely to be users of the SDT are the TRADOC system manager for a developing system, training developments (for the related school) combat developments, DARCOM Program Management staff for the developing system, the TRADOC Systems Analysis Activity (TRASANA), the DARCOM Materiel Readiness Support Activity (MRSA), and individual contractors associated with the System.

Each user will be connected to the SDT by at least one remote terminal. Some primary user organizations (e.g., training developments and the DARCOM Program Manager) are likely to have more than one terminal since they will have a number of individuals with a need for SDT data base information. It is expected that the users of the SDT will have little, if any, computer skills. Consequently, all of their interactions with the SDT will be through a highly transparent user interface which will utilize menu-selection, form-filling, and question-and-answer computer dialogue techniques to elicit input data and commands. This type of transparent interface will mean that the users will be required to learn only the commands associated with calling up the SDT system. From that point on, they will be led through the utilization of the SDT and will not have to generate any more commands on their own. (They should, of course, have read the SDT Users Manual to learn how the SDT can, and should, be used.)

One of the user groups will also serve as the Data Base Directors (DBDs). The DBDs will have the same capability as the primary users for entering, storing, and accessing SDT information. The Data Base Directors will also have two additional responsibilities: (1) The DBDs will be responsible for overseeing the general development of a system-specific SDT data base, and (2) The DBDs will have the capability, together with the SDT Management Group, for batch input and for producing block diagrams to represent various system relationships.

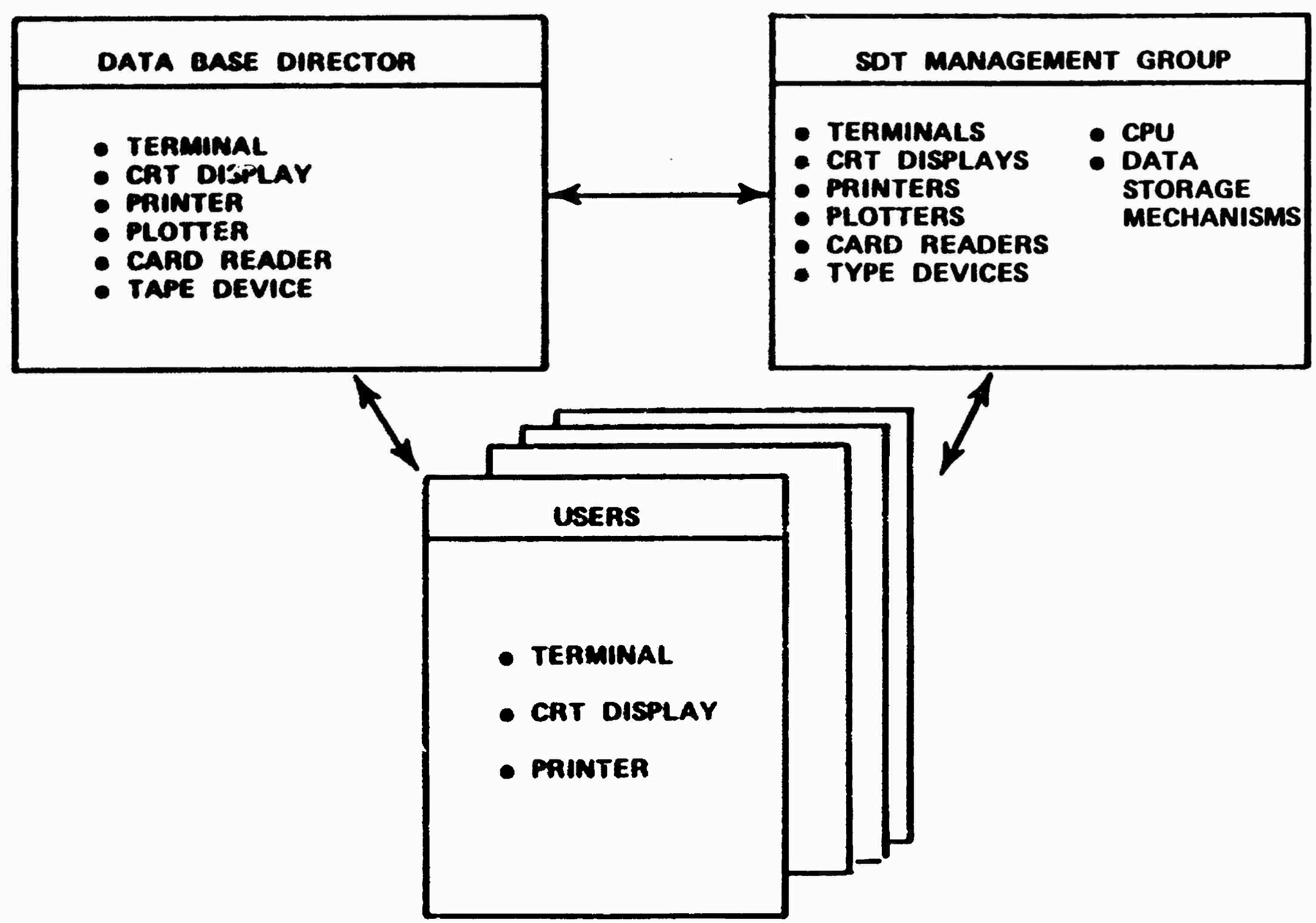
The SDT Management Group will be responsible for overseeing the application of the SDT on an Army-wide basis including the maintenance and update of the SDT data base programs relating to data input and output, data storage and retrieval and the DBMS external, conceptual, and internal models; operation of the central processor to handle SDT applications and direct its use among the various SDT users; assistance to users and DBDs in utilizing the SDT; and provision of data to other Army organizations for related applications (e.g., total force requirements analysis).

1.3.1.3 Physical Description of SDT

Figure 1-3 provides a general description of the SDT physical characteristics. The design outlined in Figure 1-3 is intended to minimize requirements for the purchase of new equipments by participating Army organizations.

1.3.1.4 Overview of SDT Processes

An overview of the general SDT processes is presented in Figure 1-4. The SDT will have the capability of inputting data in two different modes: batch input of SDT data sheets and acquisition data, and interactive input of SDT data sheets. Directions for the interactive input of data will



881-010-4

FIGURE 1-3 OVERVIEW OF SDT PHYSICAL STRUCTURE

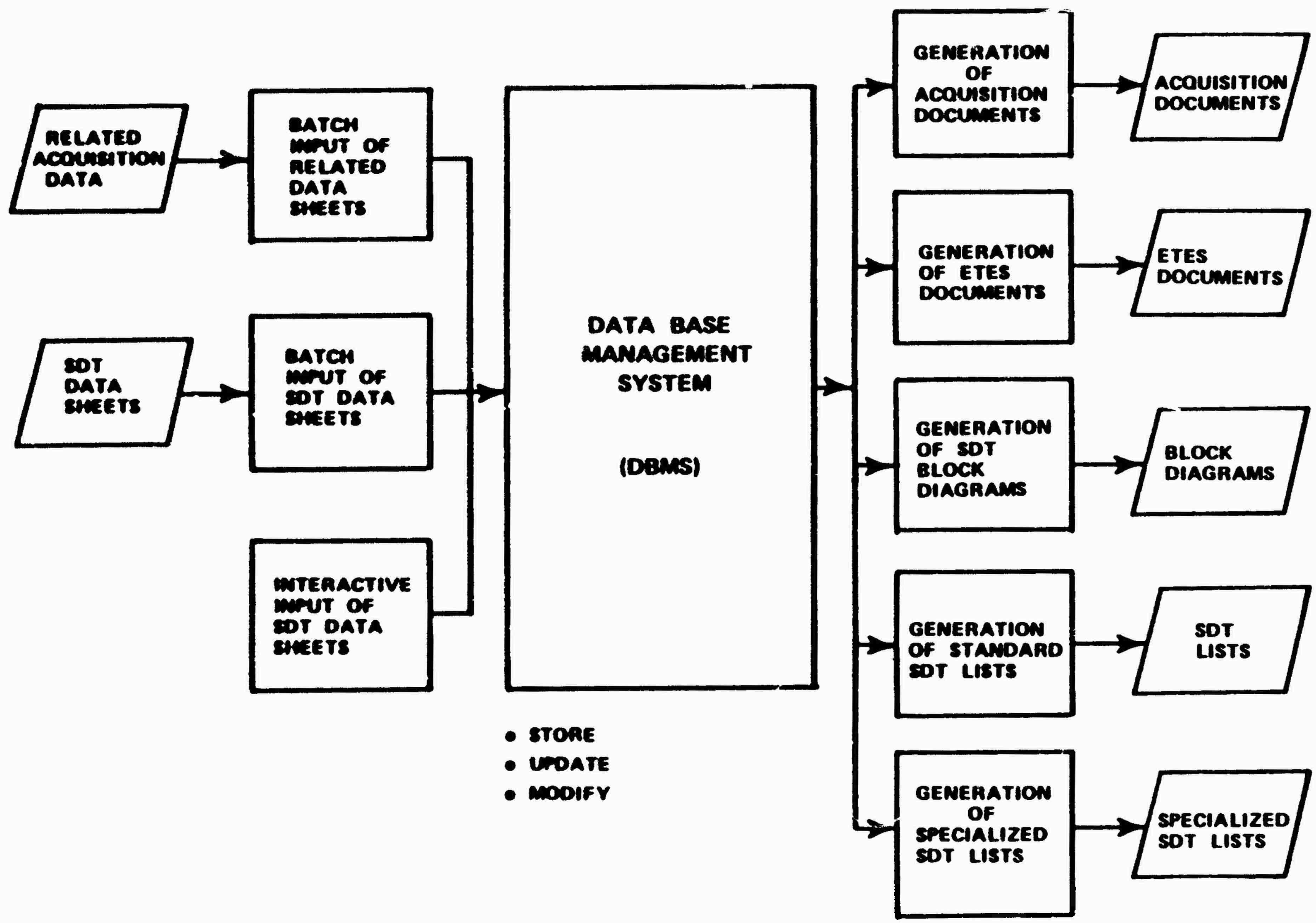


Figure 1-4 Overview of SDT Input/Output Operation

be provided by the data base director programs. The SDT input data will be translated into a form which matches an internal conceptual model contained in the DBMS. Once it has been translated, the data will be evaluated for consistency against data already in the data base and, if consistent, the data will be entered into the system-specific data base. However, this will be done only after the Data Base Directors have determined that the user has been cleared to enter that type of data into the data base. Once in the data base, the data are continuously updated, modified, and expanded. Direction of these changes is provided by the data base director programs. These same programs are used in selecting and generating output data. Five different formats for outputing the data will be available: specialized SDT lists, standard SDT lists, block diagrams, output formatted for input into other ETES procedures, and output formatted to correspond to the format requirements of specific acquisition documents.

Once the user enters the SDT, he will have option of entering four possible modes of operation: (1) system examination - this mode is used to examine data which is currently in the data base; (2) input - this mode is used to input data; (3) update/modify - this mode is used to eliminate or modify data already in the data base; and (4) output - this mode is used to obtain a hard copy output of elements in the data base.

1.3.2 Training Estimation Aids and Procedures

These aids and procedures can be divided into two general groups: training data generation techniques and training estimation techniques. The data generation techniques are procedures for identifying comparable equipments, generating and modifying tasks, and generating and modifying courses. The training estimation techniques include procedures for selecting and assigning tasks to training settings and methods, determining the number of personnel to be trained, determining training resources, and determining training costs.

The ETES development study will focus on the development of the data generation techniques. For the most part, training estimation techniques will not be developed during the ETES study. Instead, ETES will incorporate existing estimation procedures and procedures currently being developed under other ARI projects (e.g. HARDMAN, Training Developers Decision Aid).

1.3.3 Human Simulation Models

These models will relate human task performance to overall system performance. Input for the models will be provided by the data contained in the SDT. By relating task performance to system performance, the simulation models will provide the capability for trading off training-related systems elements against other system elements.

ARI currently has an ongoing project (i.e. MOPADS) at ARI, Fort Bliss, to develop advanced human simulation models.

However, these models are rather sophisticated and are more relevant to the types of detailed human performance questions generated during the later phases of the acquisition process. Hence, the ETES will focus on (1) the development of less detailed simulation models which can be meaningfully applied to the types of general questions which are relevant during the early phases of the acquisition process and (2) the incorporation of the MOPADS data requirements into the SDT specifications. The latter effort will insure that the SDT will be able to feed MOPADS simulation models as appropriate during the acquisition process.

1.3.4 User's Guide

The User's Guide will provide a detailed step-by-step handbook describing how the other three ETES tools can and should be used to assess early training requirements.

1.4 ETES STUDY TASKS

The ETES study is broken down into five basic tasks. Figure 1-5 displays an up-to-date description of these tasks. (The terminology of the tasks has been changed slightly to reflect insights developed during the first year of the study.)

1.5 PROGRESS ON STUDY TASKS

Table 1-1 displays the activities accomplished under each task and the sections of the report relating to these activities. More details are provided below.

1.5.1 Task 1: Review of Existing Procedures

This task began with a review of existing DoD Army doctrine and operating procedures related to early training estimation and system description. The purpose of this

Figure 1-5 ETES STUDY TASKS

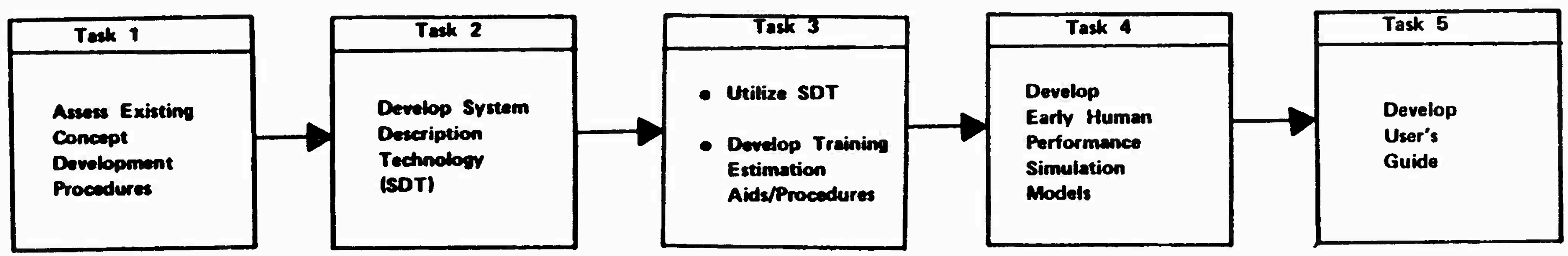


TABLE 1-1 ETES STUDY ACTIVITIES AND RELATED SECTIONS OF THE REPORT

TASK	ACTIVITY	RELATED SECTION OF REPORT
TASK 1: REVIEW CONCEPT DEVELOPMENT PROCEDURES	REVIEW OF EXISTING DOD/ARMY DOCTRINE AND OPERATING PROCEDURES RELATING TO EARLY TRAINING ESTIMATION AND SYSTEM DESCRIPTION	APPENDIX A
	INTERVIEW OF PARTICIPANTS IN ACQUISITION PROCESS	APPENDIX A
	ANALYSIS OF HARDWARE/TRAINING INTERFACE	APPENDIX A, SECTION 2
	REVIEW OF OTHER HUMAN RESOURCE DATA BASES	SECTION 2.4
	REVIEW OF PSYCHOLOGICAL RESEARCH RELATED TO ENGINEERING DESIGN PROCESS	APPENDIX C
	REVIEW OF RESEARCH ON HUMAN COMPUTER INTERACTIONS	APPENDIX D
	REVIEW OF AUTOMATED TOOLS FOR APPLICATION TO SDT	SECTION 3

TABLE 1-1 (continued)

TASK	ACTIVITY	RELATED SECTION OF REPORT
TASK 2 DEVELOP SDT	SPECIFICATION OF SDT DATA ELEMENTS	SECTION 2.0, APPENDIX B
	SELECTION/DESCRIPTION OF VEHICLE FOR SDT	SECTION 3.0
	DESCRIPTION OF SDT USERS	SECTION 4.0
	SPECIFICATION OF SDT PHYSICAL CHARACTERISTICS	SECTION 4.0
	SPECIFICATION OF SDT OPERATION INCLUDING INPUT/OUTPUT MECHANISMS, OPERATIONAL USE, AND LIKELY HUMAN-COMPUTER CONVERSATIONS	SECTION 4.0, APPENDIX B
TASK 3 • UTILIZE SDT • DEVELOP TRAINING ESTIMATION AIDS/ PROCEDURES	TO BE COMPLETED IN REMAINING PORTION OF STUDY	–
TASK 4: DEVELOP EARLY HUMAN PERFORMANCE SIMULATION MODELS	TO BE COMPLETED IN REMAINING PORTION OF STUDY	–
TASK 5: DEVELOP USER'S GUIDE	TO BE COMPLETED IN REMAINING PORTION OF STUDY	–

review was to identify needs and problems associated with current procedures and potential roles for ETES in ameliorating these problems. This review was supplemented by a number of interviews with users in the field. The results of the review and interviews were assessed and integrated into a description of the current acquisition process. The gaps in this process were associated with early training estimation and system description and the likely role of the SDT (see Appendix A).

In addition to the review of existing Army procedures, four different behavioral/information science areas related to the SDT were reviewed: human resource data, automated tools which might serve as a possible vehicle for the SDT, psychological research related to design, and research on human-computer interactions (see Section 2.4, Section 3.0, Appendix C, and Appendix D respectively).

Psychological research related to design was examined to identify the individual cognitive processes relevant to early system design and description. The review of human-computer interaction was conducted to identify guidelines for construction of the SDT human-computer interface.

1.5.2 Task 2: Develop SDT

Utilizing the information developed in the previoius steps, a detailed description of the data elements to be described by the SDT was developed (see Section 2). A particular class of automated tools (data base management systems) was then selected and specific tools within this class were examined in detail (see Section 3). Finally, a detailed description of the SDT users, physical characteritics, input/output mechanisms, and operational characteristics was developed (see Section 4).

1.5.3 Tasks 3, 4 and 5

These three tasks will be performed during the remaining portion of the ETES study.

SECTION 2 - SPECIFICATIONS FOR SDT

This section provides a detailed set of specifications for the functions which must be performed by the SDT, and a general set of requirements for the mechanisms which must be utilized to perform these functions.

The specifications described in this section were developed by examining (a) current Army procedures for system development, requirements analysis (functional analysis), task generation, and training development; (b) non-Army research and work in these four areas; (c) previous attempts to develop system-specific human resource data bases; and (d) previous discussions of SDT requirements in Status Report 1, Status Report 2, and Status Report 3.

The section is divided into five subsections. The first subsection provides an overview of the functional requirements which must be accomplished by the ETES. The second subsection provides a detailed description of the ETES functions and the types of output data associated with each function. The third subsection outlines some general requirements for SDT data input/output mechanisms. The fourth subsection provides a preliminary listing of the sequence in which the SDT functions must be performed. The fifth subsection briefly reviews past efforts which have attempted to identify what should be in system-specific human resource data bases.

2.1 OVERVIEW OF SDT REQUIREMENTS

The SDT is one of four major components of the Early Training Estimation System (see Figure 2-1). The SDT is clearly the most important component of ETES since it provides all the basic system information required by the other ETES components. (This importance is reflected in the amount of resources and time devoted to SDT development.)

The basic goal of the SDT, as outlined on page three of the ETES study RFP, is to:

> . . . provide the Army training and hardware development community with an advanced technology for early generation of improved system descriptions suitable for input into emerging automated training and hardware development aids.

To effectively estimate early training resource requirements, the SDT must describe (a) training programs and their associated resources, (b) the tasks which drive these training programs, (c) the system designs which generate the task requirements, and (d) the functional requirements for which the system designs have been developed. An overview of the application of the SDT to these four system elements and their role in system development is described in Figure 2-2.

In its initial application to a system, the SDT is used to describe the system functional requirements which are generated during functional analysis. These requirements specify the functions which must be performed if the system is to satisfy its designated need. The SDT can be applied

FIGURE 2-1 ETES COMPONENTS

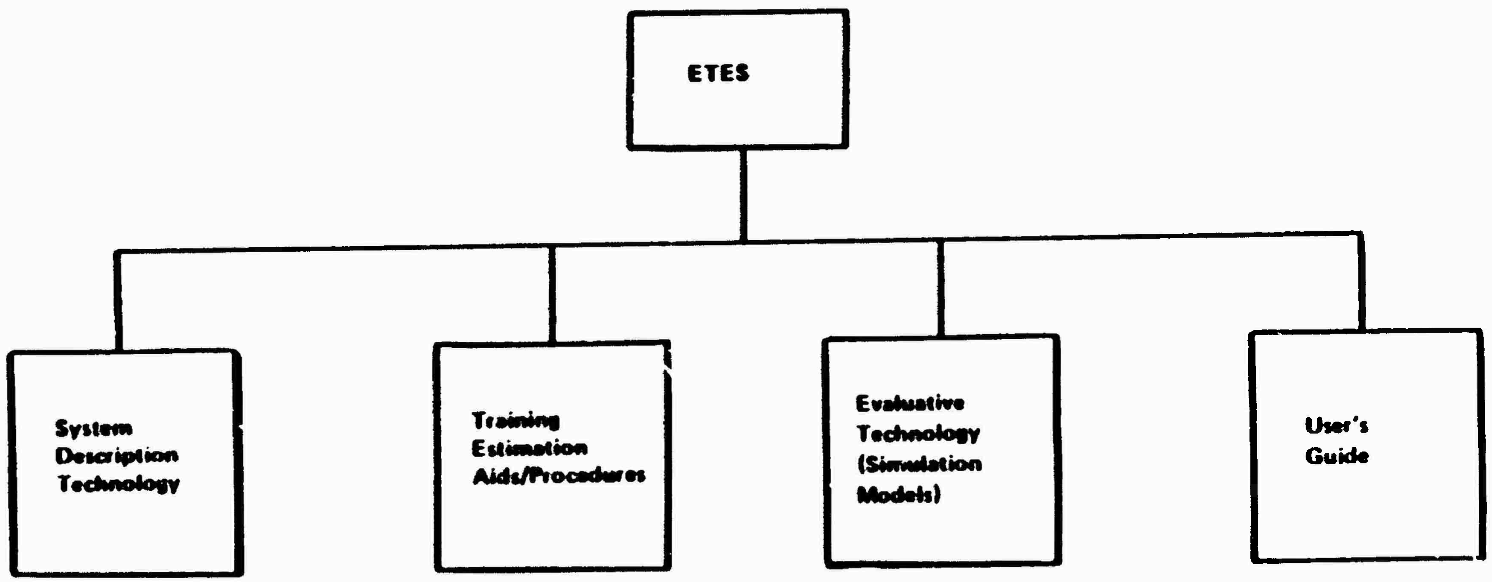
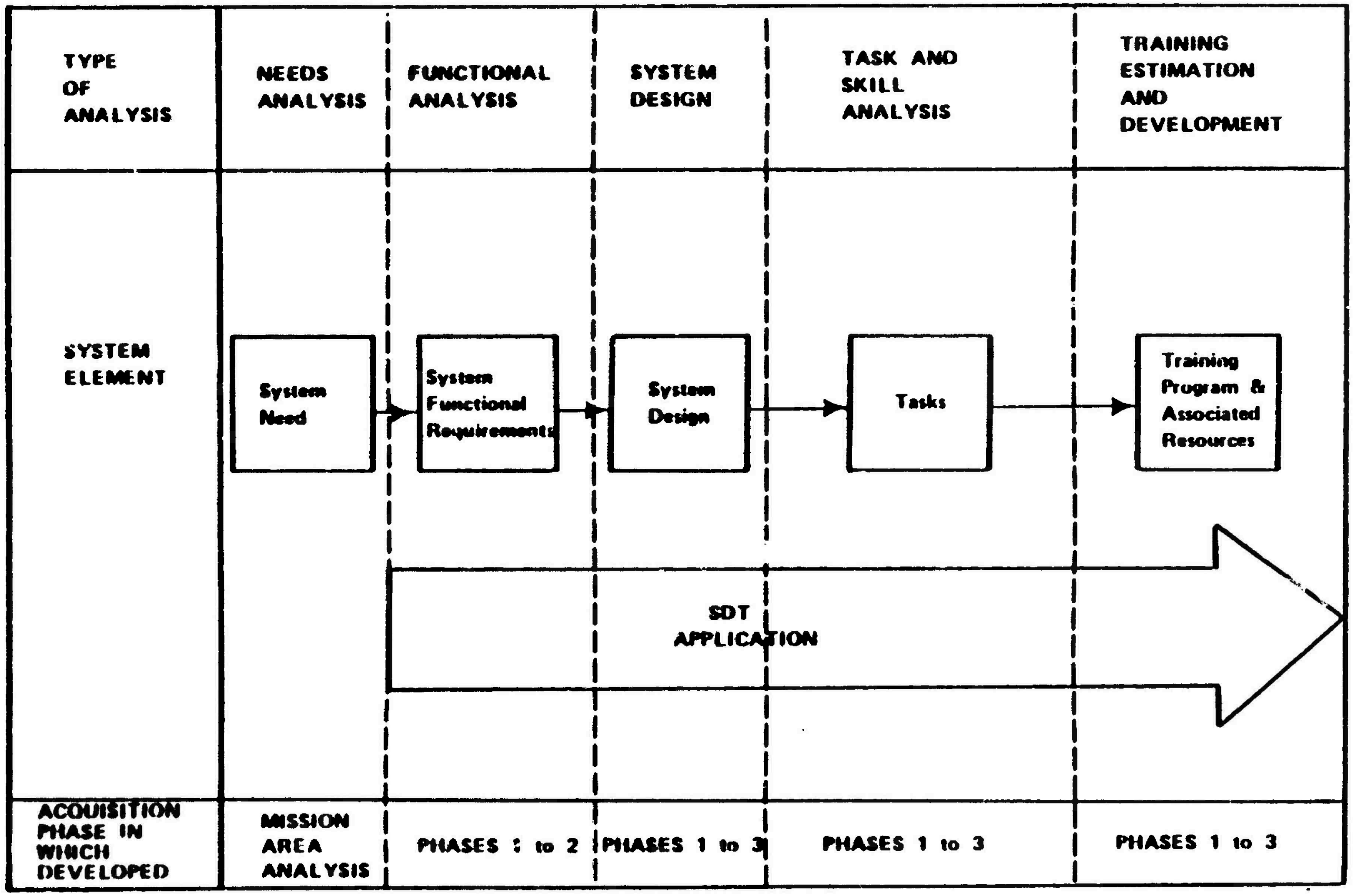


FIGURE 2-2 SDT APPLICATION



in a functional analysis as soon as the need for a particular system has been specified. Formally, this occurs at the approval of the requirements document at Milestone O, the milestone which initiates the Concept Exploration phase of the acquisition process. However, in actuality, the SDT could probably be used to describe functional requirements even prior to Milestone O if the need for a particular system had been identified earlier.

Once the functional requirements for a system have been developed and described via the SDT, system designs can be generated. These designs specify possible mechanisms for performing the desired functions. These mechanisms include equipment, personnel, and software. Once developed, the system design can also be described with the SDT.

Once the mechanisms for achieving the functions have been identified in the design concepts, the human tasks which must be performed to utilize the system designs can be specified. These tasks, which are the key building blocks of training development, must also be carefully documented in the SDT. With the tasks identified and specified in the SDT, training estimation aids and procedures can be used to determine training program elements, estimate training resources, and develop training products. The resulting training program and its associated resources can then be documented in the SDT.

2.1.1 Role of SDT in the Acquisition Process

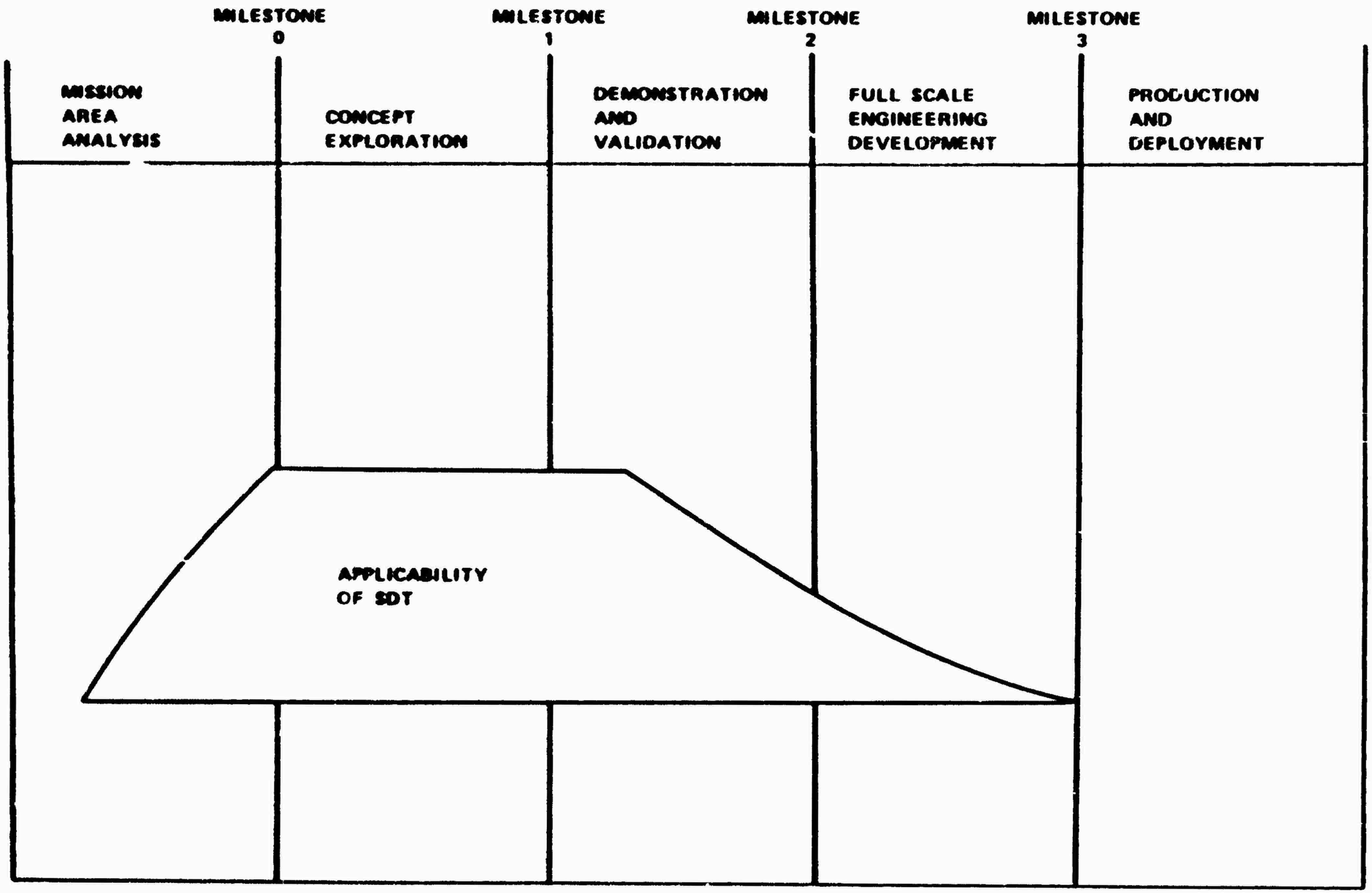
The SDT, like the other components of ETES, is primarily designed for application during the Concept Exploration phase of the acquisition process, which runs from Milestone

0 to Milestone 1 (see Figure 2-3). However, the SDT may also be used during mission area analysis if the need for a particular system has been specified. (Again, it should be noted that this is likely to occur between the time the decision is made to develop a requirements document and its final approval at Milestone 0.) In addition, the SDT may be used during the phases of the acquisition process which follow Concept Exploration. The primary purposes of the SDT applications during the later phases would be to (1) estimate more detailed tasks and training resource requirements, (2) determine the impact of subsequent design changes on task and training requirements via the data base management capabilities of the SDT, and (3) to develop general estimates of task and training requirements for systems which fall behind schedule.

2.1.2 A Basic Data Problem in Early Training Estimation

To provide the necessary information for early training estimation, the SDT must describe functional requirements, system designs, tasks, and training program elements during the earliest phases of the acquisition process. However, there is a basic data problem confronting the analyst who attempts to develop such a description. During the earliest phases of the acquisition process, only functional requirements or very general design concepts are available--information on tasks which are the critical building blocks of training is generally not available. Thus, if the SDT were simply to describe the current state of the system during the earliest phases of the Weapons System Acquisition Process (WSAP), estimation of training resources would not be possible since the data needed for training estimation do not exist during this phase.

FIGURE 2-3 ROLE OF SDT DURING ACQUISITION PROCESS



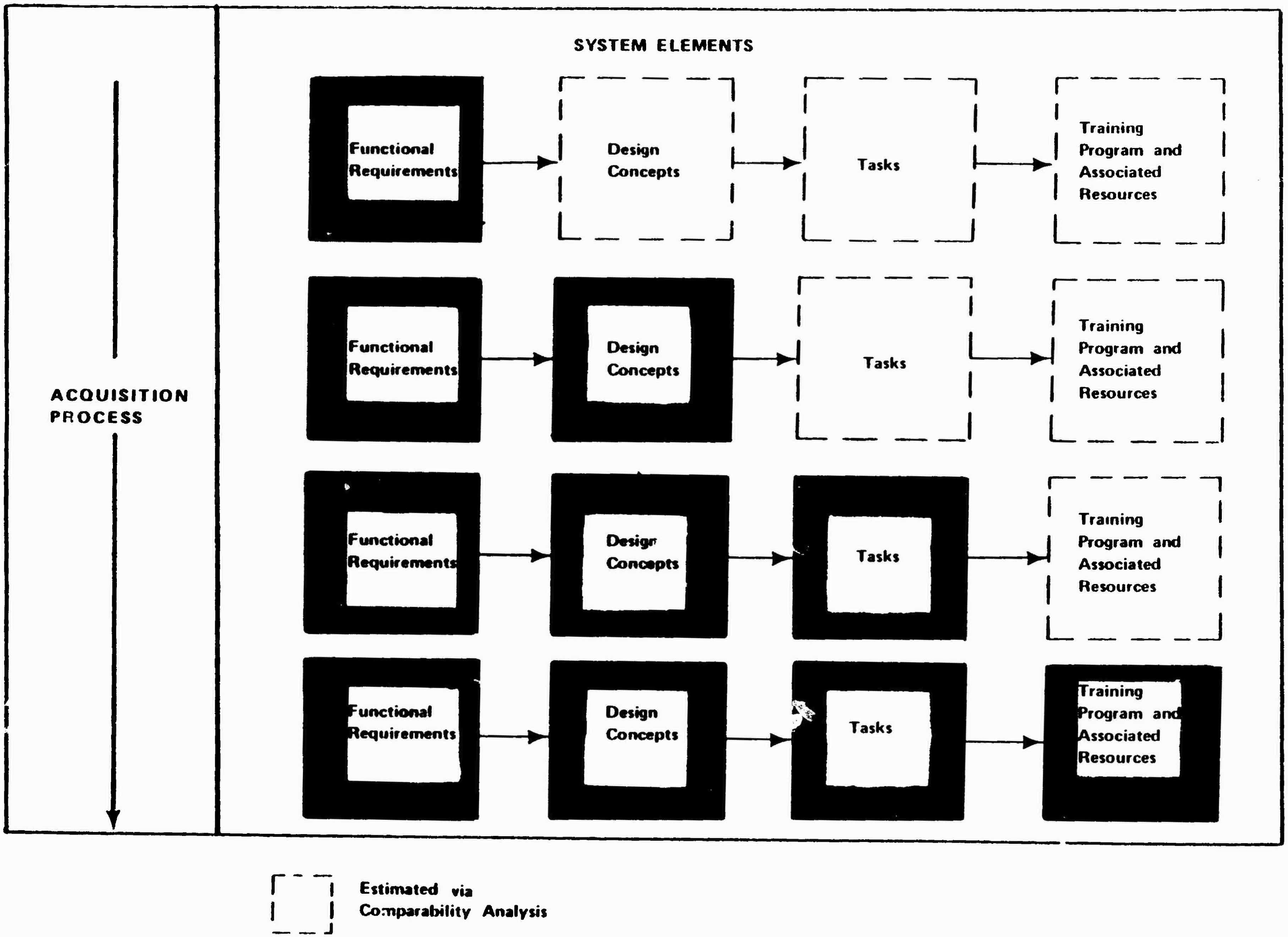
- Solution to Data Problem

To circumvent the data problems described above, the following procedure can be employed. During the earliest phases of the acquisition process when only information on functional requirements is available, a systematic comparability analysis can be conducted to identify the existing subsystems which must closely meet the projected requirements of the new system. Data on these comparable systems can then be obtained and modified to meet the differential requirements of the new subsystem. Thus, by utilizing design and task data from comparable existing systems, systematic estimations of early training requirements can be made when only functional information on the projected system is available (see Figure 2-4). Later, as actual design concepts are developed, the comparability analyses can be used to develop estimates of tasks and training program elements. Still later, when the actual system tasks are available, only the training program elements must be estimated.

- Implications for SDT

The above discussion indicates that the SDT must not only be capable of describing the current state of the system during the earliest phases of the acquisition process, it must also be capable of (1) describing projected system elements and alternative system concepts, (2) relating alternative system concepts to a common framework so that meaningful comparisons can be made, and (3) updating and refining system information as more accurate and more detailed data is developed.

FIGURE 2-4 SYSTEM DEVELOPMENT



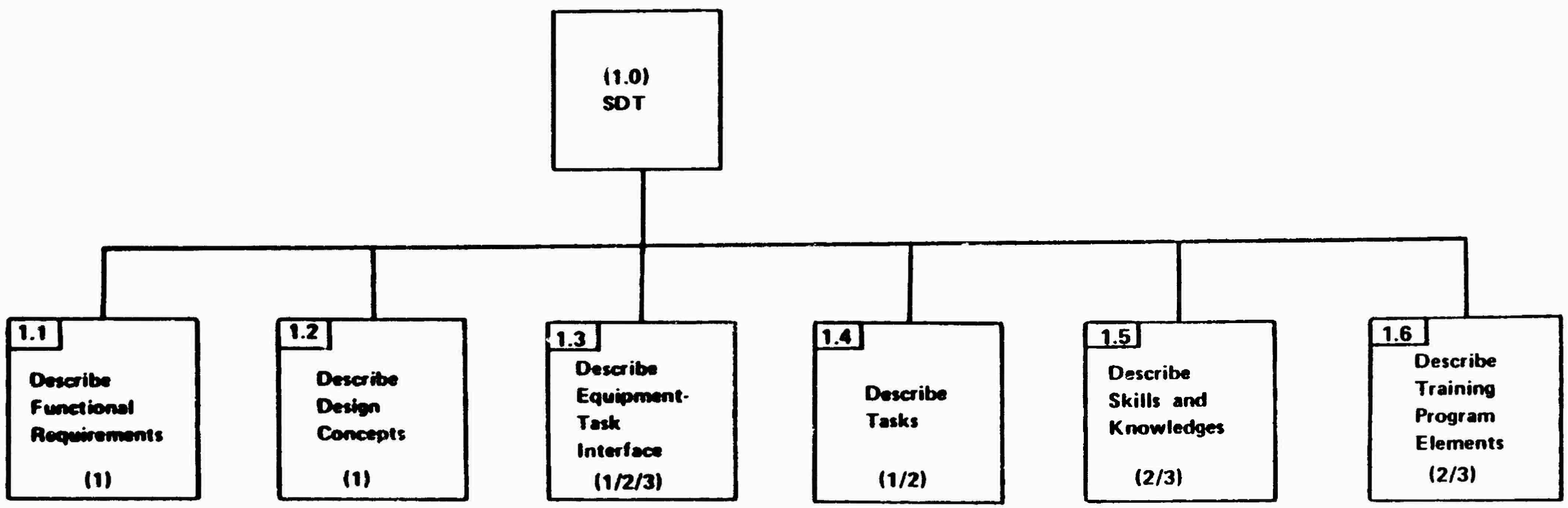
2.2 OVERVIEW OF SDT FUNCTIONS

To develop more detailed specifications for the SDT, a functional analysis was performed on the SDT itself and the results of this analysis were documented in a series of hierarchical functional diagrams. Figure 2-5 provides an overview of the system elements which must be described by the SDT. The elements are comparable to the four system elements described in previous sections (functional requirements, design concept, tasks, and training program elements). However, tasks are broken down into three functions (equipment-task interface, behavioral task elements and features, and skills and knowledges) because more detailed descriptions are required in each of the task areas.

Each function in the diagram is coded to indicate what its developmental priority should be during the construction of the SDT. Functions labeled "1" have the highest priority and should be included in the earliest versions of the SDT. Functions labeled "2" have the next highest priority and functions labeled "3" have the lowest priority.

The major factors used in assigning developmental priorities to the functions were (1) relevance to task generation--functions related to information which was required for task generation were given a higher priority than functions which were not, (2) relevance to the Concept Exploration phase--acquisition process functions which were more likely to be utilized during the Concept Exploration phase were given higher priority, (3) adequacy of present description formats--functions which are not being described adequately via present procedures were given high priority, (4)

FIGURE 2-5 SDT



Development Priority

(1) = Should be included in earliest version of SDT
(2) = Should be included in second version of SDT
(3) = Should be included in later version of SDT

relevance to training developer input needs--information that must be provided to the training developer tended to be given a higher priority than information developed by the training developer.

2.2.1 Functional Requirements

Figure 2-6 lists the SDT functions which must be accomplished during functional requirements analysis. Table 2-1 lists the outputs that must be produced for these functions. Examples of each of these outputs are provided in Appendix B. These examples should only be considered as preliminary estimates of output formats. The exact output format will depend on the mechanisms which are selected to accomplish each function. The examples are only designed to represent the "types of information" which should be provided as output.

The first three system elements related to functional requirements (hierarchical structure, activity flow, and information flow) are concepts which are taken directly from recent discussions of software requirements analysis and are defined as follows:

- Hierarchical Structure - the hierarchical arrangement of functions and their corresponding subfunctions.
- Activity Flow - a representation of the sequence in which system functions are performed during the mission.

FIGURE 2-6 FUNCTIONAL REQUIREMENTS

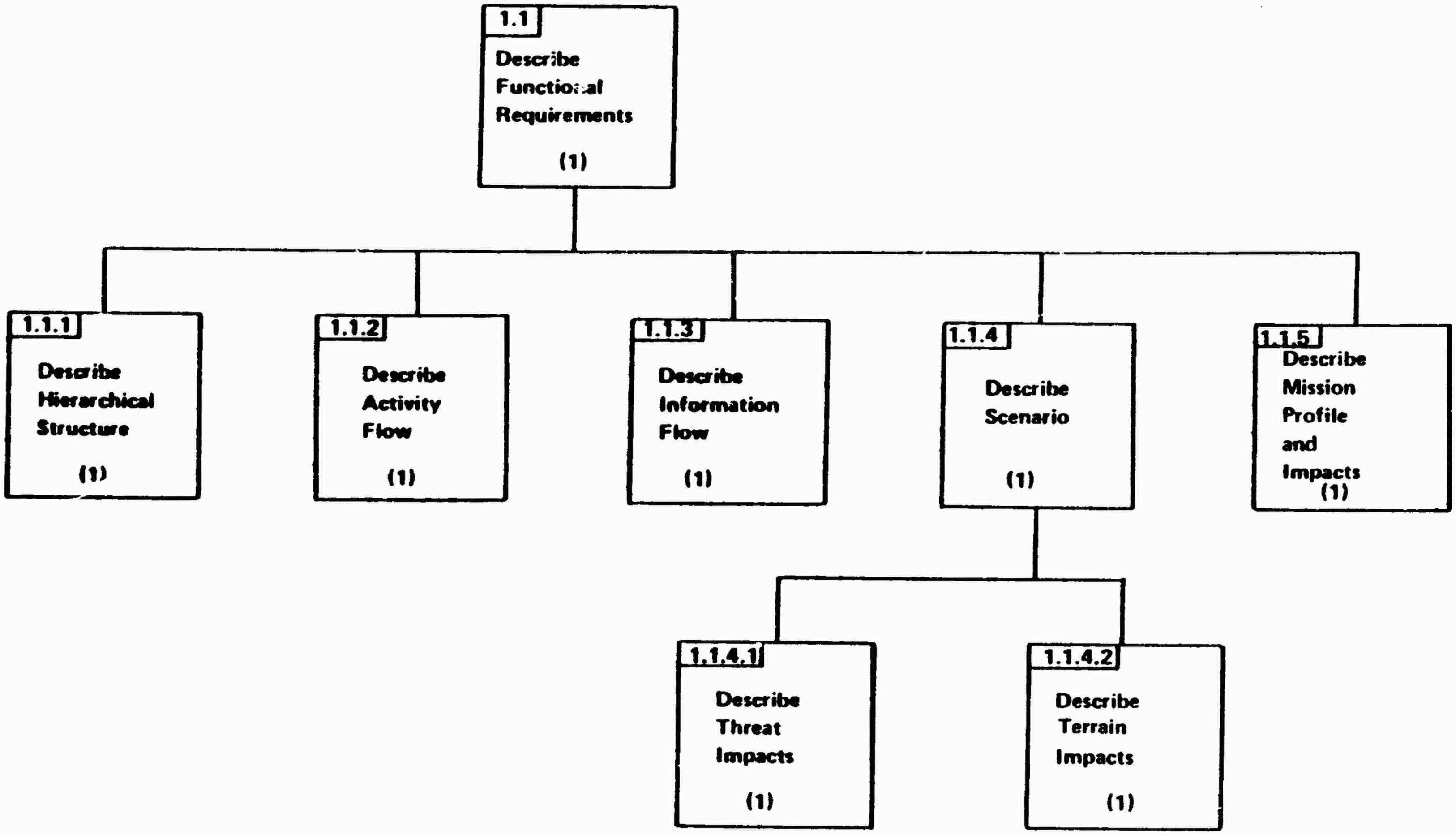


Table 2-1 OUTPUTS RELATED TO FUNCTIONAL REQUIREMENTS

OUTPUT	PRIORITY
• List Hierarchical Structure	(1)
• List Activity Flow*	(1)
• List Information Flow*	(1)
• List Performance Goals by Function	(1)
• List Terrain Impacts on Functions*	(1)
• List Threat Impacts on Function*	(1)
• List Mission Profile Impacts on Functions	(1)

*Tentatively for operational functional requirements only. May also be used with selected maintenance functions.

* Information Flow - the flow of inputs and outputs (in informational terms) between system functions and between system functions and the external environment.

At the highest level, it is likely that the mission-related functional requirements of each system can be broken down into four major functional areas (see Figure 2-7).[1]

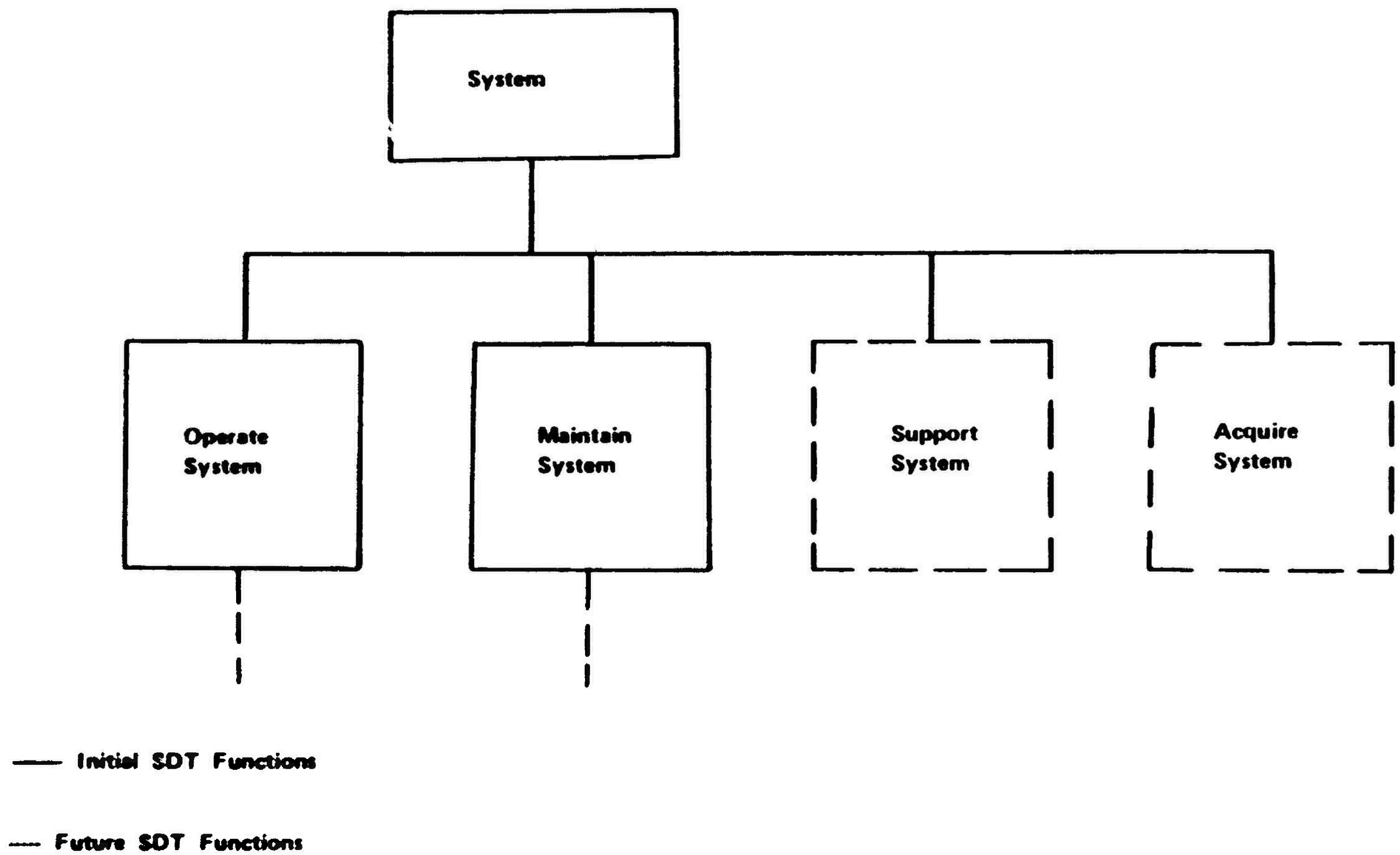
The performance goals for each function are similar to the types of goals described in requirements documents such as the MENS. Whenever possible, these goals must be described in quantifiable terms with minimum and maximum allowable values specified. The performance goals are extremely important since they will be the primary source for the identification of system performance measures during subsequent training analyses. These performance measures will be utilized in the ETES simulation models which will relate task performance to system performance.

The threat and the terrain (e.g., geography, climate) information describe the external environment in which the system must operate. The mission profile describes what the likely goals of the system will be. The SDT will not attempt to provide detailed descriptions of the threat terrain and mission profile as there are likely to be documents specifically devoted to accomplish this task (e.g., terrain and threat information is contained in SCORES

---

1 Two other system functional requirements, "support the system" and "acquire/dispose the system" are not directly mission-related and tentatively are not considered for inclusion in the SDT.

FIGURE 2-7 HIGH LEVEL FUNCTIONAL BREAKDOWN FOR WEAPONS SYSTEMS



documents). The SDT must simply summarize the important variables in each of these three areas, the degree to which the system can be expected to encounter specific environments or act under each mission profile (in quantifiable terms), and the likely impact of these variables on specific system functions (Appendix B).

It should be noted that the current specifications for the SDT functional requirements do not include descriptions of the acquisition goals (e.g., schedule or cost goals). (These goals had been included in earlier versions of the SDT specifications.) The acquisition goals were purposely excluded from the current SDT specifications because it was determined that (1) acquisition goals could be described via current tools and (2) detailed specification of these elements is not necessary for task generation.

2.2.2 Design Concepts

Figure 2-8 lists the design concept elements which must be described by the SDT and Table 2-2 lists the outputs estimated to be required to accomplish these functions. Examples of each of these outputs are provided in Appendix B.

The generic equipment functions list the general type of equipment (e.g., cab, engine, hull) which can be used to satisfy a set of system functions but do not describe the specific piece of equipment used to perform these functions (e.g., M109 cab, GE engine). As the design process progresses, the approved design concept will proceed down the generic equipment hierarchy. In fact, it is possible to identify several different levels of design concept

FIGURE 2-8 DESIGN CONCEPTS

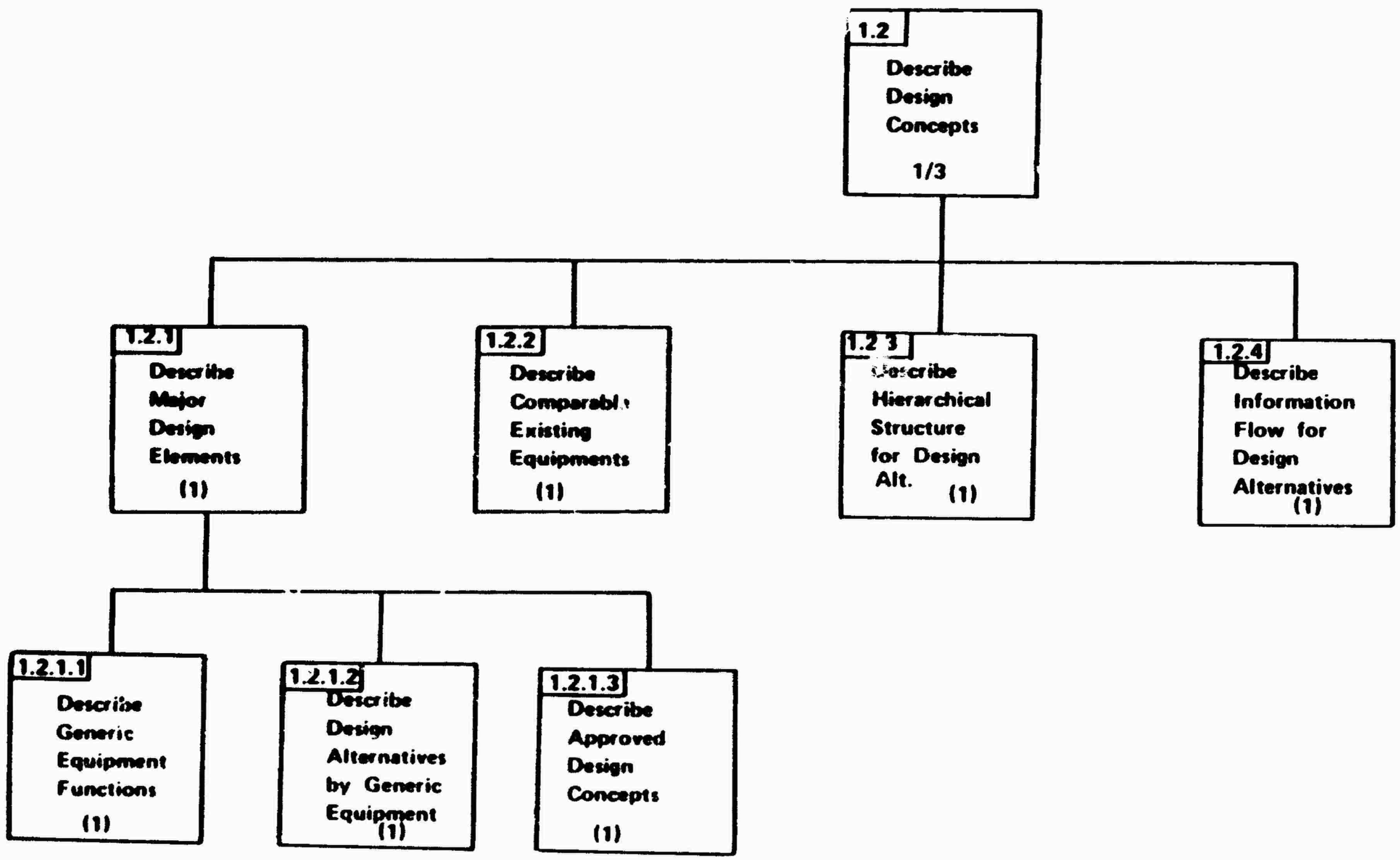


Table 2-2 OUTPUTS RELATED TO DESIGN CONCEPTS

OUTPUT	PRIORITY
• List Hierarchical Structure for Generic Equipment	(1)
• List Hierarchical Structure for Design Alternatives	(1)
• List Alternative Design Concepts by Generic Equipment	(1)
• List Information Flow for Design Alternative	(1)

development. These levels are described in Table 2-3 and the sequence in which the design concepts at each of these levels is developed is listed in Figure 2-9. A large scale system which is closely following the principles outlined in OMB Circular A109 will go through each of the levels listed in Table 2-3.[2]

A smaller system, a system involving a product improvement, or a system not following the principles outlined in A109, can begin the design process at a lower level in the design process.

2.2.3 Equipment-Task Interface

Figure 2-10 lists the equipment-task interface elements which must be described by the SDT and Table 2-4 lists the outputs estimated to be required to accomplish these functions.

Detailed specification of the task performance data (1.3.2) has not been provided because the exact nature of the simulation models which will utilize this performance data has yet to be specified. It was possible to estimate the general types of maintenance performance data that will be required for the maintenance simulation model. These estimations are based upon DRC's current work in maintenance network modeling. It is expected that the maintenance performance simulation model will be based upon these networks.

---

2 OMB Circular A-109 indicates that initial system should be described in purely functional terms.

Table 2-3 LEVELS OF DESIGN CONCEPT DEVELOPMENT

LEVEL I - ALTERNATIVE PLATFORMS

Eg, Self-propelled howitzer vs. multiple launch rocket system.

Separate generic equipment structures are required for each candidate platform with commonalities identified.

LEVEL II - ALTERNATIVE GENERIC SUBSYSTEMS

Eg. System A uses fire control computer to perform function, System B does not (function performed manually).

Different generic equipment structures are required at the subsytem level with commonalities identified.

LEVEL III - ALTERNATIVE SUBSYSTEMS

Eg. System A uses GE engine, System B uses Chrysler engine.

Same generic equipment structures at subsystem level, but different design alternatives associated with these generic subsystems

LEVEL IIIA - ALTERNATIVE GENERIC COMPONENTS

Eg. System A uses GE engine with new built-in-test equipment, System B does not.

Different generic equipment structures at the component level.

LEVEL IIIB - ALTERNATIVE COMPONENTS

Eg. System A uses GE engine with existing carburator, System B uses GE engine with new carburator.

Same generic equipment structures at the component level, but different design alternatives are associated with generic component.

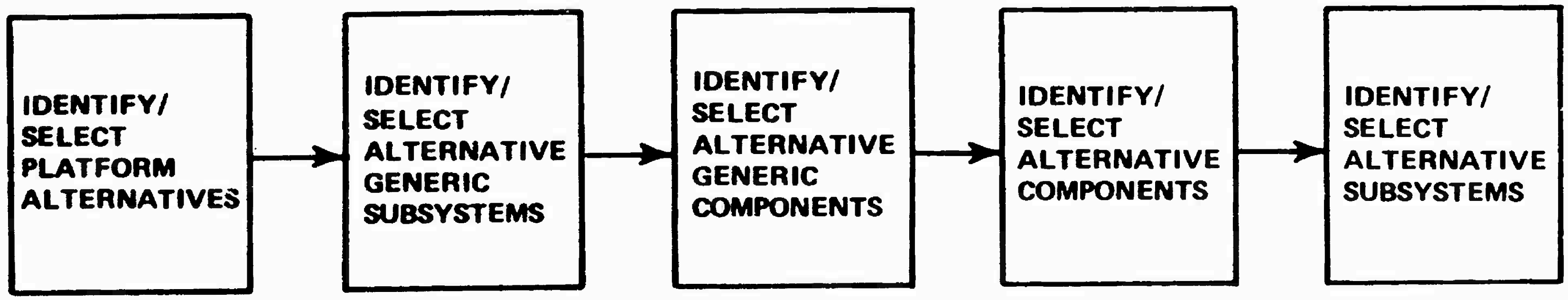


FIGURE 2-9 SEQUENCE FOR IDENTIFYING DESIGN ALTERNATIVES

881-010-1

FIGURE 2-10 EQUIPMENT TASK INTERFACE

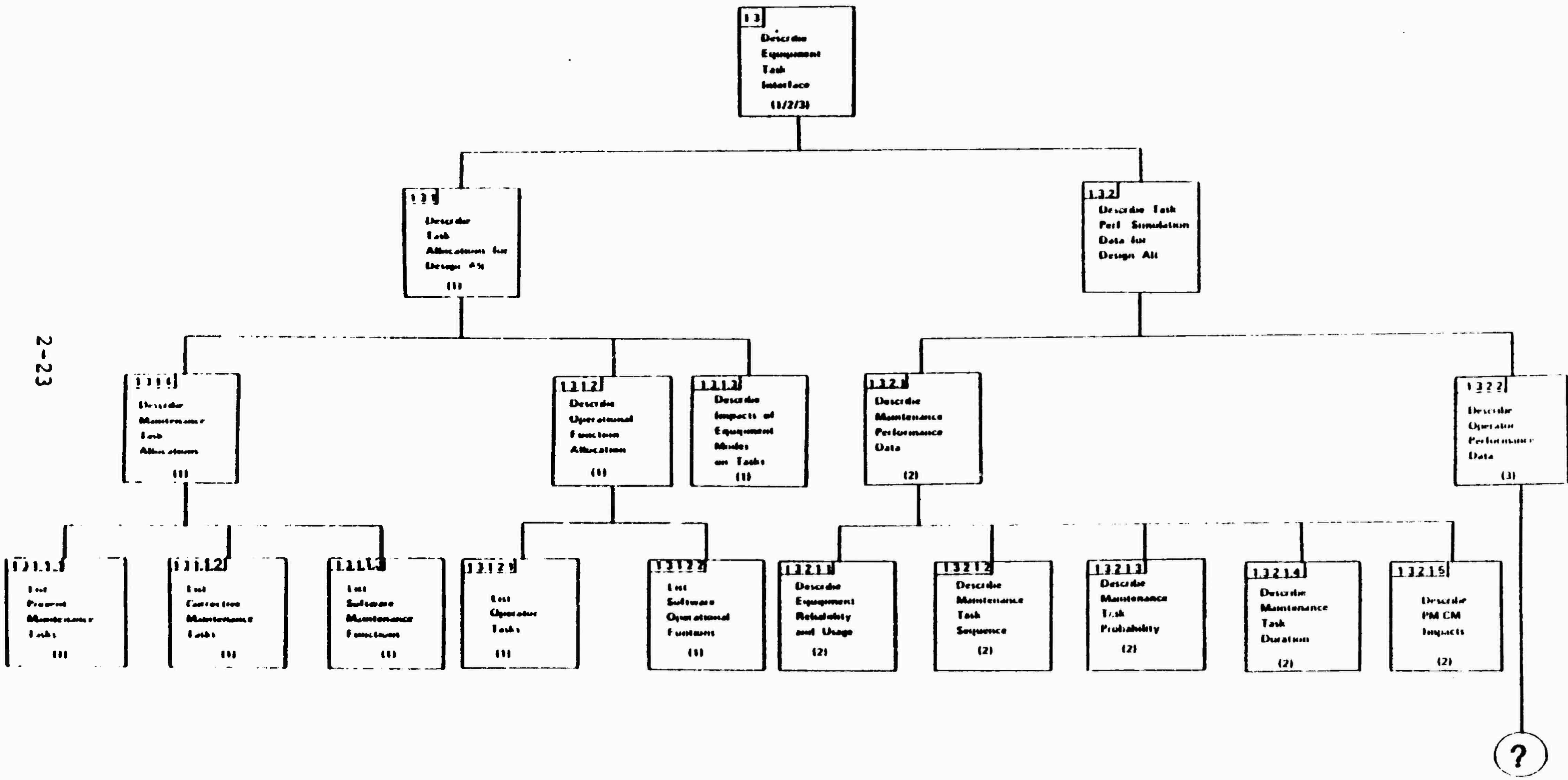


Table 2-4 OUTPUTS RELATED TO EQUIPMENT-TASK INTERFACE

OUTPUTS	PRIORITY
• List corrective maintenance tasks by equipment and function for each design ALT	(1)
• List preventive maintenance tasks by equipment and functon for each design ALT	(1)
• List operator tasks by equipment and function for each design ALT	(1)
• List impact of equipment modes on tasks	(1)
• List reliability data and usage data by equipment for each design ALT	(2)
• * List maintenance task sequence, probability, and duration by equpment for each design ALT.	(2)
• * List impacts of preventative maintenance tasks on corrective maintenance tasks by equipment for each design ALT	(2)
• * List operational performance data by equipment for each design ALT.	(3)

*It is possible to group this information under the behavioral task analysis area.

2.2.4 Behavioral Task Elements and Features

Figure 2-11 lists the task information elements which must be described by the SDT and Table 2-5 lists the outputs estimated to be required to accomplish these functions.

It is possible, and in fact likely, that the task activity flow and task information flow data will be required as input into the task performance simulation models. If this information is required as input into these performance simulator models, it can be included in the SDT function related to the task performance (function 1.3.2) and need not be repeated under 1.4.

The task characteristic data will contain quantitative information on the variables which will be utilized in algorithms designed to (a) determine the tasks to be trained, (b) assign tasks to training settings, (c) assign tasks (or their associated learning objectives) to methods and media. These algorithms will be developed during the construction of the ETES estimation aids and procedures. The exact nature of the task characteristics cannot be specified until further work has been done on the training estimation aids/procedures.

The task information included in SDT function 1.4.1 (task components) and 1.4.3 (task features) is designed to contain all of the relevant task elements contained in the behavioral task description worksheets which are currently applied in the Army, such as LSAR Data Sheet D specified in MIL-STD-13888-1, the Job and Task Analysis Worksheet in the Army's Job and Task Analysis Handbook (TRADOC PAM 351-4),

FIGURE 2-11 TASKS

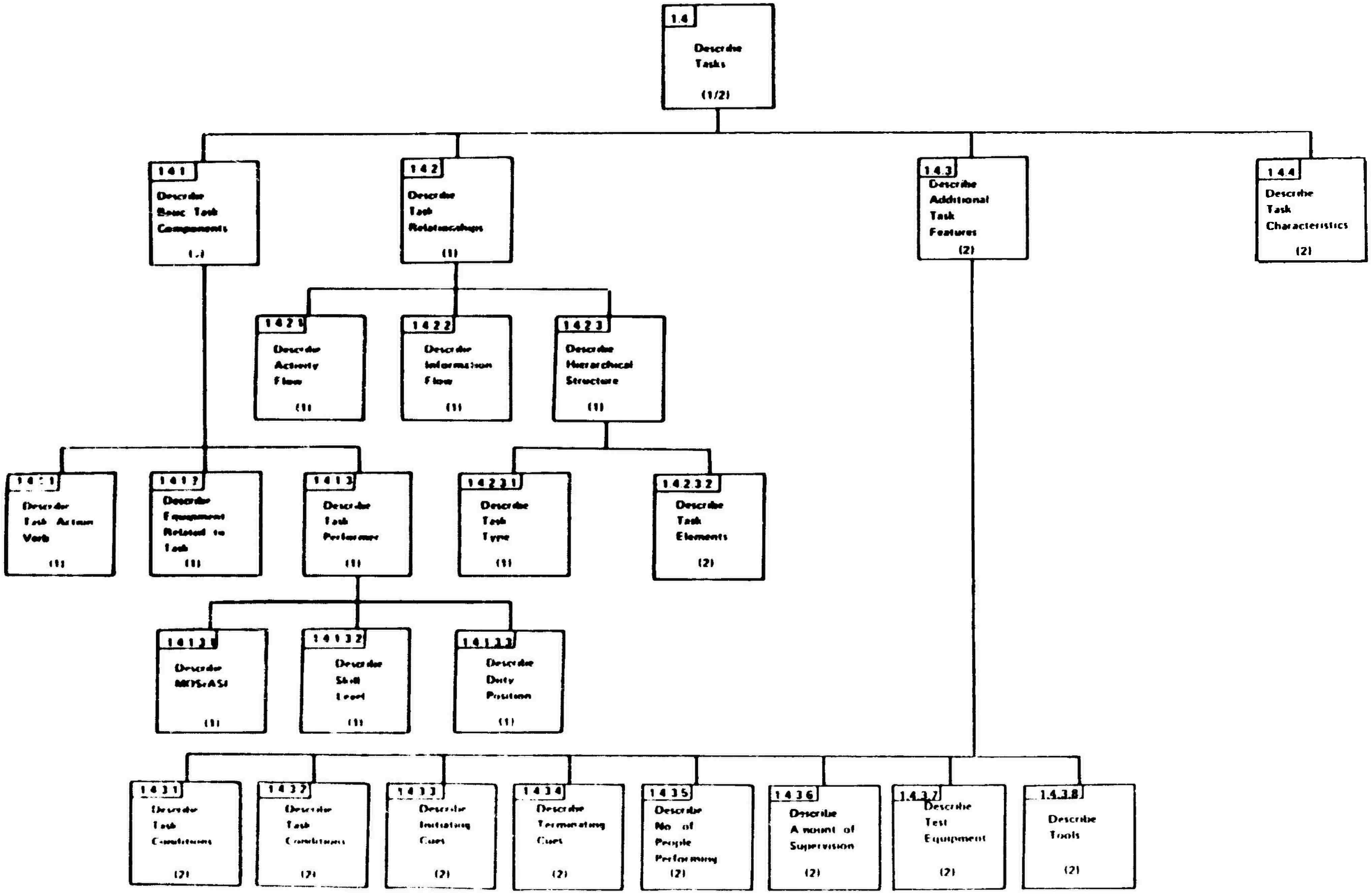


Table 2-5. OUTPUTS RELATED TO TASK INFORMATION

OUTPUTS	PRIORITY
• List tasks by MOS/ASI, by skill level, or duty position	(1)
• For each task, list conditions; standards; initiating and terminating cues, number of people performing; amount of supervision; test equipment; tools, task type, task elements; task characteristic ratings and training setting assignments	(2)
• List task activity flow	(2)
• List tasks by task type	(1)

and the DoD guidelines for contractor supplied task analyses (MIL-STD-1379A and DI-H-2025).

2.2.5 Skills and Knowledges

Figure 2-12 lists the skill and knowledge information elements which must be described by the SDT and Table 2-6 lists the output estimated to be required to accomplish these functions.

The skills and knowledges characteristic information will be used to categorize the skills and knowledges and/or quantify their characteristics. These characteristics can be used in the algorithms which assign methods and media. Again, as with the task characteristics, the exact nature of the skills and knowledge characteristics cannot be specified until more work on the development of these algorithms has been accomplished.

2.2.6 Training Program Elements

Figure 2-13 lists the training program elements which must be described by the SDT and Table 2-7 lists the outputs estimated to be required to accomplish these functions.

2.3 GENERAL GUIDELINES FOR INPUT/OUTPUT MECHANISMS

This section describes some general guidelines for the development of the SDT input/output mechanisms.

FIGURE 2-12 SKILLS AND KNOWLEDGE

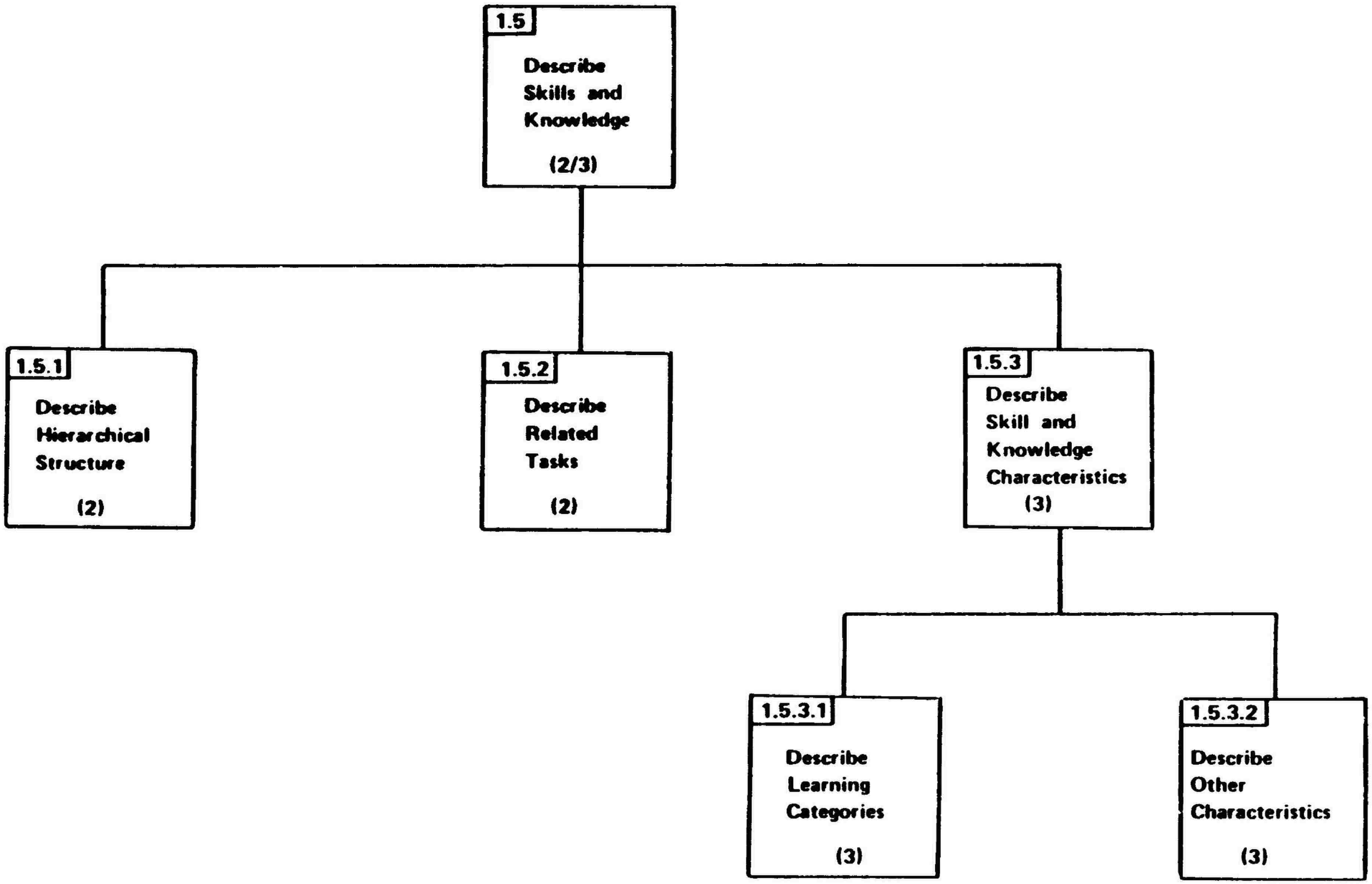


Table 2-6 OUTPUTS RELATED TO SKILLS AND KNOWLEDGE (S+K)

OUTPUTS	PRIORITY
• List S+K hierarchical structure for each design alternative	(3)
• List S+K by tasks, MOS/ASI, skill level, and duty position for each design ALT.	(3)

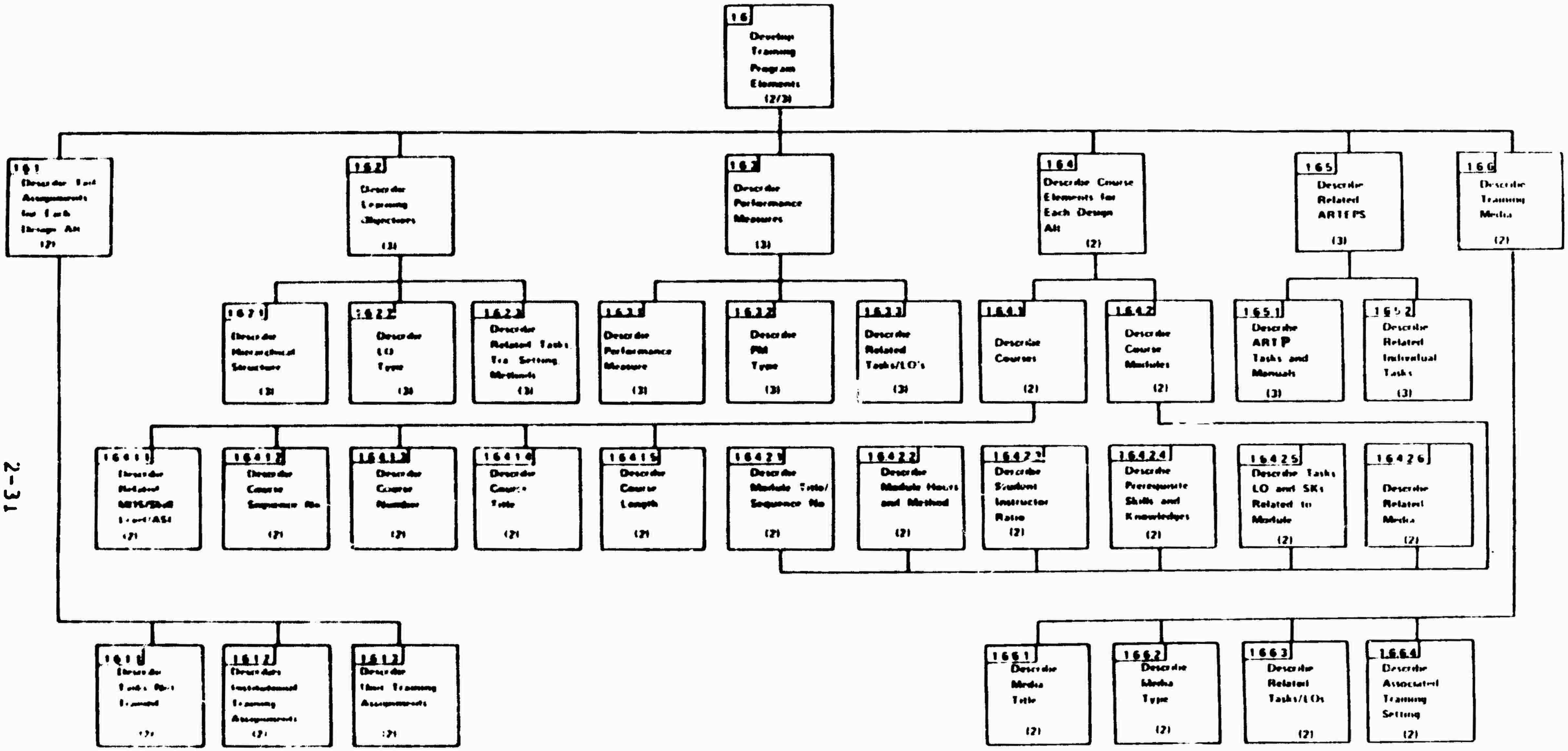


FIGURE 2-13 TRAINING PROGRAM ELEMENTS

Table 2-7 OUTPUTS RELATED TO TRAINING PROGRAM ELEMENTS

OUTPUTS	PRIORITY
• For each learning objective, list learning objective, its place in learning hierarchy, related tasks, skill and knowledges, and learning objective type	(2)
• List performance measures, related tasks and LO's and performance measure type	(2)
• List courses and their course sequence no., course no. course title, and course length by MOS for each design alternative	(2)
• For each course module within a course, list module title, hours, method, student - instructor ratios, related tasks, skills and knowledges, LO's, PMS, and media	(2)
• List ARTEP tasks and manuals and their related individual tasks	(3)
• List media, media type, related tasks, learning objectives and training setting	(2)

At a general level the SDT should ultimately meet all of the following guidelines:[3]

1. The SDT must minimize input data requirements. The SDT must not require users to repeatedly input the same information and must be able to utilize information in existing documents and data banks whenever it is possible to do so.

2. The SDT must interface with existing acquisiton procedures and documentation requirements. Whenever possible, the SDT must utilize input data required by other Army acquisition procedures and/or provide output which can be utilized in these procedures with as little modification as possible.

3. The SDT must be "user-friendly." The SDT must not require extensive training to use or apply, and must be usable by a wide range of users. The input mechanism should be as "transparent" as possible so that user responses can be elicited by the SDT and user's are not required to commit large amounts of SDT-related information to memory. In line with this, the SDT must not require the user to learn complicated computer languages.

---

3 It may not be possible to stay within all of the guidelines with the initial versions of the SDT. However, the final version of the SDT should meet all of the guidelines listed in this section.

4. The SDT must be capable of supporting multiple users. SDT must be capable of being accessed by multiple users in several different locations. The SDT data base must be "secure" so that users can only modify that portion of the data base for which they are directly responsible.

5. The SDT must be capable of maintaining data bases for several design alternatives. The SDT must be capable of describing design, task, and training program data for several alternative concepts and be capable of relating these alternative data elements to a common framework so that meaningful comparisons can be developed.

6. The SDT must deal with frequent design changes. The SDT must have the capability of quickly providing users with information on the design, task, and training program elements associated with a particular design change.

7. The SDT must be able to deal with the evolutionary and expanding features of developing systems. The SDT must be capable of incorporating increasingly detailed system information with minimum user input requirements.

8. The SDT must be flexible enough to handle a variety of different types of input. Thus, it must have the capability of handling both batch and interactive inputs.

9. The SDT must also be flexible enough to provide a variety of different types of outputs including lists, block diagrams, and formatted outputs appropriate for use in other ETES procedures and other acquisition processes.

10. To keep implementation costs to a minimum, the SDT should, to the maximum extent possible, be compatible with equipment (e.g., computer terminals) which is currently being used by the Army organizations who will employ the SDT.

11. To facilitate both software development and system flexibility, the SDT must be capable of maintaining a central data base structure which is "independent" of the specific user applications programs which access it.

2.4 SEQUENCE OF SDT APPLICATIONS THROUGHOUT THE ACQUISITION PROCESS

This section outlines, at a general level, how the SDT might be applied during the system development process. This outline describes the general sequence and types of SDT applications which are appropriate for different stages of system development. This section does not attempt to provide detailed description of the SDT utilization. Thus, it does not describe the specific organizations which will utilize the SDT or the documents and processes which will feed into or utilize the SDT. Identification of the exact users of the SDT must be made by the Army--however, likely potential users, at a general level, are listed in the description of the final SDT in Section 4.

The differing applications of the SDT throughout the acquisition process are grouped into a series of discrete periods (see Figure 2-14). Descriptions of these periods are provided in the subsections which follow.[4]

2.4.1 Period 1: Initial Functional Requirement Analysis

This period encompasses the time between the decision to meet an identified need with a hardware system (rather than with an organizational or operational change or more advanced technology development) and the time when initial functional requirements are specified. Ideally, the end item of this period is a functional requirements description that will allow system designers to develop design concepts down to the subsystem level. The functional requirements developed during this period will provide the foundation for the remaining phases of the acquisition process. Thus, they must be developed very carefully. The SDT can be used to describe the functional requirements which are developed during this phase including system functions, threat, environmental impacts on functions, mission profile and desired performance goals. No estimates of training resources are made during this period since such estimates cannot be made until the functional requirements have been specified.

- <u>Major SDT Applications</u> - Description of functional requirements and provision of input data into

---

4 These period descriptions are geared for major system acquisitions. A slightly different series of SDT applications would be required for minor systems or for product improvement changes.

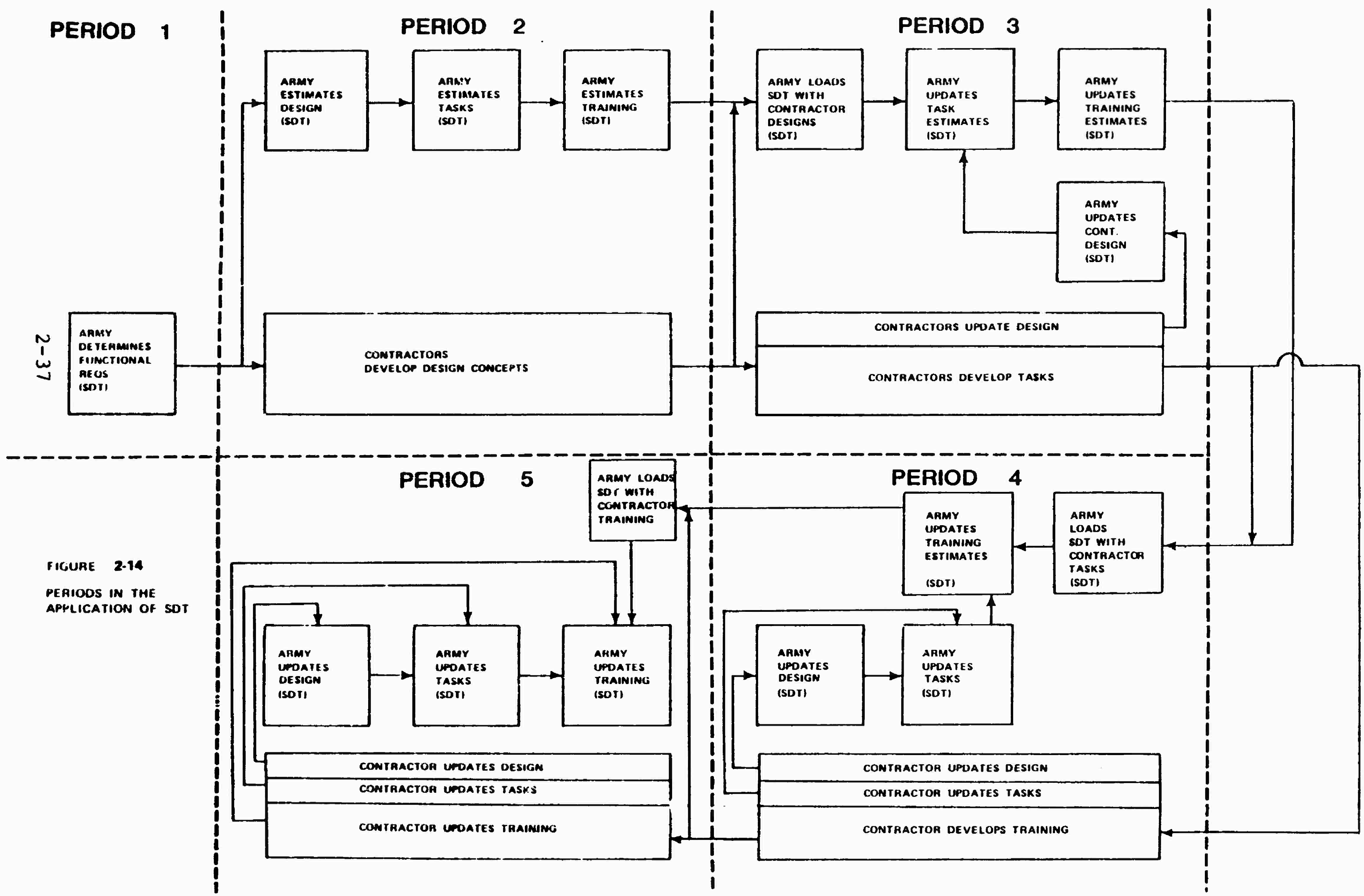


FIGURE 2-14

PERIODS IN THE APPLICATION OF SDT

requirements documents and other initial acquisition documents requiring information on functional requirements.

2.4.2 Period 2: Initial Training Estimation--Contractor Design Alternatives Not Specified

This period covers the time between the specification of the initial functional requirements and the time when contractors have completed their initial design concepts. Thus, during this period information on contractor design concepts is not available. During this period, the initial functional requirements can be examined and the comparable existing systems which come closest to meeting these functional requirements can be identified. Design, task, and training information on these systems can then be collected and modified to reflect the projected system requirements. The outputs of these design, task, and training program analyses can be described in the SDT.[5] This information can then be used to estimate training resource requirements. This initial estimate can be then compared with the predecessor system to indicate how the projected system fits within the footprint of its predecessor. If a specific platform (e.g., howitzer, rocket launcher) has not been selected, initial training estimates should also be developed for each platform type and compared with one another during this period.

---

5 It is possible to input contractor's data into this estimation process in a step-by-step manner rather than wait until the contractor studies are complete. However, such an approach runs counter to the current practice.

It should be noted that the current practice is not to develop any systematic estimates of training resource requirements during this period but to wait until the contractors have completed their initial studies and then have experts make undocumented estimations of the general training resource requirements. This approach overlooks the fact that (1) critical design questions are being asked during this period and these questions require training input data and (2) the early training planning process requires a solid foundation on which to work at this point.

System elements should not be described at a low level of detail at this point in the acquisition. This means that (a) system design data (1.2) should only be specified down to the equipment level and only described in generic terms (e.g., fire control computer), (b) task data (1.4) should only be specified down to the task level and only specified for those systems related to new design changes (versus the equipment on the comparable existing system from which it is derived), (c) training for subsystems not related to new technologies are not changed unless deficiencies in the current training program are identified, (d) only general skill and knowledges (1.4) must be specified, (e) learning objectives (1.6.2) and performance (1.6.3) and ARTEP information (1.6.5) need not be specified, and (f) only general training media (1.6.6) requirements must be identified since the major emphasis is on identifying expensive media (e.g., training devices).

- Major SDT Applications - Documentation and development of initial design, task, skill, and training estimates; provision of input into training planning and acquisition documents;

provision of input into system tradeoff analyses; provision of input into evaluation of general training alternatives; and provision of input into contractor studies for concept investigation.

2.4.3 Period 3: Training Estimation for Identified Design Concepts

This period covers the time between the completion of the contractors' initial design concept studies and the development of initial task data for finals which have been built to represent these design concepts.

The application of the SDT during this period is similar to the application of the SDT during Period 2 with three major exceptions. First, design concepts no longer have to be estimated but can be taken directly from the contractor reports. Second, and most important, design, task, and skill data can be taken to a lower level of detail and thus more detailed estimates of training program elements and training resources can be developed. The level of detail to which one can go may vary from subsystem to subsystem. In general, it is possible to go to lower levels of detail with systems with smaller technological change than with systems associated with larger technology changes. Third, with the formal identification of design concepts, greater emphasis can be given to the examination of training alternatives (that is, of alternative ways of training for the same tasks). This examination of training alternatives will take place during Cost and Training Effectiveness Analyses (CTEA).

- <u>Major SDT Application</u> - Documentation of alternative designs; documentation and development of task, skill, and training estimates; provision of input into training planning, training analysis and acquisition documents; provision of input into system tradeoff analyses and evaluation of alternative designs; provision of input into evaluation of training alternatives; provision of input data into and/or the receipt of output data from ongoing contractor concept development studies; and evaluation of impacts of design changes within each design alternative on tasks, skills, and training.

2.4.4 Period 4: Training Estimation for Identified Tasks

This period encompasses the time between the initial development of tasks by the contractors for the alternative design concepts and the development of training program elements.

The application of the SDT during this period is similar to the preceeding period with three major exceptions. First, task data no longer must be estimated but can be directly obtained from contractor input data. Second, design, task, and skill data can be taken to a lower level of detail permitting more detailed estimates of training program elements and resources. Third, more specific training alternatives can be examined.

- <u>Major SDT Applications</u> - Documentation of alternative designs and their associated tasks; documentation and development of training program data; provision of input into the development of criteria for evaluating contractor supplied task data; provision of input into training planning and acquisition documents; provision of input into system tradeoff analyses and evaluation of alternative designs; provision of data for the evaluation of detailed training alternatives; provision of data to, and/or the receipt of output data from ongoing contractor concept development studies; and evaluation of the impacts of design changes within each design alternative on tasks, skills, and training.

2.4.5 Period 5: Training Development for Selected System

This period encompasses the time between the initial development of training data for the selected system and the completion of the development of the training program for that system.

The period differs from the previous period in three major ways. First, as the period progresses, training program data need no longer be estimated--actual training program data can be utilized. Second, task, skill, and training data must be carried down to the lowest level needed for training development. The SDT data elements need not be described at these lowest levels; however, all general SDT data elements should be completed. Third, unlike Period 2-4 where the major focus of the SDT was on the provision of information for training estimation, during Period 5 the major focus of the SDT is on data base management--that is, keeping track of minor design or task changes and their impacts on other system elements.

- Major SDT Applications - Documentation of system design, tasks, skills, and training program elements; provision of criteria for evaluating input into the development of contractor training program elements; evaluation of the impacts of changes of one system element on other system elements; provision of input into training planning and acquisition documents; input into the evaluation of system tradeoff analyses; and provision of input into the evaluation of detailed training alternatives.

2.5 PAST EFFORTS IN DEVELOPING SYSTEM-SPECIFIC DATA BASES

One of the major sources of information which was utilized in constructing the SDT specifications described in the previous subsections were past efforts in developing system-specific human resource data bases. Table 2-8 lists the major past efforts at developing human resource data bases. These efforts are reviewed in the subsections which follow.

2.5.1 Logistics Support Analysis Record

One major effort which is closely related to the objectives and goals of the SDT is the Logistics Support Analysis Record (LSAR). The role of the LSAR in the acquisition process is discussed in Appendix A. MIL-STD-1388 states that the goal of the LSAR is to be the "single source of validated, integrated design-related logistic data pertinent to the acquisition program."

Table 2-9 lists the system elements that are described by the LSAR and the major weaknesses of the current LSAR in

Table 2-8

PAST EFFORTS AT HUMAN RESOURCE DATA BASE DEVELOPMENT*

(1) Logistics Support Analysis Record (LSAR)

(2) Unified Data Base of Air Force Human Resource Lab

(3) Consolidated Data Base (CDB) of Navy/Army HARDMAN Projects

(4) Structured Approach to Training (SAT) Program for the B1-Bomber

(5) Navy Enlisted Professional Information Support System (NEPDISS)

*Efforts are listed in terms of their decreasing relevance to the ETES SDT.

Table 2-9

OVERVIEW OF LSAR AND ITS MAJOR WEAKNESSES

System Elements Described by LSAR

- Equipment (work breakdown structure, work unit code, nomenclature, reliability, maintainability, failure symptoms, failure effect and criticality, maintenance concept)
- Tasks (task code, frequency, elapsed time, skill specialty, man hours, requirements for training equipment, support equipment, tools, task elements, aggregate maintenance man-hour requirements)
- Support and Test Equipment (physical characteristics, associated equipment, associated tasks, associated training, special skill requirements)
- Facilities (associated equipment and tasks, general requirements, lead times, type of construction, utilities, facility unit cost)
- Skills (associated task and equipments, specialty codes, aptitude, rank/rate, special physical and mental requirements, educational requirements, additional training requirements)
- Supply Support (part no. and nomenclature, physical description, associated equipment, allowance quantity, distribution)

Major Weaknesses of LSAR

- Does not describe system functional requirements
- Does not provide adequate description of operator tasks
- Does not describe task characteristics or performance information
- Does not describe collective tasks
- Does not adequately describe skills
- Does not adequately describe training program elements
- Does not provide mechanism for describing estimated or projected elements
- Is not applied in early phases
- Does not have data base management capability
- Cannot generate tasks or other input data

*Many of these limitations are apparently being dealt within the present LSAR improvement programs.

respect to the goals and objectives of the SDT. As Table 2-9 indicates, the LSAR has several weaknesses which limit its use as a comprehensive system description technology for human resource assessment.

First, there are several important system elements (e.g., system functional requirements, collective tasks) which the LSAR does not describe. Failure to describe the system functional requirements is particularly distressing, since these functional requirements provide the foundation on which all other system elements depend. Lack of a systematic description of functional requirements makes it extremely difficult for training developers and others who are tasked with relating their particular system elements to overall mission performance and its associated functions. For instance, it makes it extremely difficult to relate human tasks to mission performance. Given its lack of a capability for describing system functional requirements or projected system elements, it is not surprising that the LSAR is currently not applied during the concept exploration phase of the acquisition process and seldom, if ever, applied during the validation and demonstration phase. Hence, its value as a data base to support early human resource assessment is very minimal indeed.

Second, there are a number of other systems elements which are described by the LSAR but are not described adequately or in enough detail (e.g., operator tasks, task characteristics, training program elements skills). The emphasis of the LSAR on maintenance assessment and maintenance tasks is quite obvious. This emphasis makes it extremely difficult to develop or maintain adequate descriptions of operator tasks. For all types of tasks, the

LSAR does not fully describe the task characteristics and performance information that is needed by training and/or human factors specialists to adequately assess their components of the system. The training portion of the LSAR places an emphasis on training equipment and devices and ignores other important aspects of the training program (e.g., learning objectives).

Third, at a more conceptual level, the LSAR does not provide an adequate capability for describing estimated or projected system elements. Such estimates are necessary during the early phases of the acquisition process.

Fourth, the LSAR was not conceived as an automated data base management system for system description -- that is, as an automatic system for describing, updating, and expanding system concepts and communicating this information to system users. It should be noted that the Army, through the DARCOM Materiel Readiness Support Activity, has been a leader in "automating the LSAR". However, this automation apparently refers only to the use of computerized algorithms for aggregating certain LSAR elements or for presenting printed outputs of reports. It is not designed to be an interactive system. More important, the automated LSAR does not provide for the automated description of system concepts, updates, changes and expansions through a comprehensive data base management system. This is due to the fact that the LSAR does not have a systematic internal structure linking the various system elements to one another.

2.5.2 Air Force Human Resources Lab Unified Data Base

The Air Force Human Resource Lab (AFHRL) has initiated a program to develop a Unified Data Base (UDB). The goals of the UDB are very similar to the SDT (see Thomas, Newhouse and Hankins, 1980; Thomas and Hankins, 1980). Ultimately, the UDB is designed to provide "a centrally located data base of human resource-related information for utilization in the weapon system acquisition process to influence hardware concepts and design". The UDB is to be supported by a Data Generating Technology Data Base (DGTB) which is intended "to generate generic data to fill in the needs of users where the data systems, and likewise the UDB, would leave voids." Thus, the DGTB is somewhat similar to the ETES training estimation aids and procedures.

To date, past efforts on UDB development have focussed on (1) an assessment of existing historical data bases which would feed the UDB, particularly the projected portions of the UDB, (2) a description of the weapon system design process with respect to the potential use of the UDB, (3) an assessment of user needs in terms of adequacy of current technology and data[6], and (4) the development of a plan for UDB/DGTB development.

At the present time, a description of the actual data elements to be included in the UDB is not available (this is

---

6 In the examination of the utilization of human resource data in tradeoffs, it is interesting to note that lack of information and lack of appropriate analytical tools were seen as two of the major types of limitations on the use of human resource assessment.

to be developed in future phases of the study). However, by examining the types of historical data bases which are projected to be used by the UDB, it is possible to make some estimates of what it will contain and to assess some of its potential "limitations." These "limitations" point out the differences between the UDB and the ETES SDT. These differences are actually quite significant despite the similarity in the goals of these two systems (see Table 2-10).

The first limitation of the UDB is its emphasis on maintenance tasks and personnel. The UDB, like the Air Force Coordinated Human Resources Technology, emphasizes maintenance behavior and the use of historical data bases related to maintenance. There is little relevant discussion of the procedures and mechanisms for developing or describing operator tasks or training requirements.

This emphasis on maintenance tasks is closely related to a second "limitation" of the UDB; namely, its emphasis on aircraft systems and on Air Force data bases. In the Air Force, the role of enlisted operators is much less significant than it is in the Army or Navy. Hence, it is not surprising that the UDB has focused on the maintenance of aircraft systems.

Third, there are numbers of other system elements which the UDB would appear, at least at the present time, not to describe. These elements include functional requirements, collective or team tasks, task characteristics, and performance data suitable for training and human factors analytical activities, and training program elements. (This failure to describe certain elements would not be critical if the UDB had the proper data base management structure to

Table 2-10

LIMITATIONS OF THE UDB

- Focusses almost exclusively on maintenance tasks
- Emphasizes aircraft systems
- Does not appear to adequately describe functional requirements, collective or team tasks, task characteristic or performance data, and training program elements
- Is not based upon comprehensive data base management system or structure
- Is geared for use by sophisticated users
- Cannot generate tasks and other input data

handle additional system elements. Unfortunately, it appears that it does not have this capability).

Fourth, and perhaps most important, the UDB again does not appear to be based upon a data base structure--that is, a structure which represents the implicit relationships among the various system elements. Such a data base management structure would provide a mechanism for describing the basic structure of a developing system which was independent of the various user viewpoints of the data. This data independence would increase the capability for relating various descriptions of the system to one another, for updating and refining the data, and for adding new elements to the data base in a systematic modular fashion with minimum destruction of existing programming--thus providing the basis for a true data base management capability.

Fifth, the UDB appears to be geared for use by technical personnel who have sophisticated analytical and/or computer programming experience--unlike the SDT which is geared for use by personnel with little background in computers. Because of this difference in emphasis, it is not surprising that the UDB does not specify or deal with the human factors of man-computer interactions as will the SDT, which will be specifically geared for utilization by uninitiated users and will attempt to employ the latest guidelines on human-computer interfaces (see Appendix D). Because of its lack of consideration of human factors issues, the UDB does not attempt to provide procedures for assisting the user in generating tasks or other input data elements.

2.5.3 Consolidated Data Base (CDB) of HARDMAN Methodology

The Navy has a program, called the HARDMAN program (hardware versus manpower procurement), to develop a methodology to systematically assess the manpower, personnel, and training requirements of emerging weapons systems, with particular emphasis on developing predictions for the early phases of the acquisition process. The HARDMAN methodology has been applied to a number of different Navy systems and has been modified for use by the Army and applied to the Enhanced Self-Propelled Artillery Weapon System (ESPAWS) (see Dynamics Research Reports 1980A, 1980B, and Mannle 1980 for a discussion of HARDMAN).

The application of the HARDMAN methodology is supported by the development of a system-specific "data base" which is designed to contain all of the inputs and outputs of each of the steps in the HARDMAN methodology and provide an audit trail for monitoring the data elements which are developed. Table 2-11 lists the data elements described by the CDB.

Like the other current human resource data bases, the CDB has several limitations with respect to the SDT requirements.

The major limitation of the CDB is that only parts of it are automated. Thus, it can not provide a computerized data base management capability. Another major limitation of the CDB is that, like the UDB, it does not contain a systematic scheme for relating the various system elements to one another, a scheme which would be independent of specific input and output requirements. Thus, the CDB is not really

Table 2-11

DATA ELEMENTS CONTAINED IN CDB

General System

- Requirements Documents
- Study Plans and Objectives
- Technology Base Studies
- Projected Operational Environment
- System Functions and Performance Requirements
- Program Constraints
- Minimal Essential Elements of Information List
- Audit Trail Files
- Worksheets
- CDB Index
- Predecessor Equipment List and Related Data
- Reference Equipment List and Related Data
- Predecessor and Reference Reliability Data

Manpower*

- Workload Taxonomy
- Indirect Workload Factors
- Task Event Networks
- Manpower Model Data
- Manpower Metrics and Associated Values
- System Manning (MOS, Skill Level, Duty Positions)

Training*

- Task and Skill Data
- Course Catalogue
- Course Outlines
- Course Methods/Media
- Course Costing Data
- Course Scenario Information
- Career Path Information
- Training Concept
- Training Device and Equipment
- Steady State Resource Requirements
- Steady State Course Costs
- Replacement Personnel Requirements
- Task Selection and Assignment Algorithms
- Facilities Requirements

Personnel*

- Career Path Data
- Career Path Statistics (Attrition, Promotion, Upgrade)

---

*All elements for predecessor, reference, and baseline systems except where noted.

a true data base management system since it does not have an automated capability for linking various system elements to another or for retrieving data elements.

Finally, the CDB does not provide any extensive automated capabilities for generating input data formats or actual input data elements.

2.5.4 SAT Program for the B-1 Bomber

The Structural Approach to Training (SAT) program for the B-1 bomber represents a relatively early attempt to develop a system-specific data base to support instructional systems development (see Sugarman, Johnson and Ring, 1975).

The SAT consisted of two major elements, a data base (the contents of which are displayed in Table 2-12) and two computerized aids -- one aid is a sorting model for the storage, retrieval, collating, and updating of mission/function task analyses and supporting data; and the other is an analytical model for providing cost and training estimates of the B-1 bomber training system.

The SAT data base is interesting in that it is probably the only past effort which has attempted to (1) systematically describe task characteristics in a format which is amenable to the application of automated training aids for determining the tasks to be trained and selecting methods and media, and (2) systematically describe the task performance characteristics of equipment (e.g., relationships of tasks to controls and displays). Such task performance data is critical to human task performance simulation models. The SAT also had a number of other

Table 2-12

SAT DATA ELEMENTS AND LIMITATIONS

System Elements Described by SAT Data Base

- Tasks (title, task element number, operator behavior, task duration, crew interaction, previous task element, task characteristics, and performance data)
- Control/display information (associated system, synonyms)
- Behavioral objectives (title, initial conditions, concurrent behaviors, performance criteria, enabling and ancillary objectives, operators, interactions, task elements, objective criticality, objective difficulty)

Limitations of SAT Data Base

- Is geared for one specific system
- Is not designed to provide generic data base management capability
- Does not systematically describe system functional requirements and design concepts
- Does not include training program elements in automated portion of the data base
- Is geared for sophisticated users
- Cannot generate tasks and other input data

interesting features, such as a task action verb dictionary which listed task synonyms.

However, despite its desirable features the SAT data base also has several limitations which restrict its applicability to the SDT. First, the SAT data base elements and programs were specifically designed to fit one system--the B-1 bomber. Thus, all of its task and control/display dictionaries and structures are only applicable to that system. The SAT was not designed to be a generic data base system which could be applied across a wide range of weapon systems.

Second, the SAT does not describe several important system elements such as functional requirements and design/hardware elements.

Third, training program elements are described but not included in the automated data base.

Fourth, the SAT is geared for very sophisticated users with extensive computer experience.

Fifth, the SAT is not structured to assist users in developing input data formats or actual input data elements such as tasks.

2.5.5 Navy Enlisted Professional Development Information Support System (NEPDISS)

The objectives of the NEPDISS are more limited than the goals of the other human resource data bases described above. The NEPDISS is specifically designed to store and retrieve data related to training program development (see Davis, 1977, for a description). Thus, it is primarily designed to describe task and training data (see Table 2-13). Its only description of equipment-related concepts is in the task statements of the task portion of the data base. Other major limitations of the NEPDISS are its lack of capability for describing projected system elements, its total lack of appropriateness for use by uninitiated users, its lack of a capability for generating tasks and other data impacts, and most important, its lack of a true data base management capability for updating and refining system elements.

Despite the weaknesses, it is important to note that the NEPDISS is especially strong in describing task and skill related requirements which are appropriate for training and personnel analysis.

2.5.6 Other Data Bases

There are a number of other data bases which attempt to deal with some of the issues related to the SDT. For instance, the Consolidated Occupational Data Analysis Program (CODAP) and the Training Developments Information System (TDIS) are Army data bases which also deal with task description. The CODAP focusses on tasks from the perspective of a single MOS while the TDIS focusses on common tasks which are applicable across MOS. Neither one is geared for use in describing the design, task, and training characteristics of an emerging weapon system. Nevertheless, the aspects of these systems which are relevant to the SDT (primarily the task descriptions) were examined in detail during the development of the SDT specifications and these systems will continue to be monitored as the SDT is developed.

Table 2-13

NEPDISS DATA LIMITATIONS

- Does not describe system functional requirements, design concepts, training program elements or collective tasks.
- Is geared for use by sophisticated users.
- Cannot generate task and other input data.
- Is not designed to describe projected system elements.
- Does not provide comprehensive data base management capability for updating and refining system elements.

SECTION 3 - SELECTION OF AN AUTOMATED TOOL FOR SDT DEVELOPMENT

This section presents the results of a review of automated tools which were considered as possible vehicles for the development of the SDT. The chapter is divided into six sections. The first section presents an overview of the different types of automated tools which were examined during the review. The next two sections present the results of a review of two different classes of automated tools: requirements analysis tools and data base management systems. The final three sections evaluate database management systems and select a database management system suitable for SDT development, implementation, and operation.

3.1 OVERVIEW OF AUTOMATED TOOLS

The central need for early training estimation is a systematic method of communicating weapon system information to the participants in the acquisition process (e.g.: training developers, combat developers, materiel developers, etc.). These participants are generally uninitiated in the use of computer equipment and systems. With this focus, two major classes of automated tools were examined during Task 1: requirements analysis tools and data base management systems. The review began with an examination of requirements analysis tools and was completed with the review of data base management systems.

As more and more data was obtained on the current procedures and problems, and available tools were examined in detail, a firm picture of the requirements for the SDT developed. It was determined that a data base management system was the tool which could best meet the SDT requirements (see Section 2 for a description of the SDT specifications and Section 4 for a description of a final SDT which incorporates many of the data base management system concepts discussed in this chapter).

3.2 REVIEW OF REQUIREMENTS ANALYSIS TOOLS

The review of requirements analysis tools was conducted in a three-stage process by DRC's software engineering group. During the first stage, DRC surveyed government reports, IEEE Software Engineering Transactions, and other trade publications to determine what tools were available in the area of requirements analysis. Fortunately, a comprehensive review of requirements analysis tools had just been completed by Devorkin and Obenodorf (1979). Further investigation indicated that this report contained all requirements analysis tools with sufficient maturity for possible use in the SDT.

During the second phase of the review, the methodologies listed in Devorkin and Obendorf were reviewed in more detail. Each review began with an examination of the available literature on the methodology. Following the literature review, individual users were interviewed by phone. With the aid of user comments and knowledge of the SDT requirements, criteria were developed for identifying methodologies with a high degree of potential application to the SDT. The evaluation criteria were as follows:

1. Applicability

The methodology must be capable of building a data base of the conceptual information normally available during the early phases of a developing or evolving system. This data base must be capable of refinement as more specific system information becomes available. It must be capable of describing requirements, design concepts, human tasks, and training program elements.

2. Understandability

The methodology must be capable of being understood by the types of "personnel" who are likely to use the SDT.

3. Demonstratability

The methodology must have been applied to a number of different types of projects.

4. Transportability

The methodology must be capable of being implemented at a minimum of cost on standard business processors used in military/government agencies.

5. Training

The methodology must have an existing formal training program available to the user.

6. Sponsorship

The methodology must have a specific government agency, university or industry committed to enhancing the methodology to meet additional user needs as they become known. The methodology must reside in the public domain.

While investigating the first few methodologies, it became evident that there were two main thrusts in the area of requirements definition methodologies. One thrust emphasized graphics representation, primarily through functional flow block diagrams, as a means of specifying relationships between system elements. Another thrust emphasized a high level conceptual language as the mechanism for specifying relationships between these system elements. Because there was a good deal of overlap between the tools within each of these two thrusts, particularly among the language-based tools which are all basically more advanced derivatives of earlier work conducted by the ISDOS project at the University of Michigan, it was decided that the tools listed in Devorken and Obendorf would be evaluated in terms of the six criteria listed above, and that the tool in each of the two major thrust areas with the highest evaluations on these criteria would be selected for further analysis in the third stage of the review.

Table 3-1 displays the requirements analysis tools which were evaluated during this stage and summarizes their assessment.

The two tools selected for further analysis were the ICAM Definition Language or IDEF, which was determined to be the best graphics based tool, and the Problem Statement Language/Problem Statement Analyzer, which was selected as the best language based tool. During the third stage of the review, these two tools were examined in even greater detail.

Table 3-1

REQUIREMENTS METHODOLOGIES AND RESULTS OF PRELIMINARY ASSESSMENT

Methodologies*	Preliminary Assessment	Requiring Further Investigation
Computer-Aided Design and Specification Tool (CADSAT)	Tool is derived from PSL/PSA but lacks continuing support by a government agency	no
Design Analysis System (DAS)	Tool has insufficient history of usage and is intended for in-house usage at Hughes	no
Digital System Development Methodology (DSDM)	Tool is not automated and has insufficient history of usage	no
Echo Range Methodology	Tool has insufficient history of usage and is geared for use for sophisticated users	no
Higher Order Software (HOS)	Tool is highly mathematical and not geared for use by uninitiated users	no
ICAM Definition Language (IDEF)	Selected as best representative of graphics-based tools	yes
Input/Output Requirements Language (IORL)	Tool is geared for use by rather sophisticated users	no
Martin Marietta System Design Methodology	Tool has insufficient history of usage and is directly derived from PSL/PSA	no
Program Specification Language/Program Specification Analyzer (PSL/PSA)	Selected as best representative of language-based tools	yes
Requirement and Development Language/Requirements and Development Analyzer (RDL/RDA)	Tool is in development stage and is directly derived from PSL/PSA	no
Software Factory	Tool is currently not a comprehensive automated system	no
Software Requirements Engineering Methodology (SREM)	Tool is derived from PSL/PSA but does not have the user history that PSL/PSA has had	no
Structured Analysis and Design Technique (SADT)	Tool is not automated. It is actually more of a methodology than a tool. Best features of tool are contained in IDEF.	no
Xerox System Methodology	Methodology is currently under development and not yet transportable.	no

*List derived from Devorkin and Obendorf (1979).

- AUTOIDEF

The IDEF tool was developed by the Air Force's Integrated Computer-Aided Manufacturing (ICAM) Project Office. IDEF was originally developed to describe the "Architecture of Manufacturing" for an idealized computer aided manufacturing plant. The IDEF format is very similar to the Structured Analysis and Design Technique (SADT) developed by Softech, Inc. and, in fact, is derived from it.

The IDEF had several advantages over other tools which were readily apparent. First, the ICAM project office has a long-term commitment to continuing to develop IDEF as new needs are uncovered. Second, IDEF was recently automated in a version called AUTOIDEF. Before this automated capability, the IDEF tool had no real capability for automatic storage, update, and retrieval of the diagrams which are its major mechanism for describing information. Thus, without this automated capability, IDEF would not merit even initial consideration as a SDT vehicle. Third, AUTO IDEF is supported by a software package developed on Wright Patterson's CDC processor. Fourth, it has been extensively applied within the ICAM project.

To examine AUTOIDEF in greater detail, DRC (1) interviewed several users, (2) obtained and examined in detail AUTOIDEF user manuals, (3) obtained the source code and determined what it would take to transport the system to a non-CDC processor, and (4) obtained a hookup to the Wright-Patterson computer and attempted to utilize AUTOIDEF to describe SDT-related elements for a dummy example. The results of the detailed examination were not encouraging. First, it appeared that in its current state, the AUDOIDEF is difficult to use and requires a fairly long time (one half hour) to develop a single functional flow diagram. In addition, the IDEF does not appear to be a tool which is appropriate for uninitiated users, since it requires learning a relatively complex command language (by SDT standards), and is geared for users with an existing computer background.

- PSL/PSA

PSL/PSA, like two other major language based tools, the Computer-Aided Design and Specification Tool (CADSAT) and the Software Requirements Engineering Methodology (SREM), was derived from initial work done at the University of Michigan ISDOS project. PSL/PSA was chosen over the other tools for the stage three review because it has had wider usage, has several sponsors (University of Michigan and the PSL/PSA Users Group) committed to fund continuing development, and has a fully developed training program.

To examine the PSL/PSA in more detail, DRC (1) obtained and examined the user manuals, (2) determined what it would cost to purchase usage of the PSL/PSA, and (3) sent several members of its software engineering group to a PSL/PSA course to see first-hand what actual PSL/PSA applications looked like. Unfortunately, as with IDEF, the results were not encouraging. PSL is a fairly abstract language that is beyond the capabilities of the uninitiated user who is expected to utilize the SDT. In addition, the documentation fo PSL/PSA is oriented to the technical, rather than the unitiated, user.

- Summary of Review of Requirements Analysis Tools

In summary, current requirements tools do not appear to be suited for the types of uninitiated users who will utilize the SDT. This is not surprising when one considers that all of these tools were specifically designed to describe software requirements for large complex systems. Hence, they are designed to be utilized by technical personnel who have fairly sophisticated backgrounds in computers. (The tools were designed by software specialists for software specialists.)

At a slightly more conceptual level, another factor contributing to the complexity of the requirements analysis tools is that they are designed to be extremely flexible tools which can be utilized to describe any type of system. This type of flexibility necessitates a certain degree of abstractness. This high degree of flexibility and its associated abstractness may actually be a hindrance in describing the elements of the weapons systems in the SDT. (A descriptions of these elements is contained in Section 2 and subsection 3.5.)

Finally, it should be noted that while requirements analysis tools deal with an important aspect of early training estimation (i.e., system description), they are not really geared for dealing with other important ETES related problems; namely, the update and refinement of these system descriptions and their communication to a wide range of participants in the acquisition process.

3.3 REVIEW OF DATA BASE MANAGEMENT SYSTEMS

Data Base Management Systems (DBMSs) were reviewed next. Generally, DBMSs fulfill the SDT evaluation criteria of applicability, understandability, demonstratability, transportability, training, and sponsorship that were identified in Section 3.2. In addition, most DBMSs have the following advantages for the SDT:

1. The DBMSs are designed to store many data items that are related to one another. The SDT consists of many data items with complex interrelationships. Therefore, DBMS technology facilitates the development of the SDT.

2. Data is centrally located and controlled. This simplifies data sharing among multiple users.

3. They can be fitted with data access aids that are easy to use. These aids allow a user to input, modify, delete, and output data using English-like phrases and commands.

4. Access to data items can be restricted. Unauthorized users cannot view, modify, delete, or output restricted data items.

5. They can be structured to present each user with a different view of specific data within the data base. In other words, each user can be presented with data in a format that is meaningful to him alone.

6. The format of a data item is independent of the computer program that is accessing it. This is significant if future software systems--other than the SDT--wish to access the data in the SDT data base. The development of this interface is simplified.

7. Standards can be enforced on data items and on their physical storage in the data base.

3.3.1 Overview of Data Base Management Systems

Before proceeding to examine DBMSs from an SDT perspective, it may be useful to review exactly what a DBMS is. This will be accomplished in a two-step fashion by first defining what a "data base" is and then outlining the essential features of a data base management system.

3.3.1.1 "What is a Data Base?"

An automated data base may be defined as a computerized and integrated collection of stored operational data used by the applications groups of a particular enterprise.[1] The key word in this definition is "integrated." The data elements of a system are likely to have relationships or associations which one could use to link these elements to one another. A data base is integrated when it incorporates information on these relationships as well as information on the data elements themselves. This information can be used to store

---

1 This definition is a modification of an existing definition by Engels (1971)

and retrieve data. It should be noted that, strictly speaking, a data base need not be resident in a computer or its associated media. However, all automated data bases will be stored on a computer or related media and all modern DBMSs are automated. It is clear that only an automated data base can meet the storage and retrieval requirements of the SDT. The term "operational data" is used to refer to data which is pertinent to the ongoing activities of an enterprise. Operational data excludes input data, work queues, output data (such as messages or reports) or any other form of temporary information.

3.3.1.2 Advantages of a Data Base

The major advantage of a data base is that it provides the enterprise with integrated, centralized control of its operational data. This centralized control can, in turn, (1) reduce redundancy in stored data, (2) avoid inconsistency in stored data, (3) allow for greater sharing of data, (4) permit standards to be enforced, (5) permit security restrictions to be applied, and (6) permit a greater capability for checking and maintaining data. If a data base is used in conjunction with a DBMS it can also provide an additional advantage: namely, "data independence." Data independence is achieved by maintaining an internal structure of the data which is independent of the individual applications of the data and individual user viewpoints. This data independence may be contrasted with data dependent systems in which the way data is stored and the way it is accessed are dictated by the structure of the applications.

3.3.1.3 Data Base Management System

Perhaps the best way to describe the essential features of a DBMS is to outline an "architecture" for a typical DBMS. Such an architecture is displayed in Figure 3-1. This architecture is taken directly from Date (1977). DBMS architectures are typically divided into three general levels: internal, conceptual, and external. The internal level is concerned with the way in which the data is actually stored physically. The external level reflects the users' views of the data. The conceptual level provides the medium for linking the internal and external views. The conceptual model provides a general community view of the data base since it contains an abstract representation of the entire data base. This community view is to be contrasted with the external views of individual users who typically will only have a view of a portion of the data base.

Perhaps another way to describe these three different levels of a DBMS is to refer to the structures that psycholinguists use to describe human language. The external view of a DBMS can be construed as being roughly analogous to what psycholinguists describe as the "surface structure" of language, while the conceptual level can be construed as being analogous to the "deep structure" of language and the internal structure can be construed as being roughly analogous to the physical structures in the brain for representing speech.

It is possible for each external user to have his own "language" for utilizing the data base, although in many cases all or a large number of users can use the same

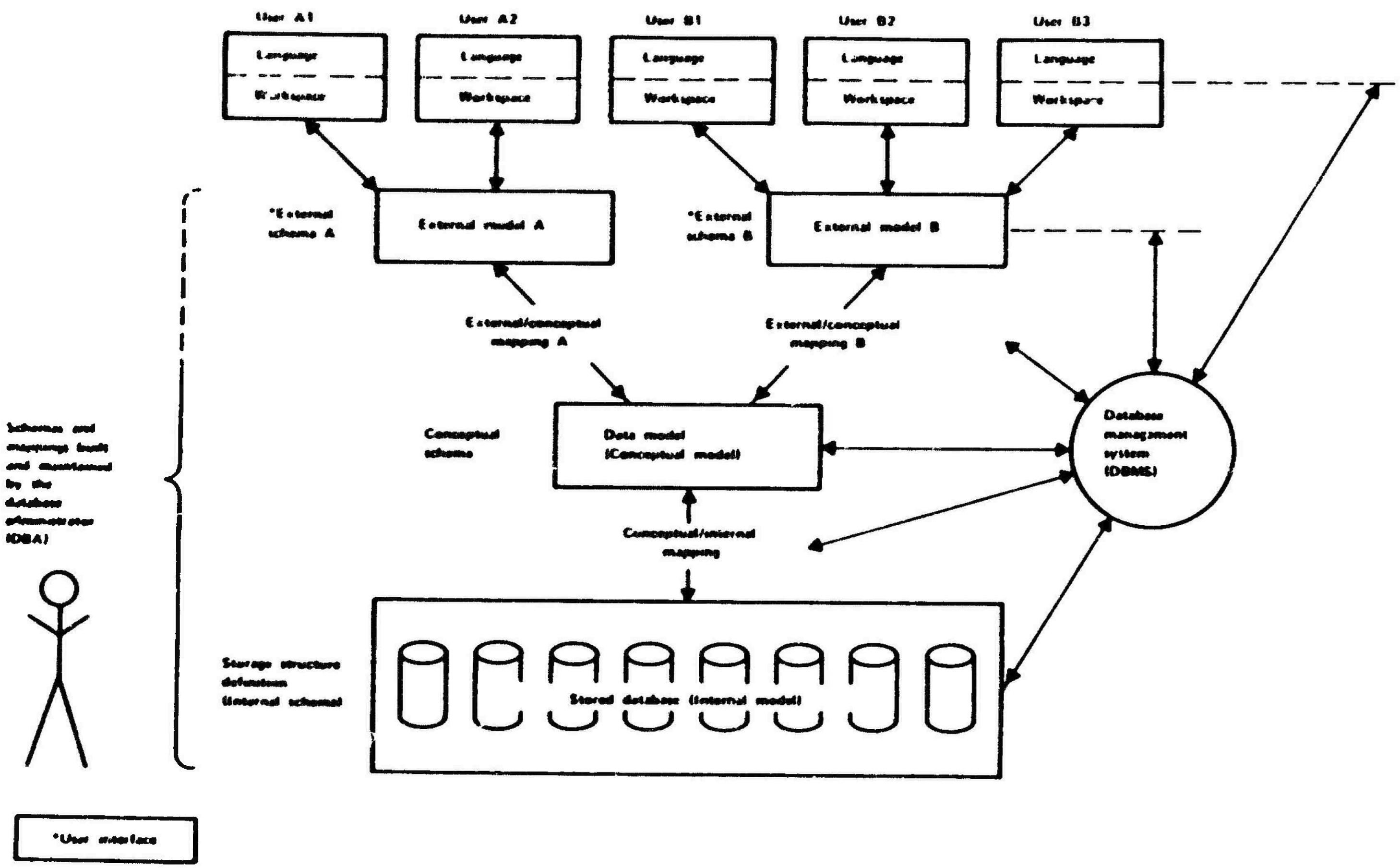


Figure 3-1 AN ARCHITECTURE FOR A DATABASE SYSTEM

*Figure Taken from Date (1977)

language. (As we shall see in the description of the final SDT in Section 4, the SDT will utilize a single language for all users.) A subset of a user's language must include the data sublanguage for storing and retrieving information. Each user may have a workspace for receiving and transmitting data transferred between the user and the data base.

The conceptual model is defined by a conceptual schema which includes a definition of each of the various types of conceptual information in terms of content only (storage or access features are not described). Thus, the conceptual model provides the definition of the total data base content. The conceptual model is critical in that all other aspects of the DBMS are affected by the conceptual model. It has a major effect on the format and structure of the data sublanguage which is used to store, update, and retrieve information from the data base.

3.3.2 Types of Data Base Management Systems

Data Base Management Systems can be categorized by the type of conceptual model they employ to define their data base structure. There are three general categories: the relational approach, the hierarchical approach, and the network approach. More details on these three approaches are presented in the sections which follow.

3.3.2.1 Relational Data Bases

An example of a relational model is contained in Figure 3-2. Each row in the table can be described as an entity while the columns can be described as attributes. Each

S

S No.	SNAME	STATUS	CITY
S1	Smith	20	London
S2	Jones	10	Paris
S3	Blake	30	Paris

SP

S No.	P No.	QTY
S1	P1	300
S1	P2	200
S1	P3	400
S2	P1	300
S2	P2	400
S3	P2	200

P

P No.	PNAME	COLOR	WEIGHT	CITY
P1	Nut	Red	12	London
P2	Bolt	Green	17	Paris
P3	Screw	Blue	17	Rome
P4	Screw	Red	14	London

S = Supplier
P = Part

FIGURE 3-2 RELATIONAL DBMS FORM

Figure Derived from Date (1977)

table represents a series of relationships between entities and between entities and attributes. A crucial feature of a relational data structure is that associations between rows or entities are represented solely by the data values in columns drawn from a common domain. It is characteristic of the relational approach that all information in the data base, both entities and associations, are represented in a single uniform manner, namely tables. This uniformity of data representation leads to a corresponding uniformity and simplicity in the commands required in the data sublanguage to utilize a relational data base (e.g., delete). Therefore, relationally structured data bases are generally easy to understand and use.

3.3.2.2 Network Approach

The network approach is similar to the hierarchical approach in that it has several different types of records, which are associated with one another via links (Figure 3-3). However, a network is a more general structure than a hierarchy because a record may have any number of immediate superiors-unlike the hierarchical approach which has one superior. The network approach thus makes it easier to represent many-to-many correspondences more directly than does the hierarchical approach. Therefore, a network structured database can represent "typical" relationships among data items, more so than a hierarchical structure.

3.3.2.3 Hierarchical Approach

Just as the basic model underlying the relational model can be represented by a table, the hierarchical model can be represented by a tree structure (see Figure 3-4). The

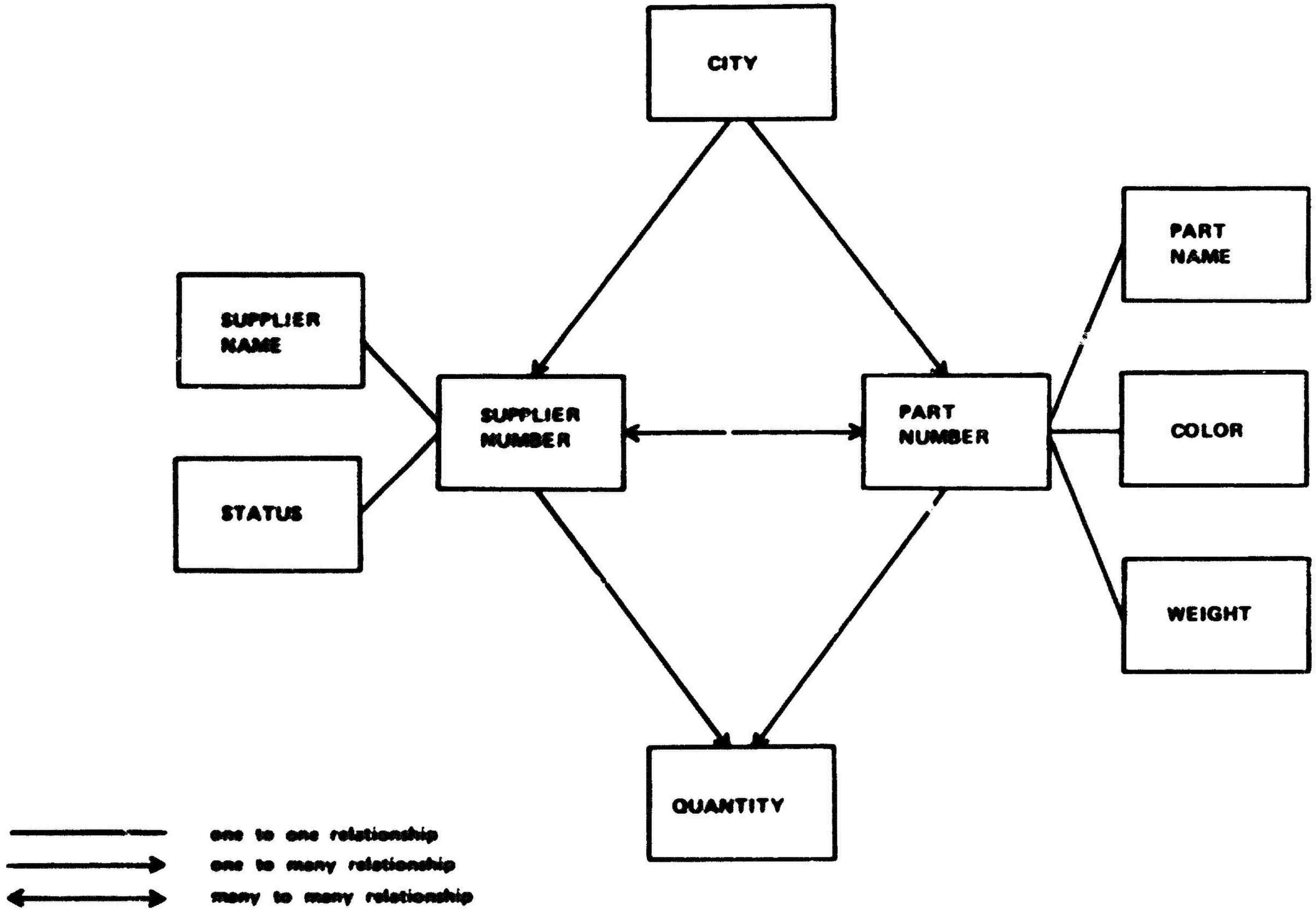


FIGURE 3-3 NETWORK DBMS FORM

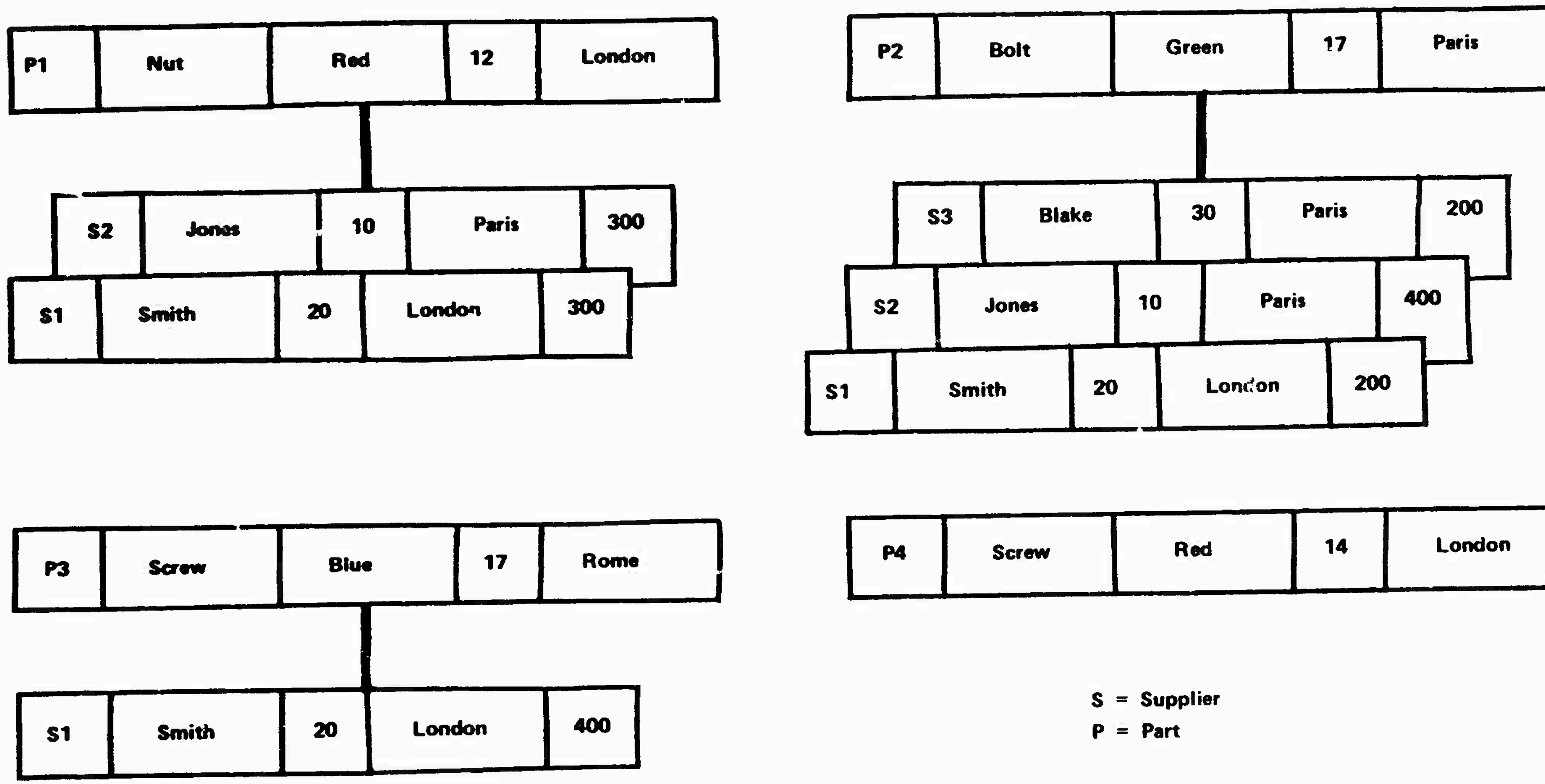


Figure 3-4 HIERARCHICAL DBMS FORM

* Figure Derived from Date (1977)

hierarchical model can be described as a single file, containing records arranged into trees. The files in a hierarchical approach can be more confusing than the files in a relational approach because (1) they contain several different types of records and (2) they contain links connecting occurrences of these records. A fundamental aspect of the hierarchical approach is that any record occurrence can only be accessed when its context (that is, a record's relationship to its superior) is taken into account.

3.4 THE APPLICATION OF DBMS TECHNOLOGY TO THE SDT

Because DBMS technology can fulfill the requirements of the SDT better than existing requirements analysis tools, the SDT will be developed using a DBMS.

Selecting a DBMS for a particular application--such as the SDT--is difficult. Many factors affect this decision. Some of these factors are the following:

- Logical structure of the data base at the conceptual level,

- Specific features of the DBMS,

- Capabilities of supporting facilities, and

- Cost of the DBMS and its supporting facilities.

For any application, a DBMS with a relational structure at the conceptual level has several advantages. First, a relational data base is easy to develop and comprehend.

Therefore, developing software systems that interact with a relational data base is less complicated. Second, user access to data items is simple. Inquiries and retrievals are conducted through tables of related data items. Third, this structure facilitiates access to both individual data items and groups of data.

User interaction with a relational DBMS may be slower. This is because the relational DBMS software performs more tasks to process a given command than a network or hierarchical DBMS. Also, relational DBMSs are relatively new and are not as developed as network or hierarchical DBMSs.

The network structure reflects real world situations because it may be accessed from any point within the data base. Also, this structure has the support of the Conference of Data Description Languages (CODASYL) Programming Language Committee (PLC) Data Base Task Group (DBTG), the group that proposed standards for DBMS programming languages.

The representation of relationships among data items in a network data base may be complex. This can complicate the development of application programs that interact with the DBMS.

For data items that logically fit into a hierarchical structure, a hierarchical DBMS is appropriate. The structure of a hierarchical data base is easy to understand. Therefore, developing software systems that interact with a hierarchical data base is simplified.

The disadvantages of a hierarchical DBMS are two. One, data items do not always logically fit into a hierarchical structure. Two, access to a hierarchical data base is limited (i.e., from the top of the structure to the bottom of the structure). Therefore, the design of the data base has a great impact on the amount of time the DBMS requires to access a particular data item. Data items near the top of a hierarchical structure can be accessed more quickly than those at the bottom of the structure.

For the SDT, the order of desirability of the logical structure of the DBMS at the conceptual level is relational, network, and hierarchical. The remaining DBMS features are examined in Section 3.6.1.

3.5 RESTRUCTURING OF THE SDT INTO A RELATIONAL FRAMEWORK

Because a DBMS with a relational structure was preferred, the SDT components were restructured into a relational-like framework. The components are described in "systems" terms and their interrelationships are defined.

3.5.1 Discussion of the SDT from an Entity-Attribute-Relationship Perspective

A number of systems analysts (e.g., Teichroew, Mascovic, Hershey, and Yamamoto, 1980; and Chen 1976) have pointed out that systems can be described in terms of three basic elements: entities, attributes, and relationships. These three elements can be described as follows:

- Entities - Correspond roughly to nouns in English language. They are those objects and ideas which can be used to describe basic system elements.

- Attributes - Correspond roughly to adjectives in English language. Attributes formalize important properties of entities. Each attribute has associated with it a set of values.

- Relationships - May be compared with English verbs. More properly, they correspond to the mathematical definition of binary relations; statements of associations between two elements.

This simple entity-attribute-relationship framework has wide ranging implications. For instance, this framework has served as the cornerstone for requirements analysis tools developed in the ISDOS project at the University of Michigan. Thus, it provides the foundation for all language-based requirements analysis. More important, as was noted above, it is directly congruent with the basic elements of a relational data base where entities correspond to rows in a relational table, attributes correspond to columns, and relations correspond to the table entries.

The SDT was restructured into an entity-attribute-relationship framework and implicit entity and attribute classes and SDT relationships were identified. Table 3-2 lists the implicit entity and attribute classes which were developed for the SDT. Table 3-3 describes the different types of relationships which will be required and Table 3-4 describes how these different types of relationships can be applied to the different classes of entities and attributes.

Table 3-2

IMPLICIT ENTITY AND ATTRIBUTE CLASSES FOR DEVELOPING WEAPON SYSTEMS

Entities	Major Attributes
Functional requirements data - Functions	Performance goals, threat impacts, environmental impacts, mission profile impacts
Design Data	
- Generic equipment functions	Approval status; comparable existing equipments
- Design alternatives	Comparable existing equipments, degree of difference between existing equipment , reliability
- Design alternative component inputs	
- Design alternative component outputs	
Tasks/software functions	
- Human tasks (individual)	Performance data, conditions, standards, initiating cues, terminating cues, no. of people performing, amount of supervision, task characteristics, task assignments, task type, task elements
- Human task inputs	
- Human task outputs	
- Software functions	
Tools/test equipment	
- Tools	Tool type, comparable existing tool
- Test equipment	Test equipment type, comparable existing tool
Personnel	Function (operator, maintainer, other), MOS, skill level, paygrade, duty position number
Skill and knowledges	Skill and knowledge characteristics
Performance measures	PM type
Learning objectives	LO type, training setting, training method
Courses	
- Course modules	Course seq. no.; course no.; title, length, hours; method, student/instructor ratio
Media	Media type; training setting
ARTEP (collecting tasks)	Related manuals

*Attributes specifying relationships between entities are not listed (e.g., a number or code relating tasks to equipment).

Table 3-3 TYPES OF RELATIONSHIPS REQUIRED BY SDT

(H) Hierarchical Relationships - A is a member of B. Relationship indicates that one entity is a member of larger class of entities.

(A) Activity Relationshps - A occurs before (or after) B. Relationship indicates the sequence in which entities (functions or tasks) are performed.

(IO) Input/Output Relationships - Entity A is an input (or output) of entity B. Relationship indicates inputs and outputs associated with entity.

(As) Associative Relationships - A is associated with B. Purely associative relationships.

(D) Duplicative Relationships - A duplicates B. Different versions of a general entity class for different design alternative.

Table 3-4 IMPLICIT RELATIONSHIPS IN SDT

Hierarchical Relationships (H)

- Within each entity class, each class can be subdivided into a number of subclasses.
- Each attribute class can also be further subdivided into a number of subclasses.

Activity Relationships (A)

- Activity relationships are required for lower level functional requirements
- Activity relationships are required for tasks (separate relationships for operator and maintenance tasks)
- Activity relationships are required for courses (to represent course sequence)

Input/Output Relationships (I/O)

- Input/output relationships are required for generic equipments
- Input/output relationships are required for design alternatives
- Input/output relationships are required for tasks

Associative Relationships (As)

- All attributes must be associated with their respective entities.
- The following items must also be associated with one another
  - Generic equipment with function
  - Design alternative with functions and generic equipment
  - Tasks with functions, generic equipment, design alternatives, tools and test equipment, personnel
- Software functions with functions; generic equipment; design alternative; tasks
- Tools and test equipment with design alternatives
- Skill and knowledges with tasks
- Learning objectives with tasks, skills and knowledges, media, courses, and course modules
- Performance measures with tasks, learning objectives and skills and knowledges
- Courses with personnel, design alternatives and generic equipment
- Course modules with learning objectives, tasks, skills and knowledges and media
- Media with tasks
- ARTEP tasks with media, tasks and functions

Duplicative Relationships (D) - possible with any entity

3.6 SELECTION OF A DBMS FOR THE SDT

The selection procedure consisted of the following three steps:

1. Determine the requirements of the SDT that apply to the selection of a DBMS (Section 3.6.1),

2. Select the DBMSs that fulfill these requirements (Section 3.6.2),

3. Comapre the selected DBMSs to determine the DBMS(s) most applicable to the SDT (Section 3.6.3).

Section 3.6.4 reviews the applicable DBMSs. Alternatives for developing and operating the SDT are considered in Section 3.6.5. A specific recommendation for developing and operating the SDT is given in Section 3.6.6.

3.6.1 Determination of the SDT Requirements that Apply to the Selection of a DBMS

A DBMS with a relational structure at the conceptual level was desired for the SDT. However, this must be weighed against other DBMS features that are necessary for development of the SDT. These features are the following:

1. Concurrent batch and on-line applications - A batch operation (i.e., reading punched computer cards) can be performed at the same time as an on-

line operation (i.e., a user accessing the DBMS through a CRT terminal).

2. Concurrent on-line access for multiple users - More than one person can simultaneously access the DBMS through separate CRT terminals.

3. Security restrictions of the DBMS - The DBMS is inaccessible to unauthorized users.

4. Aids for developing user-friendly interfaces - Programming tools that simplify the use of a DBMS.

5. Query-facility - An automated aid that simplifies the examination and retrieval of data items in a data base.

6. Data dictionary - A software tool that contains descriptions of data items and their relationships, but not the data items themselves. It is used to control the development and operation of a data base.

7. Report generator - An automated utility that simplifies the formatting and output of printed reports on the data in a data base.

8. Variety of available application languages - The DBMS must interface with more than one programming language (i.e., COBOL, PLI, FORTRAN, PASCAL, etc.).

9. System accounting facility - An automated utility that monitors the use of the DBMS resources and its supporting facilities.

10. Journaling or logging facility - An automated utility that monitors additions, changes, or deletions of data in the data base. It is used as an "audit trail" for data base operations.

11. Recovery facilities - Automated utilities that restore a data base to its configuration at an earlier point in time. They are used after a computer failure to return the contents of a data base to their previous values.

12. Variable length segments - the DBMS will accept data items whose physical storage length may vary. This feature is useful for data items with unknown storage requirements or for data items with storage requirements that could change.

These 12 DBMS characteristics apply to the development and operation of the SDT. Their presence in a DBMS is required.

3.6.2 Selection of DBMSs that Fulfill the Requirements of the SDT.

Fifty-one (51) commercially-available DBMSs were surveyed through a study of DATA PRO reports (70E-01B-61a and D30-100-002) and other literature.

Table 3-5 summarizes the characteristics of the 51 DBMSs that were surveyed. It consists of six items: Vendor of

TABLE 3-5

CHARACTERISTICS OF COMMERCIALLY-AVAILABLE DBMSs

DBMS	Vendor of the DBMS	Supporting Hardware	Approximate Usage	Primary Data Organization	Approx. Price	Applicability to SDT
*ADABAS	Software AG of North America	IBM: 360,370, 303x, 4300	Moderate	Network	$2500/ month $40-160K	Very high
ADMINS/II	Admins, Inc.	DEC: PDP-11, VAX-11	Very low	Relational	N/A	Low
AMBASE	Amcour Computer Company	DEC: PDP-11	N/A**	N/A	$15.5K	Moderate
BASIS	Battelle, Columbus Laboratories	IBM; CDC; Cyber; Univac; DEC	Very low	Relational-like	$38K	Moderate
CREATE	Complete Computer Systems	Data General: NOVA & Eclipse	Very low	N/A	$18K	Moderate
*DATACOM/DB	Applied Data Research, Inc.	IBM:360,370	Low	Relational	$47-57K	Very high
DATA DEMON	Gemini Information Systems, Inc.	Perkin-Elmer; IBM: Series/1	Very low	N/A	$1000/ month $17.5K	Low
DBM-1	Condor Computer Corp.	Cromenco: System/3	Very low	N/A	$10K	Low
DBMS	Prime Computer, Inc.	Prime: 400 & 500	N/A	Network	$20K	Low
DBMS 2	EGS Systems, Inc.	Modular Computer: MODCOMP	Very low	N/A	N/A	Very low
DBMS-10	DEC	DEC: System-10	Low	Network	$30K	High
DBMS-11	DEC	DEC: PDP-11	N/A	Network	$16.5K	Low
DBMS-20	DEC	DEC: System-20	Very low	Network	$30K	High
DBMS-300	Compudata Systems, Inc.	DEC: 300 Series	Very low	N/A	$100/mo $5K	Very low
DBMS-990	Texas Instruments, Inc.	TI: DS990, Models 6 & 8	N/A	N/A	$2K	Very low
DL/I DOS/VS	IBM	IBM: 370, 303x, 4300	High	Hierarchical	$434/mo	High
DMS II	Burroughs Corp.	Burroughs: B 700 or 800	Low	Network	$23.25K	High
DMS 90	Sperry Univac	Univac: Series 90 or 80	Very low	Network	N/A	Moderate
DMS-170	Control Data Corp.	CDC:6000; Cyber 70, 170, 700	Very low	Network	$730/mo	High
DMS-1100	Sperry Univac	Univac: 1100 Series	Moderate	Network	N/A	High
DMS/1700	Dedicated Systems, Inc.	Burroughs: B1700	N/A	N/A	[illegible]K	Low
DNA-Data Base Manager	Exact Systems & Programming Corp.	DG: Nova or Eclipse	Very low	N/A	N/A	Very low

*DBMS selected for further study.

**Not available.

Table 3-5 (continued)

DBMS	Vendor of the DBMS	Supporting Hardware	Approximate Usage	Primary Data Organization	Approx. Price	Applicability to SDT
DPL	National Information Systems, Inc.	DEC: System 10 or 20	Low	Hierarchical	$985/mo $38-47K	High
DRS/XBS	A.R.A.P.	IBM, DEC, Univac, CDC	Low	Network	$22-60K	High
EASE DBMS	Bloodstock Computer Services, Inc.	DEC: PDP-11	Very low	N/A	$6.5K	Low
GIS/2	IBM	IBM: 360 or 370	N/A	Hierarchical	$620-970/mo	Moderate
IBDB	Tesseract Corp.	IBM: Series/1	Very low	N/A	$4.1K	Low
*IDS—I/II (DM-IV)	Honeywell Info. Systems, Inc.	Honeywell: Series 6000 & 60 Level 66	High	Network	$400-$2K/mo	Very high
*IDMS	Cullinane Corp.	IBM: 360,370, 303x, & 4300	Moderate	Network	$50K/yr	Very high
IDOL	Science Management Corporation	Wang: 2200; IBM: Series 1	Low	N/A	$364/mo $9500	Moderate
IMS	IBM	IBM: 360,370, 303x, 4300	Very high	Hierarchical	$1045/mo	High
Infoflex DBM	Interactive Info. Systems, Inc.	DEC: Datasystem 500 Series	Very low	N/A	$12K	Low
INFOMEDIA	Mead Technology Laboratories	IBM: 360 or 370	N/A	N/A	$2000/mo $140K	Low
INFOTRIEVE	Educational Data Systems	EDS: Point 4; DG: Nova	Very low	N/A	$2000	Moderate
INGRES	INGRES, Inc.	DEC: PDP-11	Low	Relational	N/A	Low
IQ/NET	Infodata Systems, Inc.	IBM: 4300	N/A	Network	$40K	Moderate
INQUIRE	Infodata Systems, Inc.	IBM: 360, 370	Low	Network	$70-150K	High
MADMAN	G.E. Company	DEC: PDP-11	N/A	Relational-like	$20K	Low
MIDMS	National Technical Info. Service	IBM: 360	N/A	N/A	$450	Moderate
MINDS	Minnesota Datasystems, Inc.	BTI: 4000, 5000, or 8000	Very low	N/A	$3-5K	Low
*MODEL 204	Computer Corp. of America	IBM: 360, 370, 303x, or 4300	Very low	Network	$80-150K	Very high
OASIS	University of Windsor	IBM: 360, 370, or 303x	Very low	N/A	$30K/yr	Low
ORACLE	Relational Software, Inc.	DEC: PDP11 or VAX	N/A	Relational	$48-96K	Moderate
OS 200 DB	Honeywell	Honeywell: 200 or 2000	Very low	N/A	Bundled free with Hardware	Very low
PLUS/4	Century Analysis, Inc.	NCR: 101 or above	N/A	N/A	$10K	Low
OCRT	The Management Group, Inc.	IBM: 360 or 370; Honeywell: 6000	Very low	N/A	$25K	Moderate

Table 3-5 (continued)

DBMS	Vendor of the DBMS	Supporting Hardware	Approximate Usage	Primary Data Organization	Approx. Price	Applicability to SDT
*RAMIS II	Mathematica Products Group	IBM: 360 or 370	High	Hierarchy	$22-43K	Very high
*SEED	International Data Base Systems, Inc.	IBM; CDC; HP; and DEC	Very low	Network	$8.5-25K	Very high
Supersetup	The Automated Quill, Inc.	DG: Nova or Eclipse	Very low	N/A	$4.9K	Low
SYSTEM 1022	Software House	DEC: System 10 or 20	Low	Relational	$24K	High
SYSTEM 2000/80	Intel Systems Corporation	IBM; CDC; and Univac	Moderate	Network	$45K	High
TOTAL	Cincom Systems, Inc.	IBM & most major minicomputers	Very high	Network	$18.5K	High

the DBMS, Supporting Hardware, Approximate Usage, Primary Data Organization, Approximate Price, and Applicability to SDT.

Vendor of the DBMS refers to the company that distributes the DBMS. It was included to provide additional information.

Supporting Hardware refers to the computer hardware configuration on which the DBMS will operate. In some instances, only the names of the hardware manufacturers were included. In these instances, the DBMS will operate on more than one configuration (or model) of the listed computer hardware.

Approximate Usage is an estimate of the number of installations of the DBMS. It is based on the following scale:

- Very high - greater than 1,500 installations,
- High - 1000 to 1,499 installations,
- Moderate - 500 to 999 installations,
- Low - 100 to 499 installations, and
- Very low - less than 100 installations.

It infers a rough measure of the popularity of a DBMS within the computer-user community. However, this inference does not imply that a greatly used system is better than one of less usage.

Primary Data Organization is the logical structure of the data base at the conceptual level. The structure can be relational, network, hierarchal, or a combination of these three. However, only the most commonly referenced structure is listed.

Price refers to the basic system purchase price that was quoted to DATA PRO in late 1980. The purchase price generally includes an unspecified monthly maintenance charge. Monthly or yearly lease/rental plans are occassionally included.

The column entitled Applicability to SDT is a composite of the 12 SDT requirements identified in Section 3.6.1. Each DBMS was examined to determine how many of these 12 requirements it fulfilled. The scale used follows:

- Very high - All 12 requirements fulfilled,
- High - 9 to 11 requirements fulfilled,
- Moderate - 5 to 8 requirements fulfilled,
- Low - 3 to 4 requirements fulfilled, and
- Very low - 2 or fewer requirements fulfilled.

This column is the deciding factor for selecting DBMS alternatives for the SDT. Only those DBMSs that ranked very high (fulfilled all 12 SDT requirements) were selected for further analysis. These DBMSs are marked with an asterisk (*).

3.6.3 - Comparison of the Selected DBMS Alternatives

The seven DBMSs that were selected for further analysis were ADABAS, DATACOM/DB, IDS-I/II (DM-IV), IDMS, MODEL 204, RAMIS II, and SEED. Each of these DBMSs fulfilled the 12 SDT requirements.

Table 3-6 displays the factors that were selected to evaluate the seven DBMS alternatives. These factors, and their rating systems, follow:

I Primary Data Organization - The major data base structure at the conceptual level of system architecture. The three main types, ranked in order of their applicability to the SDT, are relational (3), network (2), and hierarchical (1).

II Usage - The number of installations of the DBMS. The scale for usage is: over 1,500 installations (3), 500 to 1,500 installations (2), and under 500 installations (1).

III Selected Features - Of the 12 SDT requirements examined earlier, six were selected for further study. These six requirements were fitted to the following scale: enhanced (3), sufficient (2), and insufficient (1). The scale signifies the extent to which the SDT requirement is fulfilled. For example, a rating of 1 states that the DBMS does not sufficiently fulfill a SDT requirement.

1. System security refers to the extent of data protection in the DBMS. The characteristic of an insufficient security facility is DBMS validation of the user's password. A sufficient facility has password validation and protects data from unauthorized modification. An enhanced facility combines password validation and modification protection with protection against unauthorized data viewing and/or data encryption.

TABLE 3-6

EVALUATION OF SELECTED DBMS ALTERNATIVES FOR THE SDT

	ADABAS	DATACOM/DB	IDS I/II* (DM-IV)	IDMS*	MODEL 204	RAMIS II	SEED*
I. Primary Data Organization	2	3	2	2	2	1	2
Scale { Relational–3, Network–2, Hierarchical–1 }							
II. Usage	2	1	2	2	1	2	1
Scale { >1500 Users–3, 500-1500 Users–2, <500 Users–1 }							
III. Selected Features							
1. System security	2	2	3	2	2	2	3
2. User-friendly aids	2	3	2	3	3	3	2
3. System accounting facilities	2	1	2	3	3	3	2
4. Transportability	2	2	2	2	2	2	3
5. Inquiry/retrieval	3	3	3	2	1	2	3
6. Report generator	3	3	3	3	1	2	3
Scale { Enhanced–3, Sufficient–2, Insufficient–1 }							
IV. Total	18	18	19	19	15	17	19

*DBMSs with greatest total score.

2. User-friendly aids are software tools that simplify user interaction with the DBMS. An insufficient aid simplifies the development of application programs. A sufficient aid has a "HELP" facility to provide the user with on-line documentation about system functions. An enhanced facility provides an on-line tutorial that is oriented to the uninitiated user.

3. System accoutning facilities automatically track the use of system resources. An insufficient facility would only track statistics of DBMS utilization. A sufficient facility would log these statistics and simplify the development of an equitable algorithm for billing system users. An enhanced facility would produce user billing reports based on logged statistics and the billing algorithm.

4. Transportability is the ability to use the DBMS on different computers. An insufficiently transportable DBMS can only be used on a single model of a computer (i.e., IBM 370). A sufficiently transportable DBMS can be used on more than one model of a line of computers of a single manufacturer (i.e., IBM 360, 370, and 3033). Enhanced transportability is a DBMS that can be used on a variety of computers of different manufacturers (i.e., IBM, UNIVAC, and Honeywell).

5. Inquiry/retrieval facility is the utility that simplifies user access to data in a data base. An insufficient facility is used through application

programming languages only. A sufficient facility is available to the on-line user. An enhanced facility has an on-line language with simple, English-like text.

6. Report generators simplify the formatting and output of the data base reports. An insufficient generator is used through application programming languages only. A sufficient generator is available to the on-line user. An enhanced generator has an on-line language with simple, English-like text.

The six SDT requirements that were not selected for further analysis were generally equivalent in each of the seven alternative DBMSs.

The DBMSs with the greatest total scores are most suitable for the develoment and operation of the SDT. These DBMSs, marked with asterisks (*), are IDMS, IDS-I/II (DM-IV), and SEED.

3.6.4 Review of the Applicable DBMSs

IDMS is a Cullinane Corporation DBMS that conforms with the CODASYL PLC DBTG specifications. It operates on several IBM mainframe computers. Some features of IDMS are the following:

- Data dictionary,
- Optional query facility,
- Optional report generator,
- Network data base structure,

- Requires 75,000 bytes of on-line storage space, and
- One year license fee of $50,000.

The strengths of IDMS are its CODASYL orientation and the good reputation of its vendor--Cullinane Corporation. It has over 700 installations.

IDS I/II (DM-IV) are Honeywell software products with over 1,000 installations on Honeywell mainframe computers. IDS I (Integrated Data Store I) was introduced in 1974 and was considered a "defacto" CODASYL PLC DBTG system. IDS II was introduced in 1975 and conforms completely to the CODASYL PLC DBTG standards. DM-IV (Data Manager-IV) is a File Management System that integrates IDS II (the DBMS) with supporting software systems to provide complete data management facilities. Some of the features of DM-IV/IDS II are the following:

- Data Dictionary,
- Query facility,
- Report generator,
- Network data base structure,
- Multiple application languages,
- Requires 12,000 words of on-line storage space, and
- Priced at $2,300/month.

The strengths of DM-IV/IDS II are its CODASYL orientation, popularity, and the support of its vendor - Honeywell, Inc.

SEED is a DBMS product of International Data Base Systems, Inc. It is primarily written in FORTRAN and has been installed on a variety of computers. However, it has relatively few installations. Some features of SEED follow:

- Data dictionary,
- Optional Query facility,
- Optional Report generator,
- Network data base structure,
- Requires 20,000 to 50,000 bytes of on-line storage space, and
- Priced from $9,000 to $15,000.

SEED's strengths are its high degree of transportability and its low price.

3.6.5 Alternatives for Developing and Operating the SDT

Three alternatives for developing and operating the SDT have been identified. They are the following:

1. DRC personnel develop the SDT on non-DRC computer equipment. This equipment is selected for its compatibility to existing Army computer hardware. The SDT is developed with a DBMS that is highly transportable - SEED - and the necessary programming aids. The SDT resides in the selected computer hardware and is accessed through remote and local terminals. The users are responsible for operating the SDT. DRC maintains the SDT, probably through a remote terminal interface.

2. DRC develops, implements, maintains, and operates the SDT on DRC's Honeywell DPS-8/52 computer. The

SDT is developed using DM-IV/IDS II, Middleware, and other programming aids available at DRC. The users access the SDT through remote terminals interfaced to DRC's computer facility.

3. DRC develops and implements the SDT on an Apple II Plus microcomputer with an interface to a DBMS that will reside in DRC's Honeywell computers. The users access the SDT DBMS through remote Apple II Plus Microcomputers with resident SDT software (i.e., the computer programs that will reside in the Apple II Plus microcomputers and perform the SDT functions). DRC operates and maintains the SDT DBMS and maintains the SDT software.

3.6.6 Specific Recommendation for Developing the SDT

At this time, DRC recommends the third alternative - developing and operating the SDT using a DRC DBMS with the user interfaces through Apple II Plus microcomputers.

The first alternative - developing the SDT on a non-DRC computer and DBMS - is rejected because of the cost of familiarizing the DRC staff with this configuration. The SDT could not be developed given the current funding constraints. In addition, the effectiveness of remote terminal maintenance of the SDT is questionable. Travel time for the SDT maintenance crew would have to be included in a cost estimate of this alternative.

The advantages of alternative two - developing the SDT using DRC's Honeywell DPS-8/52 computer and the DM-IV/IDS II - are the following:

1. DRC has the necessary computer hardware and supporting software to develop and operate the SDT.

2. DRC personnel are familiar with this computer configuration.

3. DRC's DM-IV/IDS II is one of the three DBMSs recommended for the SDT.

4. DRC computer facilities are available seven days a week, 24 hours a day.

Alternative two is recommended if alternative three is technically and/or economically infeasible.

Alternative number three - developing the SDT on an Apple computer with an interface to a DRC DBMS-requires further study to determine its technical and economic feasibility.

SECTION 4 - SDT DESCRIPTION

This section provides a detailed description of the SDT. The description builds upon all of the research and development conducted during the first year of the study, including the review of existing DoD and Army policies and procedures (Appendix A), the review of psychological literature relating to the design process (Appendix B), the review of literature relating to human-computer interactions (Appendix C), the SDT information specification (Section 2), and the review of automated system description tools (Section 3). The description outlined in this chapter is expected to serve as an example of the major SDT development efforts which will occur during the second year.

The chapter is divided into seven sections. The first section will provide an overview of the optimal SDT characteristics. The second section will describe the likely users of the SDT. The third section will provide a description of the physical characteristics of the SDT. The fourth section will provide an overview of the basic SDT process. The fifth section will describe its different modes of operation. The sixth section will provide a more detailed description of the operational uses of the SDT by listing examples of the types of interactions that can be expected under each mode of operation. The final section will describe the features of an initial version of the SDT. During the present three year ETES study, the goal will be to develop the initial version and then to augment it with as many features of the optimal version as is possible until the funds allotted in the present study have been expended. This strategy is necessary to insure that an operational SDT product will be available at the end of the study. Actually, even the initial version of the SDT will be a rather sophisticated and powerful tool.

4.1 OVERVIEW OF SDT FEATURES AND RELATIONSHIP TO OTHER SECTIONS

The SDT will be an automated tool for describing actual and projected system elements, including functional requirements, design concepts, task skills, training program elements and their associated resources; for storing the above information; for changing and updating this information; and for transmitting the information among all of the participants in the acquisition process.

A detailed description of the information elements which will be described by the SDT is in Section 2. Description of the use of the SDT within the context of early training estimation and existing Army policies and procedures is in Appendix A and Section 2.4.

As a result of the review of automated tools described in Section 3, a data base management system was determined to be the best vehicle for the SDT. Finally, the SDT will be specifically designed to meet the human-computer interaction requirements for uninitiated users identified in Appendix C.

4.2 USERS OF SDT

The SDT will be accessed by three different groups: (1) Primary Users - Primary Users are the people who will actually use the SDT on a "day-to-day" basis to describe and update system concepts and to obtain information on current system characteristics (Table 4-1 lists the organizations that are likely to be the primary users of the SDT for an emerging weapon system), (2) Data Base Directors (DBDs) - the DBDs validate the design and utilization of the SDT data base, and (3) SDT Management Group - the SDT Management Group maintains and operates the SDT.

4.2.1 Primary Users

Each primary user will be connected to the SDT by at least one remote terminal. Some primary user organizations (e.g., training developments and the DARCOM PM) are likely to have more than one terminal since they will have a number of individuals with a need for SDT data base information. It is expected that all primary users will interface with the SDT in an interactive mode. To input data in a batch mode, they must transmit this data to either the Data Base Directors or the SDT Management Group who will then input the data into the system. It is expected that the primary users of the SDT will have little, if any, computer skills. Consequently, all of their interactions with the SDT will be through a highly transparent user interface that will utilize menu-selection, form-filling, and question-and-answer computer dialogue techniques to elicit input data and commands (see Appendix C for a discussion of these techniques). This type of transparent interface will mean that the users will only be required to learn the commands associated with calling up the SDT system. From that point on, they will be led through the utilization of the SDT and will not have to generate any more commands on their own. (They should, of course, have read the SDT users manual to find how the SDT can, and should, be used.)

Table 4-1 PRIMARY USERS OF SDT

- TRADOC System Manager (TSM) for System
- Training Developments (within Related School)[1]
- Combat Developments (within Related Mission Area)
- DARCOM Program Management Staff for System[1]
- TRADOC Systems Analysis Activity (TRANSANA)
- DARCOM Materiel Readiness Support Activity (MRSA)[2]
- Individual Contractors
- Others

[1]Indicates organization likely to have more than one terminal interfacing with SDT.

[2]The MRSA connection with the SDT will be designed to provide an SDT interface with the automated LSAR.

4.2.2 Data Base Directors (DBDs)

The DBDs will have the same capabilities as the primary users for entering, storing, and accessing SDT information. The Data Base Directors will have two additional responsibilities:

a. The DBDs will be responsible for overseeing the general development of a system-specific SDT data base. In this role, they will direct the SDT Management Group to set up and maintain a data base for the system; direct others to input, update, and utilize SDT data; and determine what data elements can be changed, who can change them, and when they can be changed.

b. The DBDs will have the capability, together with the SDT Management Group, for batch input and for producing block diagrams to represent various system relationships.

It is expected that the DBDs will also be uninitiated users with little, if any, computer experience. Consequently, they will interact with the SDT via the same transparent user interface that will be used with the primary users (i.e., menu selection, form-filling, and question-and-answer dialogues). Management directions for the SDT will be transmitted via normal communications media (e.g., mail, phone) and not through the automated SDT.

It is likely that one of the user organizations listed in Table 4-1 will also fill the Data Base Directors role (the most likely candidates being Training Developments, the TSM, or the DARCOM PM). Exact specification of an organization to fulfill this role will be made during the ETES implementation phase. It may be necessary to create a multidisciplinary group from combat developments, training developments, and the TSM group at a specific school to assist the DBDs in performing their functions rather than relying on a single organization.

4.2.3 SDT Management Group

The SDT Management Group will be responsible for overseeing the application of the SDT on an Army-wide basis. In this capacity, they will:

- Maintain and update the SDT-DBMS including computer programs relating to data input and output, data storage and retrieval, and the DBMS external, conceptual, and internal models.

- Operate the central processor to handle SDT applications and direct its use among the various SDT users.

- Direct and maintain the physical storage of the SDT-DBMS system programs, the system-specific data bases, and the archival files.

- Provide a batch input/output and graphics output capabilities (in addition to the standard SDT input/output capabilities).

- Assist users and DBDs in utilizing the SDT.

- Provide formal training to SDT users.

- Plan, develop, and implement short-term and long-term SDT improvements.

- Provide data to other Army organizations for related applications (e.g., total force requirements analysis).

- Promote the use of the SDT among Army organizations.

In contrast to the other two SDT user groups, the SDT Management Group is expected to have individuals with sophisticated computer backgrounds -- sophisticated enough to develop and maintain programs for all of the SDT functions. In addition to fluency in standard computer languages, the SDT Management Group will require individuals that understand the SDT-DBMS system.

4.3 PHYSICAL DESCRIPTION OF SDT

Figure 4-1 provides a general description of the SDT physical characteristics. The design outlined in Figure 4-1 should minimize requirements for the purchase of new equipment by participating Army organizations. More details on the hardware associated with the three different user groups is presented in the sections which follow.

4.3.1 Physical Description of Primary User Hardware

The primary users of the SDT will each require a terminal with a keyboard, a CRT with textual capabilities (as a

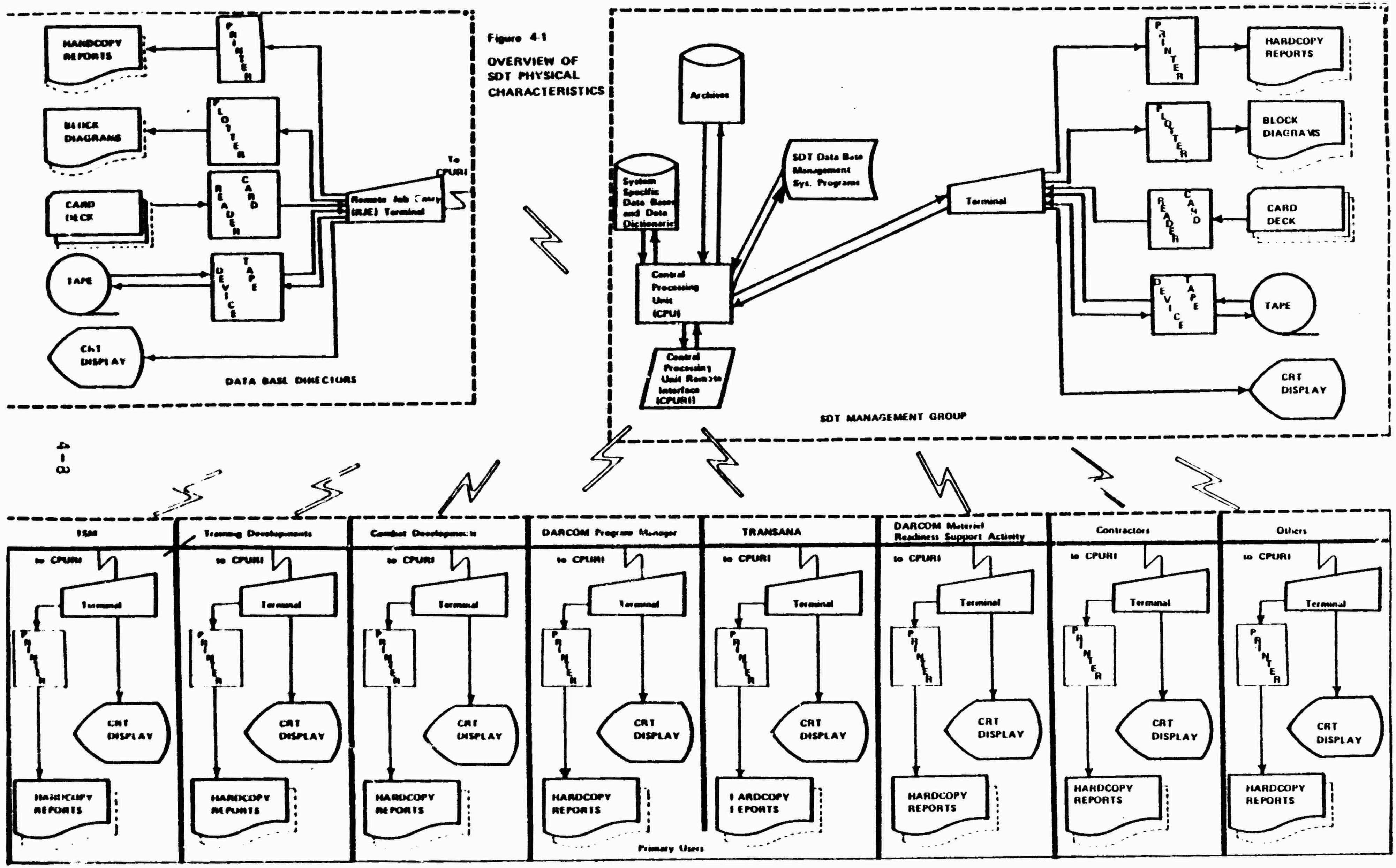


Figure 4-1
OVERVIEW OF SDT PHYSICAL CHARACTERISTICS

minimum), and a printer. This set of equipment will allow the user to input data interactively; to access, update, and modify data; and to receive outputs of SDT information in tabular listings, which will be the standard SDT output format. However, the primary users will not be able to directly input batch data or to produce output in block diagram form[1]. However, they can have these functions performed indirectly through the DBDs or the SDT Management Group. There are two major reasons for not providing the primary users with these additional capabilities. First, the SDT is geared to be utilized with existing equipment and it is likely that a number of primary users will not have access to the equipment (plotters, CRT terminal, card reader, and tape reader) required to provide batch input and graphical output. To provide these users with such equipment would be very expensive and this expense would be likely to diminish SDT utilization. Second, all of the information contained in the block diagrams could be represented in tabular format. Admittedly, this data may be slightly more difficult to understand in this format. However, the analyst need only utilize this data until a diagram is obtained from the DBDs or the SDT Management Group.

It should be noted that some primary users (e.g., training developments, program managers) are likely to have more than one of the terminal set-ups described in Figure 4-1.

---

1 An alternative conceptualization of the primary user equipment set-up which is being considered is to have each primary user have his own intelligent terminal with accompanying graphics capabilities.

4.3.2 DBDs Physical Equipment Description

The DBDs will have the same physical equipment as the primary user (remote terminal, CRT display, and printer) plus additional equipment to provide a batch input capability (card readers and tape readers) and a graphics output display capability for block diagrams (plotter). It should be noted that the DBDs is likely to be one of the primary users (most likely the TSM, PM, or training developments). Also, primary users who have a great demand for a batch input capability or a graphics output capability could add the appropriate hardware without any disruption of the overall system.

4.3.3 SDT Management Group Physical Equipment Description

The SDT will have all of the hardware capabilities of the DBD (terminal, CRT display, printer, and reader, tape reader, and plotter) plus the central processor, and physical storage capabilities for the SDT DBMS programs, system-specific data bases, archival data, and data dictionaries. Thus, the SDT Management Group will require a fairly complete data processing capability.

4.4 OVERVIEW OF SDT PROCESSES

An overview of the general SDT processes is presented in Figure 4-2. The SDT will have the capability of inputting three different types of data: batch input of SDT data sheets, batch input of related acquisition data, and interactive input of SDT data sheets. The SDT DBMS external model will provide the mechanism for reading this input data and translating it into a format suitable for the conceptual

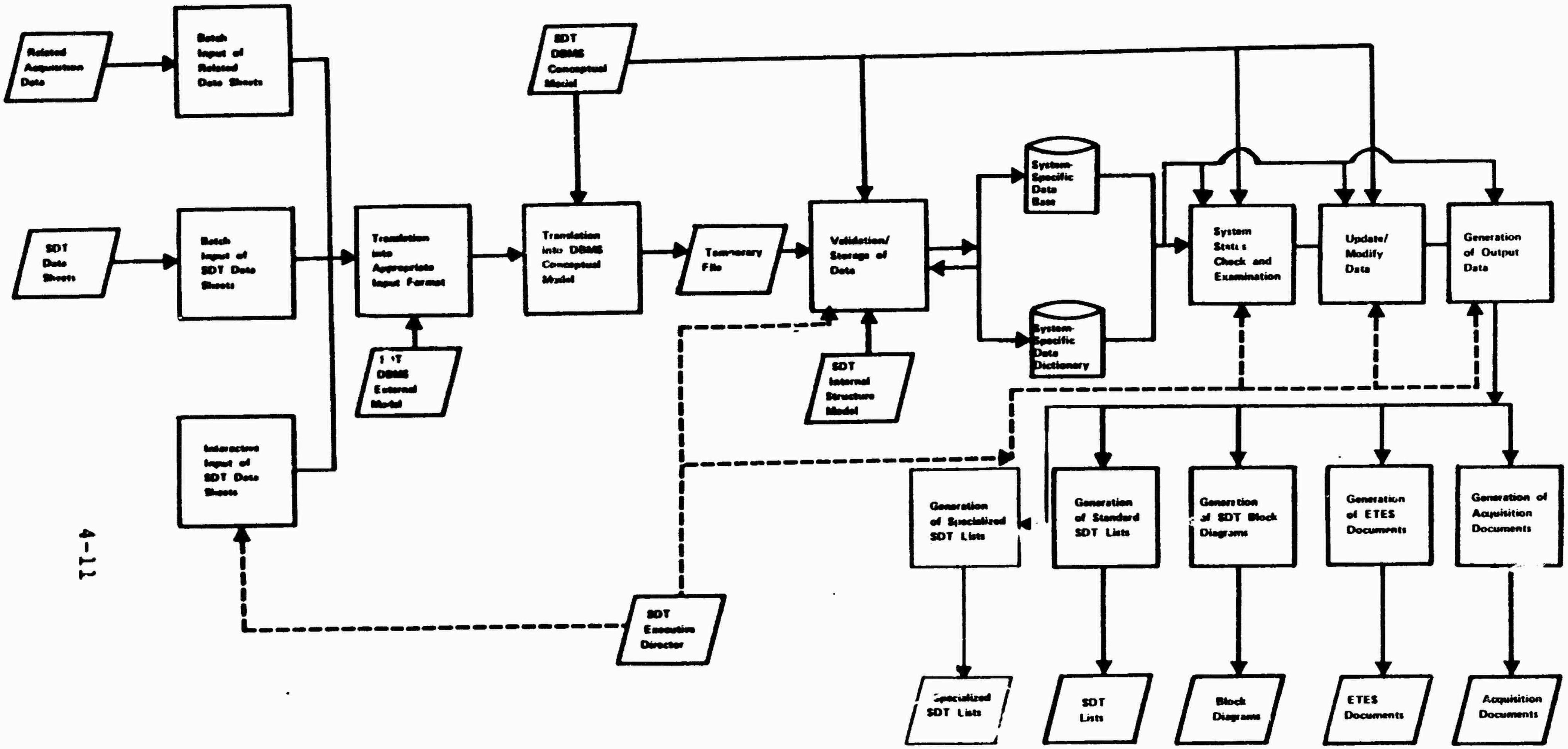


Figure 4-2 OVERVIEW OF SDT PROCESSES

model. Directions for interactive input of data will be provided by the SDT executive director programs. The SDT input data will be translated into a form which matches the conceptual model of the DBMS. Once it has been translated, the data will be evaluated for consistency against data already in the data base and, if consistent, the data will be entered into the system-specific data base. However, this will be done only after the SDT executive director has determined that the user has been cleared to enter that type of data into the data base.

Once in the data base, the data is continuously updated, modified, and expanded. Direction of these changes is provided by the SDT executive director programs. These same programs are used in selecting and generating output data. Five different formats for outputting the data will be available: specialized SDT lists, standard SDT lists, block diagrams, output formatted for input into other ETES procedures, and output formatted to correspond to the format requirements of specific acquisition documents.

4.5 MODES OF OPERATION FOR PRIMARY USERS AND DATA BASE DIRECTORS

This section describes the different modes of operation that will be available to the primary users and Data Base Directors.[2] The SDT will have five primary modes of operation: sign-on/status check, system examination, input, update/modify, and output. System examination will display

---

2 The modes of operation for the SDT Management Group, who will be concerned with software development and maintenance, will, of course, be much more complex.

selected data in the SDT data base. The Input mode is used to input data to the SDT data base. The Update/Modify mode is used to change, replace, or delete existing data in the SDT data base. Output is used to obtain a printed copy of selected data from the SDT data base.

The user entering the system sign-on/status check mode, will be able to select the mode of operation he would like to begin with. As he progresses through transactions, he may proceed to different modes in an alternating fashion. The classification of transactions into modes facilitates the sequencing of the transaction frames which will lead the uninitiated user through the SDT. More details on these different modes of operation are provided in the sections which follow.

4.5.1 Sign-On/System-Status Check

This is the first mode of operation for every user. During this mode of operation, the user is provided with (1) references for obtaining a detailed description of the SDT, (2) a general overview of the items that are currently in the data base for the system in which he is interested, and (3) a general description of the updates, modifications, and additions to the data base that have occurred since he last utilized it.

After the sign-in/system status check, the user will be provided with an opportunity to select which of the remaining four modes of operation he would like to enter first.

4.5.2 System Examination

During this mode of operation, the user can examine data which is currently in the data base. In conducting this examination, the user will have the option of either selecting from a predetermined set of information or being led through the data in top-down hierarchical fashion to examine specific elements within a class of information. In the latter approach, he will be presented with a menu of the items at each level in the hierarchy and he will be asked to select which item in the menu he would like to examine in more detail in the next frame. When the user is finished going down a branch as far as he would like to go, he can start over again at the top of the hierarchy.

4.5.3 Input Mode

This mode of operation refers to interactive input only. Thus, in the first frame, the user will be told that he can only input data interactively and that if he wants to enter batch data he must do so by sending his data to the DBDs or SDT Management Group who will have batch capabilities. To input data the user will be presented with a list of the possible input formats he may use and he will be asked to select a format. The selected format will then appear on the screen and he will begin to fill in the necessary data (as many of the data elements as possible will be filled in automatically by the SDT). When he has completed one format, the next format will appear on the screen and he will fill that in, etc. When he has completed inputting the information, the SDT will check his information for consistency with existing data and the SDT conceptual model.

The user will then be asked to correct his errors.[3] When all errors have been corrected, the user will be presented with a summary of the input data he has developed and he will be asked if he wants to place these information elements into permanent storage in the DBMS. If he does, the system will then determine if the user has been cleared to store data of that type. If he does have clearance, this information is stored. If he doesn't have this capability, the user is recommended to go to the output mode to receive a hard copy of his data and to call the DBD to clarify his role.

The input mode is used to add information to the DBMS for specific entities. There will be a separate data input sheet for each entity and different types of input sheets for the different classes of entities. If the user wants to change information, he is directed to enter the update/ modify mode.

4.5.4 Update/Modify Mode

During this mode of operation, the user may (1) eliminate a complete entity and its corresponding attributes (i.e., a single input worksheet), (2) eliminate an entire class of entities, (3) eliminate an attribute value on a specific datasheet, and (4) replace an attribute value with another.

Again the method for making these updates and modifications will be based on a top-down hierarchical approach in which

---

[3] If inconsistencies still existed between data in the data base and input data, the user would be asked to call the DBD for instruction.

the user will use menu-selection to select items and work his way down a hierarchy until he has reached a specific entity. At that point, the worksheet associated with that entity will be placed on the screen and the user can make the appropriate modifications and updates.

4.5.5 Output Mode

In this mode the user selects what data he would like to have outputted and the type of output he would like to obtain. The user will have two options for selecting the data elements he wishes to output. First, he can select a report type from among a standardized set of common output report formats (this will be the quickest way to obtain output data). Second, he can elect to output data he has worked with or developed in the previous modes. More specifically, he can have outputted all the data he examined during the examination mode, all the data he input during the input mode, and/or all of the data sheets associated with the data elements he modified or updated during the modify/update mode.

Five different types of output formats will be available: (1) specialized SDT output that describe the subsets of data examined, input, or modified by the user; (2) standard SDT outputs; (3) block diagrams (hierarchical activity flow and information flow); (4) specialized output formats tailored to produce output in a form which is congruent with the input data requirements of other ETES technologies (e.g., the simulation models); and (5) specialized output formats tailored to produce output in a format which is congruent with the input data requirements of other analyses in the acquisition process (e.g., the LSAR).

Primary users will not be able to directly obtain block diagrams at their terminal sites since they may not have graphics capabilities. However, they can receive the block diagrams from the DBD or the SDT Management Group who will have a graphics capability for producing block diagrams.

4.6 OVERVIEW OF SDT OPERATION

The SDT will utilize three general types of human-computer dialogue formats in transactions with primary users and the DBDs[4]: menu selection, form-filling, and question-and-answer. Form-filling will be used as the primary mechanism for inputting, updating, and modifying data. Menu selection will be used as the means for systematically searching through the SDT data to select data for examination, update/modification, and output. Question-and-answer will only be used in a small number of situations where there are questions involving a fairly well-defined answer space (e.g., how many input sheets do you want to enter). A more detailed description of the operation of the SDT is presented in the section which follows.

4.7 EXAMPLE INTERACTIONS

To provide the reader with a more concrete picture of the operation of the SDT, this section provides a detailed description of the types of interactions that are likely to occur under each of the five modes of operation (sign-on/status check; system examination; input; update/modify;

---

4 The SDT Management Group is expected to be able to use more sophisticated dialogue.

and output). The interactions for each mode of operation are presented on a frame-by-frame basis where each frame roughly corresponds to the information that would appear on the CRT screen at a single point in time. The example frames presented in this section are not meant to be a verbatim description of the actual frames which will be used for SDT. The examples simply demonstrate the type of information that can be expected to be incorporated in the different interaction modes. The frames are frequently interspersed with comments which summarize or further explain the types of information which can be expected to be placed on each sheet. Figure 4-3 is a schematic representation of the relationships among these frames. It represents the structure of the automated SDT.

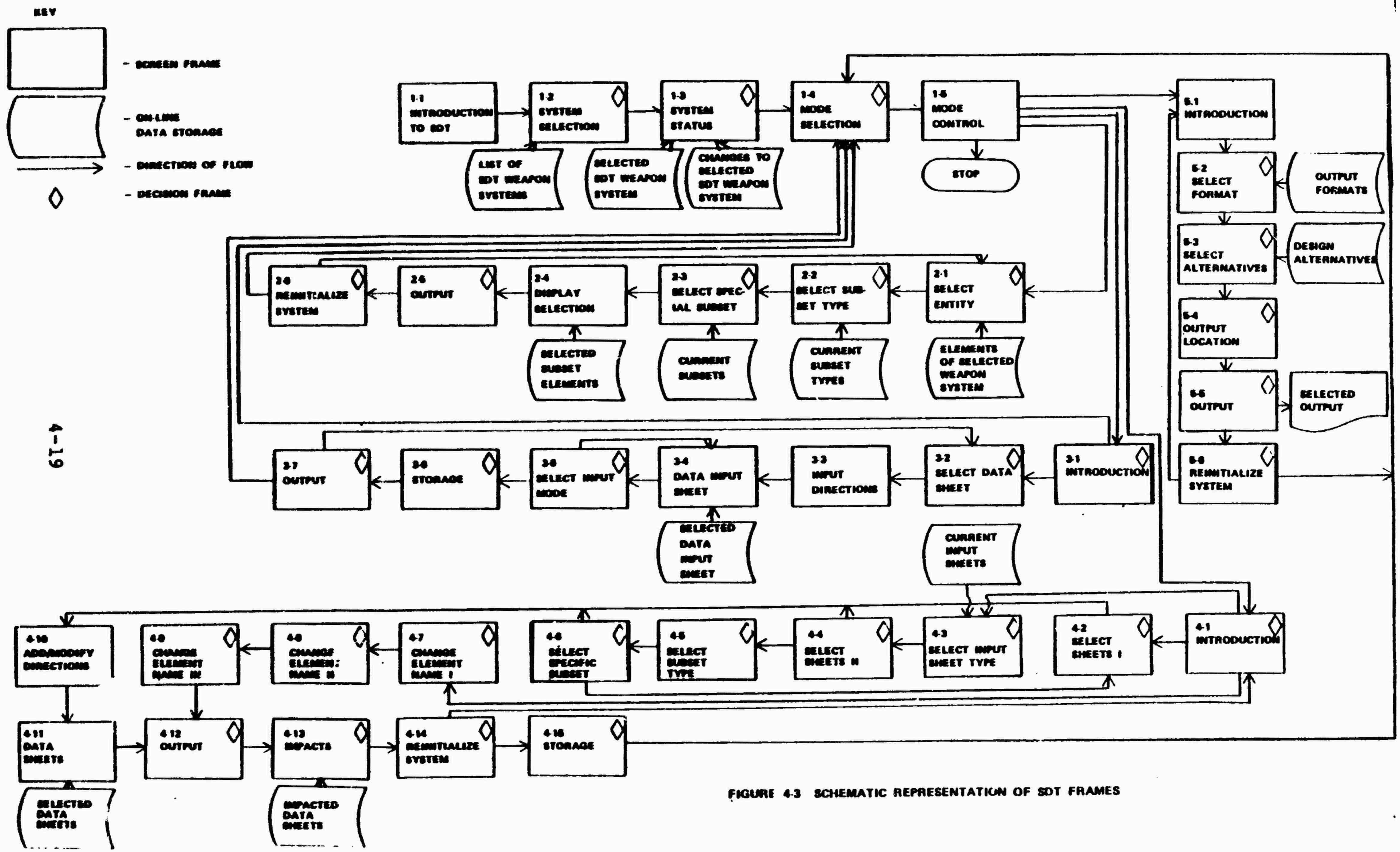


FIGURE 4-3 SCHEMATIC REPRESENTATION OF SDT FRAMES

Mode: Sign-On/Status Check Frame 1-1

Title: Introduction to SDT

Welcome to the System Description Technology (SDT). This SDT is an automated tool for describing actual and projected elements including functional requirements, design concepts, tasks, skills, training program elements and their associated resources.

You should not attempt to use the SDT until you have read the SDT users manual. You may obtain a manual from the SDT Management Group which is located at (autovon) or the data base director for your system ( ).

If you have any questions about the use of the SDT at any time, please feel free to call the SDT Management Group or your data base director.

(User would continue on to frame 1-2)

Mode: Sign-On/Status Check | Frame 1-2

Title: System Selection

Listed below are the systems which currently have an SDT data base. Please type in the number of the system you would like to work with.

1. xxx	6. xxx
2. xxx	7. xxx
3. xxx	8. xxx
4. xxx	9. xxx
5. xxx	10. xxx

Number________________

(User would continue on to frame 1-3)

Mode: Sign-On/Status Check Frame 1-3

Title: System Status

Listed below are the systems which are currently in the data base for your system and the elements which have been added or modified since your last interaction with the SDT.

	System	Person Making Last Transaction	Date of Last Transaction	Items Added Modified Since Your Last Transaction
1.				
2.				
3.				
4.				
5.				
6.				
7.				
8.				
9.				
10.				

(User would continue on to frame 1-4)

Mode: Sign-On/Status Check   Frame 1-4

Title: Mode Selection

The SDT has four modes of operation:

1. System-Examination: This mode is used to examine data which is currently in the data base, etc. (see page _)

2. Input: This mode is used to input data, etc. (see page _).

3. Update/Modify: This mode is used to eliminate or modify data already in the data base, to replace obsolete or incorrect data values, etc. (see page _).

4. Output: This mode is used to obtain a hard copy output of elements in the SDT (see page _).

5. Stop: I am finished using the SDT.

Please type in the number of the mode of operation you would like to use first. Unless you have something fairly specific in mind, you probably want to start with a system examination (Mode 1).

Number________________

(User would go to frame 1-5)

Mode: Sign-On Status Check Frame 1-5

Title: Mode Control

Before entering the mode you have chosen, it is important to point out that you can use the following commands to obtain immediate withdrawal from a mode or a particular interaction within a mode:

BACK: By typing in this command, you can immediately leave the particular set of transations in which you are involved and return to the beginning of that mode.

WAYBACK: By typing in this command, you can return to the mode selection option and select a new mode.

The above commands are only necessary when you want to interrupt the normal flow of the SDT transactions. If you do not interrupt, you will automatically be given a chance to enter another mode when you have completed your transactions in the current mode you have selected.

(User would go to the first frame of the mode he selected)

Mode: System Examination Frame 2-1

Title: Selection of Entity

Listed below are system elements currently in the data base for your system. Please type in the number of the system element you would like to examine first.

System Element	Person Making Last Transaction	Date of Last Transaction	Items Added or Modified Since your Last Transaction	Associated Design Alternative(s)
1. Functional Requirements				
2.				
3.				
4.				
5. Tasks				

(Assume that user types in No. 5: Tasks)

(User would go to frame 2-2)

Mode: System Examination Frame 2-2

Title: Selection of Subset Type

Tasks may be grouped in several different ways. Listed below are the ways in which tasks may be grouped for examination in the SDT. Please type in the number of the way in which you would like the tasks grouped for examination.

1. Tasks by general task type (operator tasks, maintenance tasks, support tasks)
2. Task by MOS and duty position
3.
4. Tasks by equipment
5.
6. Tasks by equipment and task type
7.

(Assume the user types in "6" indicating that he wants tasks by equipment and by task type.)

Listed below are the current design alternatives. Please type in the number of the design alternative for which you would like to examine the above information.

1	5
2	6
3	7
4	

Number________________

(User would go to frame 2-3)

Mode: System Examination Frame 2-3

Title: Selection of Special Subset

You have chosen to examine tasks by equipment and task type. Listed below are the hardware subsystems and the task types which are currently in the data base for your system. Please type in the numbers of the equipment and task types you would like to examine in the spaces provided below.

Equipment	Task Types
1. All subsystems	1. All tasks
2. Fuel	2. Operator task
3. Turret	3. Maintainer task
4. Engine	4. Support tasks
5.	
6.	
7.	

Equipment Numbers_______________

Task Type Numbers_______________

(assume user types in 1 for equipment and 3 for task types)

(User would go on to frame 2-4)

Mode: System Examination Frame 2-4

Title: Display Selection

Listed below are the maintenance tasks for all of the equipment subsystems.

(User would go on to frame 2-5)

Mode: System Examination   Frame: 2-5

Title: Output

Do you want a printed output of the information you have just examined?

1. yes   2. no

The computer will remember that you want an output of this infomation. You may continue to examine additional system elements. When you have completed your examinations, you can enter the output mode and have this and any other system information you have examined printed out in a hard copy. Or you can enter the output mode and obtain a printed output immediately.

(User would continue on to frame 2-6)

Mode: System Examination Frame 2-6

Title: Reinitialization of System

Do you want to continue examining system elements (that is, to stay in the system examination mode).

1. yes 2. no

(If the user typed in "1" (yes), he would be sent back to frame 2-1 to begin the examination process over. If the user typed in "2" (no) indicating that he wanted to go into a new mode of operation, he will be sent back to frame 1-4 to select another mode of operation.)

Mode: <u>Input</u> Frame 3-1

Title: <u>Introduction</u>

The steps which follow will allow you to input data interactively (that is directly on the display screen). If you want to enter data in batch mode (that is, you want to fill out data sheets manually and submit these completed data sheets or you want to enter data which is already on computer cards or tape), you must contact the data base director for your system or the SDT data base management group and make arrangements for them to enter this data into the data base for you.

Remember, this mode of operation can only be used to add an entire data sheet. If you wish to add <u>an item</u> to a data sheet, you should enter the Update/Modify mode.

Do you want to continue in this mode?

1. yes 2. no

(If he types "1" (yes), he will continue on to the next frame, 3-2. If user types "2" (no), he is sent back to mode-selection frame, 1-4.)

Mode: System Input | Frame 3-2

Title: Selection of Data Sheet

Listed below are the different types of data input sheets which are used in the SDT. Please type in the number of the type of data sheet you would like input first.

	Data Input Sheet	Description
1.		
2.		
3.		
4.		
5.		
6.		
7.		
8.	Maintenance task lists	
9.		

(assume the user typed in "8", indicating he wanted to input maintenance task lists).

How many of these data sheets do you want to input?
(User types in number)

(User would go on to frame 3-3)

Mode: Input　　　　　　　　　Frame 3-3

Title: Input Directions

Before inputting data, it is important that the following guidelines be reviewed (see the User manual for a more detailed description).

1. If you make a mistake in inputting data on a sheet and you have not left the line containing the incorrect information, you can correct the error by backspacing and typing in the line again.
   If you have left the line with the incorrect information, you cannot correct the data in the input mode. You must continue and go into the update/modify mode later to correct your errors. You can, however, eliminate the entire input sheet by typing in -- at any time.

2. You do not have to fill out all of the items in the data input sheet. Fill out as many items as you can. You will have an opportunity to add to the data sheets later.

3. The data input sheets will often have a listing of possible answers and/or a range of legitimate values for many of the data items.

4. To obtain additional guidance in filling out the data sheets, particularly in using output from other parts of the data base to help you fill out information, please see your manual, section _.

(User would continue on to frame 3-4)

Mode: Input Frame 3-4

Title: Data Input Sheets

(The user would now be presented with a formatted data input sheet which he would fill in. When he reached the end, another sheet would appear and he would type in the information for the next sheet, etc. User would eventually go on to Frame 3-5.)

Mode: Input   Frame 3-5

Title: Select Input Mode

You have just completed filling out your xxx sheet. Please type in the number of the action you would like to take next.

1. Input more data input sheets of this type

2. Input another type of data input sheet

3. Go into another mode of operation.

(If the user typed in "1" he would be asked how many more sheets he wanted to type in and then he would be presented with the sheets. If he typed in "2", he would go to frame 3-6. If he typed in "3", he will also be sent to frame 3-6.)

Mode: Input Frame 3-6

Title: Storage

Do you wish to store the data items you have input?

1. yes 2. no

(If user answers yes, the system will check to see if the user has been cleared to store data of that type. If he has not been cleared, a message will appear on the screen telling the user that he has not been cleared to modify and store data of that type and he will be asked to contact his data base director or the SDT Management Group. If he is cleared, and his data has been stored, he will be told he has been cleared and sent to frame 3-7.)

Mode: Input   Frame 3-7

Title: Output

Do you want a printed output of the input data sheets you have just examined?

1. yes   2. no

Do you want to input more data?

1. yes   2. no

(After typing "1" or "2" to the second question, the user would be sent either to frame 3-2 if he indicated he wanted to input more data or to frame 1-4 if he indicated he wanted to enter another mode. (Assume user typed in "1" (yes) for both questions.)

The computer will remember that you want an output of this information. You may continue to input additional system elements. When you have completed inputting data, you can enter the output mode and have this and any other information you have input printed out in hard copy.

Mode: Update/Modify Frame 4-1

Title: Introduction

This mode will allow you to (1) replace the current value of an item in an existing input sheet with another value, (2) eliminate an entire input sheet, (3) eliminate data items in a data sheet, (4) eliminate a system element and all of its associated data (e.g., eliminate a subsystem and all data collected with it), and (5) change the name of a particular data item.

Note: If you want to add an entire input sheet you must enter the input mode to do so.

Please type in the number of the modification you want to make.

1. Replace the current value of an item in an existing input sheet with another value
2. Eliminate an entire input sheet
3. Eliminate data items in a data sheet
4. Eliminate a system element and all of its associated data
5. Change the name of a particular data item

Number_______________

(If the user types in "1", "2", or "3", he is sent to frame 4-2; if he types in "4" he is sent to frame 4-3; if he types in "5", he is sent to frame 4-7).

(User would go on to frame 4-2)

Mode: Update/Modify Frame 4-2

Title: Selection of Sheets I

Do you have a specific sheet you would like to eliminate, update, or modify?

1. yes 2. no

If "yes" enter the numbers of those sheets

Numbers

Numbers

etc.

(If user enters "yes", the identified sheets would then appear on the screen-as in frame 4-10 after the user has answered the following questions).

Do you want to eliminate any input sheets?

1. yes 2. no

(If he answered "yes", he is asked the question which follows. If he answers "no", the specific data sheets he has chosen to update and modify will begin appearing on the screen as in frame 4-10).

Enter the numbers of the sheets you want to eliminate

Number

Number

Number

etc.

(After eliminating sheets or updating or modifying sheets, the user will be sent to frame 4-12)

Mode: Update/Modify Frame 4-3

Title: Selection of Input Sheet Type

Listed below are the different types of input sheets currently in the system. Please type in the number of the type of sheet you would like to update/modify/eliminate first.

	Data Input Sheet	Description	Design Alternative	Currently in System (X)
1.				
2.				
3.				
4.				
5.				
6.	Tasks Number			

(Assume the user types in "6")

(User would go on to frame 4-4)

Mode: Update/Modify   Frame 4-4

Title: Selection of Sheets II

Do you want to update, modify, or eliminate all the task data sheets?

1. yes   2. no

(If he answers "yes", then the input data sheets of that type will start appearing on the screen one right after another as in frame 4-11. If he answers "no", he is sent to frame 4-5, which will direct him to search through the data and select the subset of input data sheets within the more general type he would like to update/modify/eliminate. After all of the relevant data sheets have been modified or updated, the user will be sent to frame 4-12.)

Mode: <u>Update/Modify</u> Frame 4-5

Title: <u>Selection of Subset Type (example)</u>

Listed below are the different ways in which task input data sheets may be grouped. Please type in the grouping related to the subset you would like to modify/update (for example)

1. Tasks by general task type (operational tasks, maintenance tasks, support tasks)
2. Tasks by MOS and duty position
3.
4. Tasks by Equipment
5.
6. Tasks by Equipment and Task Type

(Assume user types in "6" indicating that he wants to group input data sheets by equipment and task type).

Listed below are the current design alternatives. Please type in the number of the design alternatives for which you would like to examine the above information:

1. all	5.
2.	6.
3.	7.
4.	8.

Number_______________

(After typing in the number, the user would be sent to frame 4-6).

Mode: Update/Modify Frame 4-6

Title: Selection of Specific Subset

You have chosen to group tasks by equipment and task type. Listed below are the hardware subsystems and the task types which are currently in the data base for your system. Please type in the numbers of the equipment and task types you would like to update/modify/eliminate in the spaces provided below.

Equipment	Task Types
1. All Subsystems	1. All tasks
2. Fuel	2. Operator tasks
3. Turret	3. Maintainer tasks
4. Engine	4. Support tasks
5.	
6.	

Equipment Numbers______________

Task type Numbers______________

Do you want to eliminate all of the data input sheets in this subset?

1. yes 2. no

(If he answers "yes", they are eliminated and he is sent back to frame 4-2. If he answers "no", it is assumed he wants to update/modify the individual sheets in the subset and the sheets will start appearing on the screen as in 4-10. However, as each input data sheet appears on screen, he will be given an opportunity to eliminate it. After modifying/ updating the individual input data sheets, he is sent to frame 4-12.)

Mode: Update/Modify Frame 4-7

Title: Changing the Name Of A System Element - Level 1

If you want to change the name of a particular data item, you must first identify the class of data items of which it is a member. Listed below are the different types of data items which are in the SDT. Please type in the number of the type of data item whose name you would like to change.

1. Functions
2. System Design Concepts
3. Tasks
4.

Listed below are the current design alternatives. Please type in the number of the design alternatives for which you would like to modify the name

1	4
2	5
3	6

Number________________

(After typing in the number, the user would go on to frame 4-8.)

Mode: Update/Modify Frame 4-8

Title: Changing the Name of the System Elements - Level 2

Please type in the ID number of the item(s) whose names you would like to change. These ID numbers can be obtained by examining the input data sheets associated with the system elements.

Numbers

1

2

3

4

5

6

7

8

(After filling in the numbers, the user would go on to the next frame 4-9.)

Mode: Update/Modify Frame 4-9

Title: Changing the Name of System Elements - Level 3

Please type in the new names you wish to use for the current system elements listed below.

ID Number	Current Name	New Name
xxxxx	xxxxxxx	
etc.		

(After completing these changes, the user would be sent to frame 4-12)

Mode: Update/Modify   Frame 4-10

Title: Directions for Adding/Modifying Data Sheets

You can add or modify items in the data sheets which appear on the screen by (1) skipping to the line you want to modify using the line advance key on your keyboard and (2) typing in (if adding a new item) or typing over (if replacing an existing item) the appropriate information on that line.

In addition, the following directions can aid you in inputting data:

1. If you make a mistake in inputting data on a sheet and you have not left the line containing the incorrect information, you can correct the error by backspacing and typing in the line again.

   If you have left the line with the incorrect information you cannot correct the data in the input mode. You must continue and go into the update/modify mode to later correct your errors. You can, however, eliminate the entire input sheet by typing in at any time.

2. You do not have to fill out all of the items in the data input sheet. Fill out as many items as you can. You will have an opportunity to add to the data sheets later.

3. The data input sheets will have a listing of possible answers and/or a range of legitimate values for many of the data items.

4. To obtain additional guidance in filling out the data sheets, particularly in using output from other parts of the data base to help you fill out information, please see your users manual, section___.

(User would be sent to frame 4-11)

Mode: Update/Modify Frame 4-11

Title: Data Sheets

(The user would now be presented with the data sheets which he had selected to update or modify. When he completed one sheet, another sheet would appear. After the final one, he would go on to frame 4-12).

Mode: Update/Modify   Frame 4-12

Title: Output

Do you want a printed output of the input data sheets you have just modified or updated?

1. yes   2. no

(Assume user answered "yes".)

The computer will remember that you want an output of this information. You may continue to input additional system elements. When you have completed inputting data, you can enter the output mode and have this and any other information you have input printed out in hard copy.

(After answering this question, the user will be sent to frame 4-13)

Mode: Update/Modify   Frame 4-13

Title: Impacts

Do you wish to see what other data input sheets are impacted by the changes you have made? It is highly recommended that you consider these impacts.

1. yes   2. no

(If user answers "no", he is sent to the next frame. If he answers "yes", the following information appears on the screen.)

Listed below are the data input sheets which are impacted by the changes you have made.

Data Sheet Type	Data Number	Elements Impacted	Sheet Number of Modified Sheet Producing Change

(After viewing this information, the user is sent to the next frame, 4-14)

Mode: Update/Modify Frame 4-14

Title: Reinitialization of System

Do you want to continue in the update/modify mode?

1. yes 2. no

(If user answers "yes" he is sent back to frame 4-1. If he answers "no" he is sent to frame 4-15.)

Mode: <u>Update/Modify</u> Frame 4-15

Title: <u>Storage</u>

Do you wish to store the data items you have changed?

1. yes 2. no

(If user answers "yes", the system will check to see if user has been cleared to store data of that type. If he has not, a message will appear on the screen telling the user that he has not been cleared to modify and store data of that type and he will be asked to contact the data base director or the SDT management group. If he answers "no", he will be sent to frame 1-4.)

Mode: Output Frame 5-1

Title: Introduction

The SDT can provide five major types of output:

1. Specialized SDT output - These are specialized output reports that you may have constructed in examining the system (Mode 1), inputting data (Mode 3), or updating/modifying the system (Mode 4). These specialized reports allow you to examine the specific subsets of information you chose to examine, the input data sheets you have just entered into the system, or the data sheets you have just updated or modified. (If you have indicated you wanted output in previous modes, you must select this option to obtain that output.)

2. Standard SDT Outputs - The SDT has a series of standard reports for representing the types of data that users are most likely to request. If you are in doubt as to what type of output you want, it is recommended that you select this option.

3. SDT Block Diagrams - The SDT has the capability of printing out block diagrams for selected types of system relationships. The type of block diagrams which are available are (1) hierarchical block diagrams for representing hierarchical relationships among system elements, (2) activity flow diagrams for representing sequential relationships between system elements or functions, and (3) information flow diagrams for representing the flow of inputs and outputs among system elements.

Mode: Output Frame 5-1 (cont.)

Title: Introduction

You may not have the capability for obtaining block diagrams on your system. If you select this option and do not have a block diagram capability, your output will be sent to either the DBD or SDT Management Group, who will in turn send it to you at a later date.

4. ETES Formatted Output - The SDT has the ability to produce output data in formats which are congruent with the input requirements of other ETES tools.

5. Acquisition Document Formatted Output - The SDT has the ability to output data in formats which are congruent with selected Army acquisition documents.

(User would be sent to frame 5-2)

Mode: Output   Frame 5-2

Title: Selection of Output Formats

Listed below are the output reports that are available. Type in the numbers of the output reports you want.

Specialized Output Data	Standard SDT Data
1.1 Examined Duty	2.1 Tasks by MOS, Duty Position
1.2 Input Data	2.2
1.3 Modified Data	2.3
1.4	2.4

ETES Formatted Outputs	Acquisition Doc Formatted Outputs
4.1	5.1
4.2	5.2
4.3	5.3
	5.4

Block Diagrams (please select a number and a letter for each diagram you want)

3.1 Hierarchical	A. Functions
3.2 Activity Flow	B. Design Concepts
3.3 Information	C. Tasks
	D.

Numbers

Selected_______________

(user would be sent to frame 5-3)

Mode: Output Frame 5-3

Title: Selection of Alternatives

Listed below are the major design alternatives for your system.

1.

2.

3.

4.

Listed below are the outputs you have selected. For each output, please type in the number associated with the design alternatives for which you would like to see the designated ouput.

Output	ID Number of Design Alternative(s) You Would Like To See Output For

(User would be sent to frame 5-4.)

Mode: Output Frame 5-4

Title: Output Location

Unless otherwise specified, the SDT will print all data on your printer except for block diagrams, which you must have outputted at the DBD or SDT Management Group facilities.

In the spaces provided provided below, please type in the number of the additional location (DBD, SDT Management Group, or other) where you would like to have your reports outputted. If you are happy having the reports outputted on your printer, you can leave the space blank. However, if you are requesting block diagram output, you must specify where you would like the diagrams output. The codes for location sites are:

1. Your site
2. DBD for your system
3. SDT Management Group
4. Other (must be prearranged with the SDT Management Group before attempting)

In addition to an alternative location, you may also want to specify an alternative medium for outputting your data such as tapes or discs. Also in the spaces provided below please write down the number of the alternative medium on which you would like to have your output listed using the following code:

1. Tape
2. Disc
3. Card
4. Other (must be prearranged with SDT Management Group)

Mode: Output   Frame 5-4 (cont.)

Title: Output Location

If you are happy with your printer as an output medium you do not have to type in anything. Do not attempt to use an output medium with a site unless you are absolutely certain that the site has the capabilities for that medium and has been warned of your use.

Output	Alternatives	Location Number	Medium no.

(User would continue onto frame 5-5.)

Mode: Output Frame 5-5

Title: Output

(At this point, the user would receive the printed output he requested. The block diagrams would be output at the DBD or SDT management group's facilities. The user would have the capability of interrupting the output at any time by pressing a --,in which case frame 5-6 would appear on the screen. If he did not interrupt, 5-6 would appear when he finished.)

Mode: <u>Output</u> Frame 5-6

Title: <u>Reinitialization of System</u>

Do you want to continue in the output mode?

1. yes 2. no

(If he enters "yes" (1), he is sent to frame 5-1. If he enters "no" (2), he is sent to frame 1-4.)

4.8 INITIAL VERSION OF SDT

The previous subsections outlined the features of the complete SDT. This system is designed to provide an optimal tool for meeting all of the SDT requirements. It is unlikely that the optimal SDT can be fully developed within the confines of the present study. Hence, it is necessary to identify an initial SDT which can be used to demonstrate the SDT capabilities and test out key concepts before the optimal SDT is constructed.

More details on the characteristics of the initial SDT are presented in the subsections which follow.

4.8.1 Characteristics of the Initial SDT

The initial SDT will differ from the final SDT in four major areas: expected users, input capabilities, output capabilities, and the range of system items to be described.

4.8.2 Expected Users

For the development of the initial SDT, the contractor (DRC) will assume the role of both data base director and SDT Management Group. Thus, the initial SDT, unlike the optimal version, will be designed for two, rather than three, user groups.

4.8.3 Input Capabilities of SDT

The final SDT will have the capability of inputting three different types of data: batch input of SDT data sheets, interactive input of SDT data sheets, and batch input of related data. However, the initial SDT will only have the capability for interactive input of SDT data sheets.

4.8.4 - Output Capabilities of SDT

The final SDT will have the capability of producing five different types of output: (1) Specialized SDT Output for describing subsets of data examined, inputted, and modified/updated by the user, (2) Standard SDT data lists -- a series of reports listing the types of data which users are most likely to request, (3) SDT Block Diagrams, (4) ETES Formatted Output, and (5) Acquisition Document Formatted Output. The initial SDT will only provide a capability for producing (1) the Specialized SDT outputs and (2) Standard SDT data lists.

In addition, the initial SDT will not have the capability of outputting data to tape, disc, or alternative media. It will only have the capability of producing standard printed output.

4.8.5 System Elements Described in SDT

The initial SDT will attempt to describe all of the SDT system elements specified in Section 2. However, it will not be possible to describe all of the elements in the initial version of the SDT. The schedule for including the various system elements in the SDT will follow the priorities outlined in Section 2. It may not be possible to complete this schedule within the confines of the present study. In that case, items labelled "priority 3" may not be included in the initial SDT.

4.8.6 Other Characteristics of the Initial SDT

Because it is the "first cut" at the SDT, the initial SDT is also likely to have other limitations. First, it is likely that the SDT user interface will not be as smooth, efficient, and transparent as the ultimate SDT interface. Second, the documentation for using the SDT will be in draft form only and is unlikely to be as comprehensive as the documentation for the prototype. This draft will be constantly updated and refined as the study progresses.

REFERENCES

AIRMICS R&D Bulletin No. 1: Management Decision Support Systems, April 1980.

Alford, M. Software Requirements Engineering Methodology (SREM) at the age of two. IEEE Transactions, 1978, pp. 332-338.

Amkreutz, J. Cybernetic model of the design process. Computer Aided Design, 1976, 8(3), 182-192.

Army Training and Evaluation System. Task 3 Report. Softech Contract No. 1026, June 1977.

Atwood, M., and Jeffries, R. Studies in plan construction II: Novice design behavior, (TR-SAI-80-154-DEN).

Atwood, M., Jeffries, R., Turner, A., and Polson, P. The processes involved in designing software. Science Applications Inc. Technical Report. August 1980.

Bagge, C., and Rothman, D. A first-order methodology for calculating probability of mission success. Defense Nuclear Agency DNA 4843F, January 1979. (ADA088910)

Balzer, R., Goldman, N., and Wile, D. Informality in Program Specification (ISI/RR-77-59). Information Sciences Institute. April 1977.

Barina, H., Cobey, W., Rosenbaum, J., and White, S. Automated software design. IEEE Transactions, 1979, pp. 384-391.

Baron, S., and Levison, W. The optimal control model: Status and future, Proceedings of the International Conference on Cybernetics and Society, 1980, pp. 90-100.

Berkowitz, M., and O'Neil, H. An Annotated Bibliography for Instructional System Development, ARI Technical Report 426, August 1979.

Bonczek, R., Holsapple, C., and Whunston, A. Future directions for decision support. Purdue University Technical Report. June 1980. (AD-A087355)

Bonder, S. A review of Army force modernization and associated manpower, personnel, and training processes. ARI Research Note 81-6, January 1981.

Boyd, D., and Pizzarello, A. Introduction to the WELLMADE Design Methodology. IEEE Transactions on Software Engineering, Vol. SE-4, No. 4, July 1978.

Boydstan, L., Teichroew, D., Spewak, S., Yaamamoto, Y., and Stamer, G. Computer aided modeling of information systems. Paper presented at IEEE Computer Society Conference, 1980.

Brant, R., and Taffs, D. A user-oriented approach to control languages. Software-Practice and Experience, 1976, Vol. 6, pp. 93-108.

Brooks, R., and Gamet, M. Modern programming practices: Implications for human factors research. ARI Research Note 80-18, July 1979.

Brown, G. Top down design using HARTRAN. Proceedings of the IEEE, 1980, 1154-1157.

Bulfer, R., Goldman, A., and Wile, D. Informality in program specification. Information Sciences Institute Research Report. 77-59. April 1977. (AD A041667)

Burns, J. From flow diagrams to simulation model. Proceedings of the International Conference on Cybernetics and Society, 1980, pp. 7-12.

Cordes, R. Software-user interface evaluation: Methodology and tools. Proceedings of Table Human Factors Society - 24 Annual Meeting - 1980, pp. 395-399.

Cory, B., Johnson, C., Korotkin, A., and Stephenson, R. Duty modules: An approach to the identification and classification of personnel resources and requirements. ARI Technical Paper 367, June 1979.

Cheng, L. URL-URA - An Installation Guide. ESD-TR-75-36. Air Force Electronics Systems Division, January 1976.

Crolotte, A. Selecting a decision aid. Proceedings of the International Conference on Cybernetics and Society, 1980, pp. 1067-1974.

David, J., Price, J. Successful communication in full scale engineering development statements of work. Air Force Institute of Technology, August 1980. (AD A087497)

Demonstration of Input/Output Requirements Language (IORL) for AIRMICS: Final Report. September 1980.

Detrich, F. Let's return to the fundamentals of the acquisition process. AIAA Aircraft Systems and Technology Meeting, August 4-6, 1980.

Dick, R., and Koehler, E. Engineer's guide to the use of human resources in electronic system design: An evaluation. NPRDC Special Report 81-3. November 1980.

Dieterly, D. Problem solving and decisionmaking: An integration. NASA Technical Memorandum 81191, April 1980.

Donnell, M, Adelman, L., and Patterson, J. Development of a computerized training requirements and cost evaluation system for the U.S. Marine Corps. Defense Advanced Research Projects Agency, Technical Report 80-5-315-3, November 1980.

Dzida, W., Herda, S., and Itzfeldt, W. User-Perceived Quality of Interactive Systems. IEEE Transactions on Software Engineering, Vol. SE-4, July 1978.

Erbe, E., Hastwig, R., Kehman, H., Mueller, G., and Schauer, V. Integrated data analysis and management for the problem solving environment. Information Systems, 1980 (5), 273-285.

Evershed, D., and Rippon, G. High level languages for low level users. Computer Journal, 1970, 14 (1), pp 87-90.

Feigenbaum, E. Knowledge engineering: The applied side of artificial intelligence. Department of Computer Science, Stanford University Report STAN-80-812 (HPP-80-21).

Fink, C., and Carswell, W. Integrated personnel and training for TRADOC system manager (TSM): Technological gaps. ARI Research Report 1238, February 1980.

Foley, J., and Wallace. The art of natural graphic man-machine conversation. Proc. of IEEE, 1974, 61(4), 462-471.

Foley, J., and Wallace, V. The art of natural graphic man-machine conversation. Proceedings of the IEEE, Vol. 62, No. 4, April 1974.

Gannon, J. An experiment for the evaluation of language features. International Journal of Man-Machine Studies, 1876, pp. 61-73.

Garrison, E. Factors in management information system failures. MILPERCEN Alexandria, VA. Final Report, December 1980.

Geiselman, R., and Samet M. Summarizing military information: An application of schema theory. Human Factors, 1980, 22(6), 693-705.

Ghazealh, H., and Nelson. A. A foundation for an international logistics language. Air Force Journal of Logistics, 1981, 5(1), pp. 21-28.

Green, T. Conditional program statements and their comprehensibility to professional programmers. Journal of Occupational Psychology, 1977, 50, pp. 93-109.

Greenstein, J. The use of models of human decision making to enhance human computer interaction. Proceedings of the International Conference on Cybernetics, 1980, pp. 968-970.

Halpern, M. Foundations of the case for natural-language programming. IEEE Spectrum. 1977 (March), pp. 140-144.

Hamilton, M., and Zelchin, S. Requirements definition within acquisition and its relationship to post-deployment software. Higher Order Software Inc. November 1979.

Heidorn, G. Automatic programming through natural language dialogue: A survey. IBM Journal of Research and Development, 1976, July, pp. 302-313.

Herot, C. A spatial graphical man-machine interface. Information Processing, 1980, 1039-1043.

Hill, H., and Kress, G. Development of a methodology for measuring transfer of training effects for tactical training systems. ARI Research Note 80-6, September 1979.

Hill, I. Wouldn't it be nice if we could write computer programs in ordinary English--or would it?. Computer Bulletin, 1972, June, pp. 306-312.

Hoc, J. Role of mental representation in learning a programming language. Journal of Man-Machine Studies, 1977, 9, pp. 87-105.

Huff, S. Preliminary design for complex software systems using graphic decomposition. Proceedings of the International Conference on Cybernetics and Society, 1980, pp. 479-484.

Human Resources Test and Evaluation System: Prototype Handbook. ARI Draft Report, 1978.

Hurst, E., and Kohner, M. Optimization in interactive planning systems. Paper presented at NYU Symposium on Desision Support Systems, May 21-22, 1981.

Johnson, W., Rouse, A., and Rouse, W. An annotated selective bibliography on human performance in fault diagnoses tasks. ARI Technical Report 435, January 1980. (AD A086592)

Johnson, R. Information Requirements: The Major Challenges Part III: Retrieving Information from the Data Group. AIAA International Meeting and Technical Display on Global Technology 2000.

Kopstein, F., Siegel, A., Wilson, L., and Ozkapitan, H. Human Performance in Continuous Operations: Volume II. Management Guide. ARI Research Product 80-4, December 1979.

Lenzychki, H., and Finley, D. How to determine training device requirements and characteristics. ARI Research Product 80-25, May 1980.

Levine, J. Mallamad, S., and Fleishman, E. Decision aids in estimating personnel requirements. Advanced Research Resources Organizations, March 1979.

Lipha, S. Some issues in requirements definition. IEEE Proceedings, 1980, pp. 56-58.

Madlin, S. Combat operations training effectiveness analyses model: 1979 perspective. ARI Technical Report 393. July 1979.

Mahlhotra, A., Thomas, J., Carroll, J., and Miller, L. Cognitive Processes in Design. IBM Research Division Report, 1978.

Matlick, R., Berger, D., et al. Cost and training effectiveness analysis in the Army LCSMM. Litton-Mellonics Final Report. September 1980.

Matlick, R., Rosen, M., and Berger, D. Cost and training effectiveness analysis performance guide. Litton-Mellonics, September 1980.

McAfee, R., and Whinston, A. International Journal of Policy Analyses and Information Systems. 4(3), 1980, 285-295.

Merrill, P. Task analysis - An information processing approach. Paper sponsored by Office of Naval Research, Personnel and Training Research Programs, Psychological Sciences Division, Arlington, VA. (Project No. NR154-280)

Miller, L. et al. Programming in Natural English. IBM Research Report, 15 November 1974. (AD/A-3923)

Miller, R. Psychology for a man-machine problem-solving system. IBM Technical Report TR 00.1246. February 1965.

Mittermeir, R. Enhanced system analysis for requirements analysis. IEEE Transaction, 1979, pp. 300-305.

Moss, R., Sexton, G., and Kearns, J. Design and Evaluation of Complex Man-Machine Systems (Briefing).

Mumy, G., Winger, R., Bobbitt, J., Cunman, J., and Van Newhirle, W. Handbook for the Integration of Countermeasures/Counter- Countermeasure Considerations into the Materiel Development/Acquisition Life Cycle (CM/CCM-HB-80-1). March 1980.

Mullegan, B., and Funaro, J. Front-end analysis: Generic and Nongeneric models. Technical Report: NAVTRAEQUIPCEH IH-325, September 1980.

Neves, D., and Anderson, J. Knowledge compilation: Mechanism for the automation of cognitive skills. Personnel and Training Programs, Office of Naval Research, Arlington, VA 22217, Technical Report 80-4, 1980.

Norman, D. Errors in human performance. Center for Human Information Processing, University of California, San Diego, August 1980, (Report No. 8004).

Osterer, L. A user's guide for GRAPEL - graph for engineering language. Brookhaven National Laboratory, July 1976 TID-4500.

Ostrovofsky, B. Morphology of design of aerospace systems with inclusion of human factors. Final Report. University of Houston, August 1977.

Petrie, F. The Utilization of Requirement Statement Methologies in the U.S. Navy and their Impact on Systems Acquisition. Master's Thesis Naval Postgraduate School, March 1980.

Pew, R., Freehrer, C., Barron, S., and Miller, D. Critical Review and Analysis of Performance Models applicable to Man-Machine Systems Evaluation (AFOSR-TR-77-0520). April 1977.

Proceedings of the TRADOC Chiefs of Analysis Seminar. Hampton, VA, July 16-18, 1980.

Ramsey, H., and Atwood, M. A comparative study of flowcharts and program design languages for the detailed specification of computer programs. ARI Technical Report TR-78-A22, September 1978.

Ramsey, H., and Atwood, M. Cognitive structures in the comprehension and memory of computer programs: An investigation of computer program debugging. ARI Technical Report TR-78-A21, August 1978.

Ramsey, H., and Atwood, M. Man-machine interface design guidance: State of the Art. Proceedings of the International Conference on Cybernetics and Society, 1980, pp. 579-582.

Rhode, A., and Skinner, B., et al. Manpower, personnel, and training requirements for materiel system acquisition. ARI Draft Report. February 1980.

Rivlin, J., Hsu, M., and Marial, P. Knowledge based consultation for finite element structural analysis. Air Force Force Wright Aeronautical Labs, May 1980. AFWAL-TR-80-3069.

Robinson, R. Objective measurement of training readiness. USAWC Military Studies Program Paper. U.S. Army War College, Carlisle, Barracks, PA, 16 May 1980.

Rome, H., and Matuszewski, J. Applications of decision-tree oriented methodology for large-scale systems design. Proceedings of the International Conference on Cybernetics and Society, 1980, pp. 258-271.

Ross, D., and Schoman, K. Structured Analysis for Requirements Definition. IEEE Transactions On Software Engineering, Vol. SE-3, No. 1. January 1977.

Rouse, W., and Rouse, S. Measures of complexity of fault diagnosis tasks. IEEE Transactions on Systems, Man, and Cybernetics, VOL SMC-9, NO. 11. November 1979. pp. 720-72.

Rouse. W. System engineering models of human-machine interaction. Urbana, Illinois: North Holland, 1980.

Sage, A. A methodology for system design. Proceedings of the International Conference on Cybernetics and Society, 1980, pp. 272-277.

Schneiderman, B., and Shapiro, A. Toward a theory of encoded data structures and data translation. International Journal of Computer and Information Sciences, 1976, S(1), pp. 33-42.

Schneider, J., Combs, D., and Folson, T. NETGRAF: A computer graphics aid to the operation and interpretation of NETSIM, a traffic simulation model. Department of Transportation Special Report DOT-RSPA-DPB-50/80/10, March 1980.

Schofield, D., Hillman, A., and Rodgers, J. MM/1, A Man-Machine Interface. Software - Practice and Experience, 1980, 10, pp. 751-763.

Schulz, R., and Farrell, J. Job Aids: Descriptive Authoring Flowcharts for Phases I, II, III, IV, and V of the ISD Model. ARI Research Prodluct 80-13, May 1980 (5 volumes)

Seng, D. Automation of task analysis data: The LSAR/Human Factors engineering enhancement proposal. Technical Memorandum 24-80, 1980, U.S. Army Engineering Laboratory.

Shrier, S. Algorithms for system design. Proceedings of the International Conference on Cybernetics and Society, 1980, pp. 278-283.

Sidbrsky, R., and Parrish, N. Guidelines and criteria for human-computer interface design of battlefield automated systems. Proceedings of the International Conference on Cybernetics and Society, 1980, pp. 590-595.

Siebold, G. The applicability of the ISD 4-Factor Model of Job Analysis in Identifying Task Training Priority in Nine Technical Military Occupational Specialties. ARI Technical Report 432, October 1979.

Siegel, A., Leahy, W., and Wolf, J. A computer model for simulation of message processing in military exercise control and evaluation systems (ARI TR-77-A22). October 1977.

Smith. S. Man/machine interface requirements definition: Task demands and functional requirements. *Proceedings of the International Conference on Cybernetics and Society*, 1980, pp. 585-590.

Smith, S. *Requirements Definition and Design for the Man-Machine Interface in C^3 System Acquisition*. The MITRE Corporation Report No. M80-10, 15 April 1980.

Stavely, A. Design feedback and its use in software design and systems. *Proceedings of the Software Quality and Assurance Workshop*. November 15-17, 1978.

Stohr, E., and Tannivu, M. A data base for operations research models. *International Journal of Policy Analysis and Information Systems*, Vol. 4(1), 1980.

Sussman, G., Holloway, J., and Knight, T. Design aids for digital integrated systems, an artificial intelligence approach. *Proceedings of the IEEE*, 1980, pp. 612-615.

Taylor, J. Computer language and natural language taught comparatively. *Computer and People*, 1977, pp. 7-22.

Teichroew, D., and Hershey, E. *PSL/PSA: A Computer-Aided Technique for Structured Documentation and Analysis of Information Processing Systems*.

Teichroew, D., Macasovic, P., Hershey, E., and Yamamoto, Y. Application of the entity-relationship approach to information processing systems modeling. In P. P. Chen (ed.), *Entity - Relationship Approach to System Analysis and Design*. North-Holland Publishing Co. 1980.

Thomas, E., and Hankins, R. Use of human resource data in weapon system design: Identification of data/data systems and related technologies (AFHRL-TR-79-36). January 1980.

Thomas, E., and Newhouse, D. Human resource data in weapon system design: An initial plan for development of a unified data base (AFHRL-TR-80-25). November 1980.

Tung, Pei-Ti. A systematic approach to complex system design: An application to printed circuit board test system division. MIT Technical Report 13, May 1980.

Van Cleemput. W. A structural design language for computer aided design of digital systems. Stanford Electronics Laboratories. Technical Report No. 136, April 1977.

Wheeler, J. Embedded system design with ADA as the system design language. Software Technology Division, U.S. Army Communications Research and Development Command.

Whitmore, D. Requirements Specification (ASD-TR-78-45). 2 January 1979.

Willison, G. The battalion commander's guide to successful training management. Master Thesis, US Command and General Staff College, 1980. (AS A094436)

Woods, W. Multiple theory formation in high level perception. Bolt, Beranek, and Newman, Inc. Technical Report NO. 38, April 1977.

Woodson, W., and Coburn, W. Human engineering design criteria for modern control/display components and standard parts. U.S. Army Human Engineering Development Lab. U.S. Army Missile Lab, May 1980. (ADA091191)

Woodson, W., and Coburn, C. Recommended modifications to MIL-STD-1472B, Selected sections on controls, displays, and related hardware. U.S. Army Human Engineering Lab Detachment, I.S. Army Missile Lab. (DAAK40-79-C-0145), May 1980. (ADA091477)

Zampora, R., et al. Tutorial in the use of the SRI TREE language system. Stanford Research Institute, December 1974. (AD-A008 927)

APPENDIX A

Review of Existing Army Procedures

The purposes of this appendix are to (1) review Army policies and procedures for mission area analysis with particular emphasis on those policies and procedures related to early training estimation, (2) identify the deficiencies in these procedures which either limit or inhibit the effective early assessment of training requirements, and (3) describe the role of ETES correcting these deficiencies.

The Appendix is divided into three sections. The first section provides an overview of the Army acquisition process and the planning and decision making which supports this process.

The next section provides a review of mission area analyses including a detailed description of the policies, procedures, decision points, and a description of the potential role of ETES in correcting some of the current deficiencies surrounding this period of the acquisition process. This review of mission area analysis procedures updates early reviews of the Concept Exploration and Validation and Demonstration Phases of the acquisiiton process which were presented in the second monthly ETES status report.

This appendix does not attempt to provide a detailed description of all of the various processes and documents associated with the acquisition process - rather it seeks to highlight those processes which are most relevant to early training estimation. The third section provides a detailed listing of Army documents relevant to these processes. The reader seeking more detailed information is urged to consult these documents.

A.1 OVERVIEW OF THE ARMY ACQUISITION PROCESS

This section provides an overview of the Army acquisition process. The ultimate objective of the materiel acquisition process is the timely delivery of an operational system which effectively meets its mission at minimum cost to the Army. As used here the term "operational system" refers to the equipment, software support, personnel and skills required to operate and maintain the equipment, and the support systems required to maintain the equipment and personnel.

The major Army document relating to the materiel acquisition process is the Life Cycle System Management Model, DA-PAM 11-25. The Life Cycle Management Model (LSCMM) describes a series of events which may be used by the program manager for an emerging system to guide the development of his system. (The reader is urged to examine the diagram of the LCSMM events contained on page C-22 of DA PAM 11-25.) It should be pointed out that the LCSMM is only a guide and the program manager has a fair degree of freedom to delete or modify many of the events in the LCSMM. This tailoring is seen as necessary to make the acquisition process responsive to the unique needs of individual programs.

The Army Weapons Acquisition Process (AWSAP) is typically divided into four phases: (1) Concept Exploration; (2) Demonstration and Validation; (3) Full Scale Development; and (4) Production and Deployment. In addition to these four basic phrases, it is possible to add another "phase" (Phase 0) to represent the Mission Area Analysis which takes place prior to concept exploration. Detailed descriptions of the four phases are described in LCSMM Model (DA-PAM 11-25) while detailed description of Mission Area Analysis will be described in a soon to be published TRADOC Handbook on Mission Area Analysis.

The purpose of each of these phases is described in Table A-1. Each phase is terminated by a Department of the Army and/or DOD decision concerning entry into the next phase of development. These decision points are milestones 0 thru 4 of the acquisition process.

The phases of the AWSAP provide Army Managers with the general model for planning and decision-making are shown in Figure A-1. Within each phase of the acquisition process work progresses in defining and operationalizing system, organizational, operational, and support concepts. In the early phases emphasis is placed on defining the system and operational concepts. As these become more fully defined the emphasis is shifted to defining and operationalizing the support and organizational concepts. Within each phase decision-making and planning is supported by the testing of the high risk aspects of these concepts. These test results are then evaluated and this information influences the program planning process. The planning process is central to a successful weapon system acquisition program and allows the acquisition process to be modified to the particular

Phase	Purpose
1. Mission Area Analysis	*To identify those mission elements for which existing or projected capability is deficient and to identify opportunities for capability enhancement through more effective and less costly methods and systems.
2. Concept Exploration	*To explore and identify alternative system concepts.
3. Demonstration and Validation Phase	**To validate the selected solution(s).
4. Full-Scale Development	**Develop a pre-production system which closely approximates the final product.
5. Production and Deployment	**To acquire the system and distribute it.

*ARA 1000-1 Basic Policies for Systems Acquisition.

**DARCOM TRADOC Material Acquisition Handbook (Advance Copy).

TABLE A-1

The purposes of the AWSAP phases.

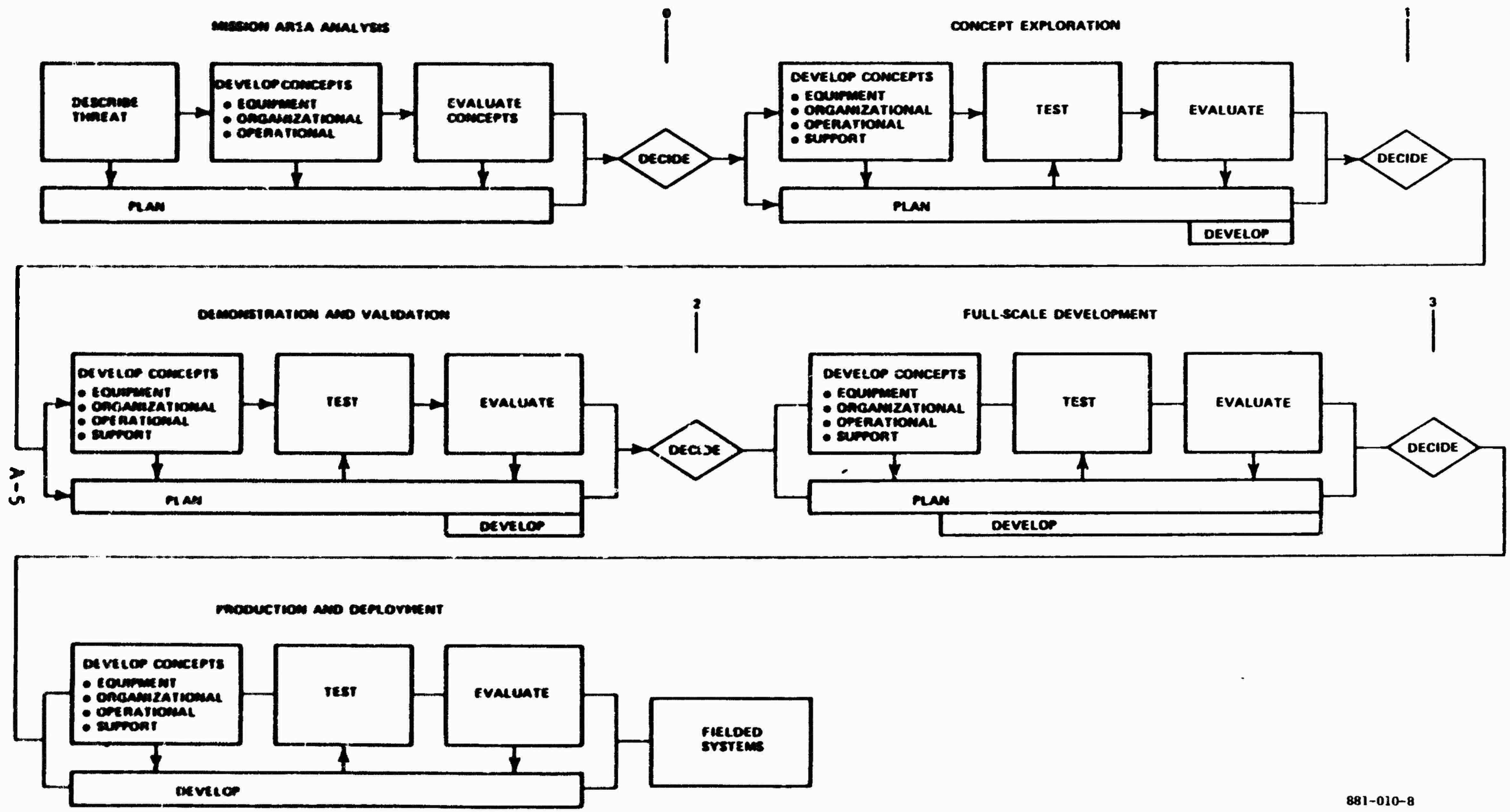


FIGURE A-1 PLANNING AND DECISION MAKING MODEL needs of each specific system. This planning aspect of the AWSAP results in the differences between programs. (e.g. Deletions in phases and activities within phases, differences in testing, and differences in timing of activities.)

The aspects of a program which must be included in the planning process are (1) The development of plans for preparing the documents required to support the decisions which terminates each phase. (2) The planning of the events required to develop the information which will be encluded in these documents (3) Acquiring the necessary resources to accomplish these events and (4) Identifying the critical Equipment, Operational, Organizational, and Support system issues for test and evaluation.

A.2 PHASE 0

A.2.1 MISSION AREA ANALYSIS/MENS DEVELOPMENT

This section summarizes the role of Mission Area Analysis (MAA) requirements as a part of the AWSAP with emphasis on Manpower and Training considerations. MAA takes place prior to the Milestone 0 and is one of the major activities related to the development of the Mission Element Need Statement (MENS).

Definitions of Mission Analysis, Mission Area, Mission Element, the MENS, and other related terms are presented in the sections which follow. DODD 5000.1 (Major Systems Acquisition), revised in March 1980, deleted the definitions of Mission Area and Mission Element previously contained in the January 1977 edition of that document. Accordingly, the former directive is cited as the basis for the definition of these terms.

A.2.1.1 Definitions of MAA Terms

Mission Analysis is any assessment of current or projected US military capability to perform assigned missions. It normally evaluates the interplay of threat, capability, operations concepts, survivability, and other factors such as environmental conditions which bear on the missions of the various Components of the Department of Defense. The primary objective of mission analysis is the identification of deficiencies, so that appropriate corrective action can be initiated. (DODI 5000.2, 19 March 1980)

Mission Area is a segment of the defense mission as established by the Secretary of Defense. (DODD 500.1, 18 January 1977)

Mission Element is a segment of a mission area critical to the accomplishment of the mission area objectives and corresponding to a recommendation for a major system capability as determined by the DoD component. (DODD 5000.1, 18 January 1977)

Mission Element Need Statement (MENS) is the document upon which the Milestone 0 decision is based (Table A-2). It identifies and defines: (a) a specific deficiency or opportunity within a mission area; (b) the relative priority of the deficiency within the mission area; (c) the Defense Intelligence Agency (DIA) validated threat forecast or other factor causing the deficiency; (d) the date the system must be fielded to meet the threat; and (e) the general magnitude

Table A-2

OUTLINE OF MENS

A. Mission

1. Mission Areas - mission areas addressed by MENS
2. Mission Element Need - nature of the need in terms of mission capabilities.

B. Threat or Basis For Need

Basis for the need in terms of an anticipated change in projected threat, in terms of exportable technology or in terms of nonthreat factors.

C. Existing and Planned Capabilities to Meet the Mission

Summary of the existing and planned capabilities to accomplish the mission.

D. Assessment of Need

Evaluation of the ability of current and planned capabilities to cope with the projected threat. This is considered to be the most important part of the MENS. The evaluation is based on the following factors.

1. Deficiency in existing capability
2. Exploitable technological opportunities
3. Force size
4. Vulnerability of existing systems

E. Constraints

Identification of the key boundary conditions including:

1. Timing of need
2. Relative priority within mission area
3. Resources required
4. Logistics, safety, health, energy, environment, and manpower considerations
5. Standardization
6. Interfaces with other systems

F. Resource and Schedule to Meet Milestone of acquisition resources that the DoD Component is willing to invest to correct the deficiency. (DODI 5001.1, 19 March 1980)

A.2.2 Mission Area Analysis Initiation

Mission area deficiencies may be identified at any organizational level of the Army structure. Deficiency detections in this category would normally stem from the observations of unit personnel experiencing the impacts of the deficiencies noted. To a great extent, this type of mission area deficiency is correctable by direct command action, or by comparatively minor operational, support and/or equipment program modifications. Mission area deficiencies resulting in the acquisition of a new weapons system usually stem from a change in the threat to be countered, or from revised US strategy. A major change in the threat may be identified by the DIA, or via the Army's intelligency activities as documented formally by the Assistant Chief of Staff for Intelligence (ACSI), Department of the Army.

Long-range US military objectives and capabilities are documented by the Joint Chiefs of Staff (JCS) in the Joint Strategic Objectives Plan (JSOP) and the Joint Strategic Capabilities Plan (JSCAP). Army implementation of the JSOP/JSCAP is published in the Army Strategic Objectives Plan (ASOP) and the Army Strategic Capabilities Plan (ASCAP). These Army documents are updated annually under the General Staff cognizance of the Deputy Chief of Staff for Operations and Plans (DCSOPS). When approved by the Chief of Staff Army (CSA), the ASOP and the ASCAP provide the primary basis for initiating MAAs within the Department of

Army (DA) General Staff and Special Staff, and within major subordinate commands (e.g., DARCOM, TRADOC), to identify any deficiencies in the Army's capabilities to meet near-term and far-term objectives.

MAAs resulting in the acquisition of a new weapons system may also be initiated to establish new capabilities in response to technologically feasible opportunities. Analysis activities included under MAA occur throughout the lifecycle of a weapon system and MAA has a continuing inpact on the AWSAP. Figure A-2 shows how MAA fits into the Army's overall force need identification and solution development process. MAA does not necessarily result in a requirement for a new materiel acquisition. Other options such as building the technology base, changing operational concepts or changing organizational concepts can also be used to satisfy a mission need. However, from an ETES perspective only mission needs which generate a material acquisition requirement are relevant.

A.2.3 MENS

After the need for a materiel acquisition is identified, an analysis is indicated to further detail this need and coordinate the development of the materiel system. The results of this analysis are documented in a MENS for major systems or in other requirements documents (e.g., LOA) for smaller systems. With the existing deficiencies identified the technology base, existing operational concepts and doctrine and organizational concepts are reviewed for possible solutions. Options for meeting the deficiencies are identified as a result of this process. These options are then assessed for adequacy in the content of the updated

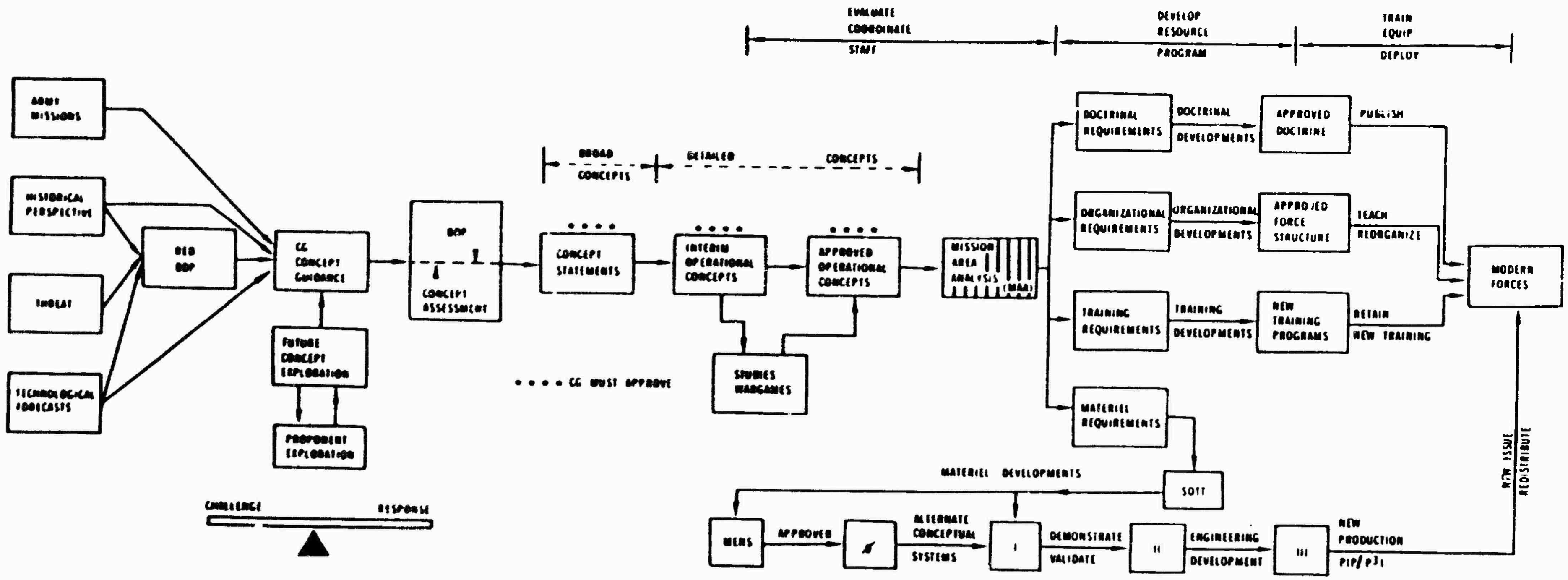


FIGURE A-2 ROLE OF MAA IN ACQUISITION PROCESS*

881-010-5

*Figure taken from TRADOC Reg. 11-7

scenarios. The outputs of this assessment are rank ordered in terms of their ability to satisfy the mission area task or subtask requirement. Along with this rank ordering of options, operational and technical constraints are also identified.

There are four possible options for meeting a battlefield deficiency identified during MAA: (1) build a technology base, (2) change the existing operational concept, (3) change the existing organizational concept and (4) initiate a materiel acquisition. Although these are shown as discrete decisions in Figure A-3, actually, these four decisions are closely intertwined. This interdependency is a primary reason for the continuing interaction between the analysis activities supporting organizational and operational concepts, new technology development, and the development of a new materiel system.

Once a decision is reached to proceed with a materiel acquisition, a management plan is developed, resource limits, milestone and development schedules, and resource requirements are identified. The information developed during this analysis is then used to develop the MENS. Figure A-4 shows the relationship between the MENS content and the information developed during the analysis.

A.2.4 Key Training Products

No key training products are required outputs of this phase of development. However, the information developed during MAA could, and should support training task analysis, system employment and operation. In particular, the following areas are important in this regard. (1) mission area tasks

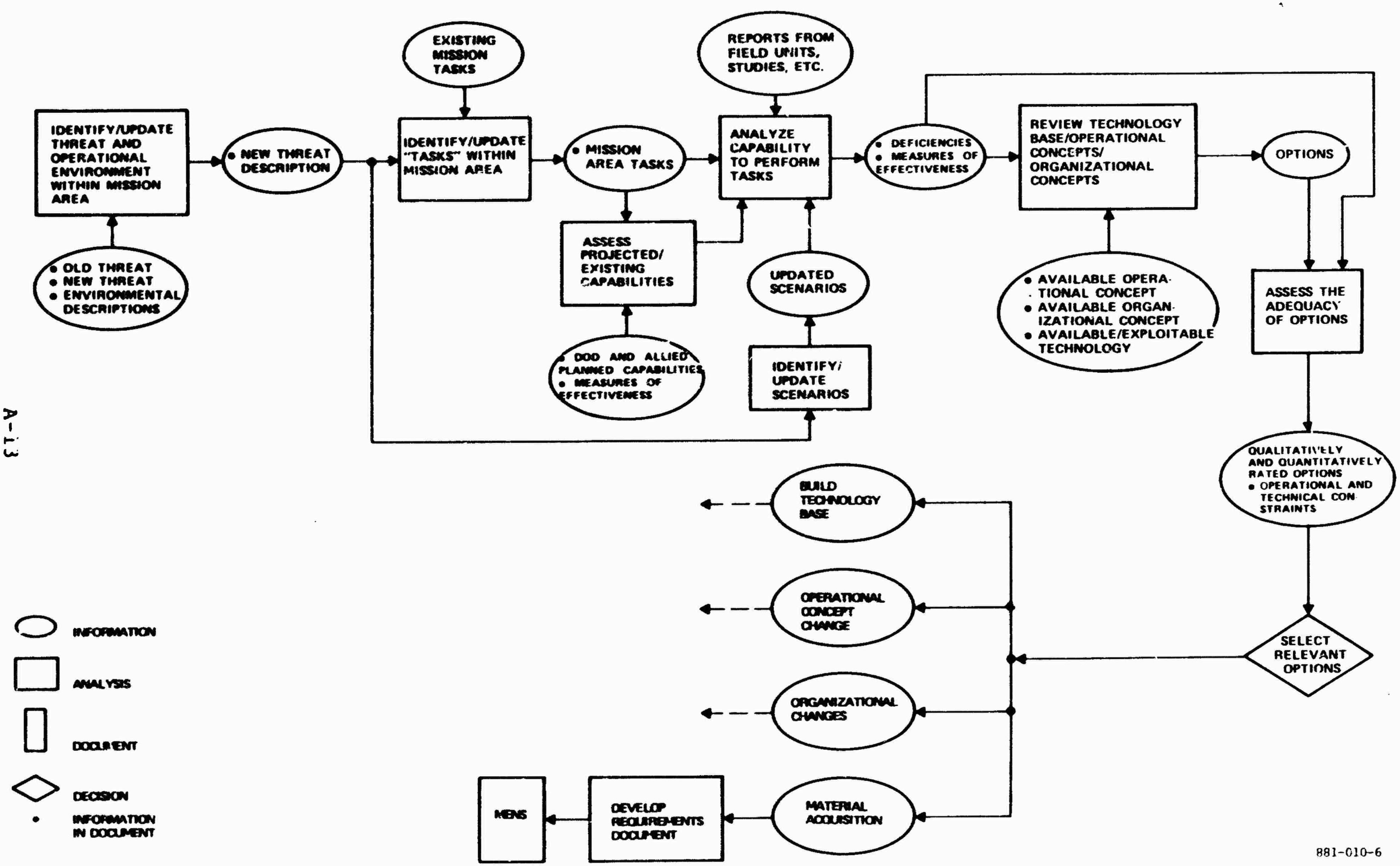


FIGURE A-3 MISSION AREA ANALYSIS

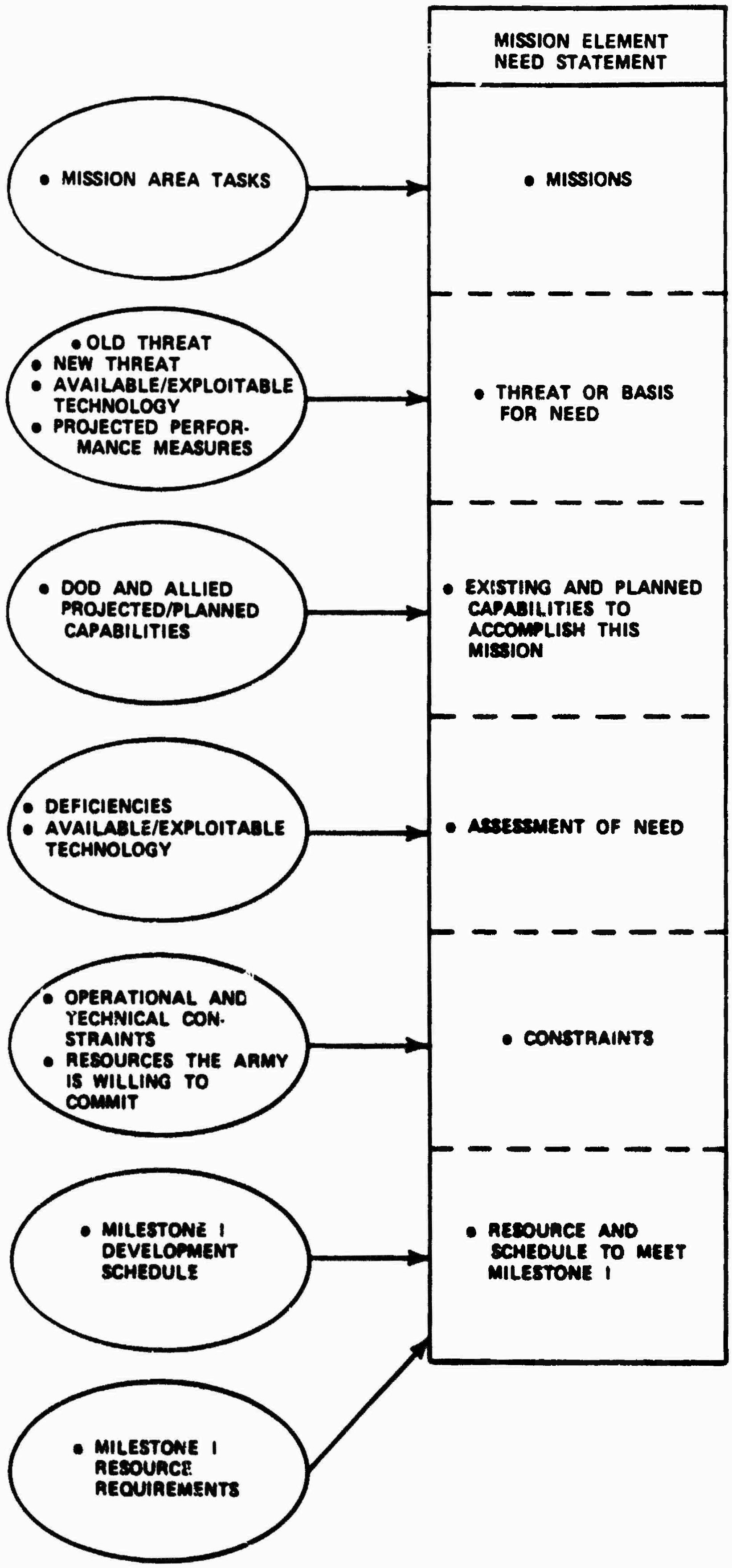


FIGURE A-4 MENS INFORMATION REQUIREMENTS

88l-l10-7

and subtasks, and task performance frequency data; (2) conditional data to include environmental data, e.g., terrain and weather conditions, threat conditions, the types of threats or targets, and the densities of these threats; (3) anticipated organizational concepts to include the organizational echelons and the responsibilities of these echelons; (4) the anticipated tactics and doctrine for employing the weapon system; and (5) data on the anticipated usage rates for the weapon system.

A.2.5 Current Gaps and Limitations

The most glaring problem surrounding mission area analyses is the lack of a systematic set of procedures for conducting these analyses. TRADOC is currently addressing this problem. Hopefully the new TRADOC procedures will include requirements for a detailed functional analysis of a new system (to be completed either slightly before or slightly after MENS development) and the systematic specifications of the system functions and their associated goals or objectives.

The availability of a systematic set of information on mission tasks, subtasks, and system functions greatly facilitate the early generation of individual operator tasks, collective tasks and tactical tasks associated with the new system. The specification of the performance goal is extremely valuable as it lays the foundation for the construction of both materiel and human performance goals.

A.3 BIBLIOGRAPHY OF ARMY DOCUMENTS

Table A-3 presents a bibliography of Army Documents related to ETES.

TABLE A-3

System Acquisition

___, DOD Directive, 5000.1, *Major System Acquisition*,

___, DOD Directive, 5000.2, *Major System Acquisition*,

___, AR 15-14, *System Acquisition Review Council Procedures*, April 1978.

___, AR 70-1, *Army Research, Development, and Acquisition*, February 1977.

___, AR 70-10, *Test And Evaluation During Development and Acquisition Of Material*, August 1975.

___, AR 70-15, *Product Improvement Of Material*, June 1980.

___, AR 70-17, *System/Program/Product Management*, November 1976

___, AR 70-27, *Outline Development Plan/Development Plan/ Army Program Memorandum/Defense Program Memorandum/Decision Coordination Paper*, March 1975.

___, AR 71-3, *Force Development User Testing*, March 1977.

__ __ __, AR 71-5, Force Development Introduction Of New Or Modified Systems/Equipment, July 1969.

__ __ __, AR 71-9, Force Development Material Objectives And Requirements, February 1975.

__ __ __, AR 1000-1, Basic Policies For System Acquisition, April 1978.

__ __ __, DA Pam 11-25, Life Cycle System Management Model For Army Systems, May 1975.

Integrated Logistics Support

__ __ __, MIL-M-1388-1, Military Standard Logistics Support Analysis, October 1973.

__ __ __, DOD DI-L-6138, Integrated Support Plan (ISP), April 1971.

__ __ __, DOD DI-S-6171A, Logistics Support Analysis Record (LSAR) Data, February 1977.

__ __ __, AR 702-2, Army Material Reliability, Availability, And Maintainability (RAM), December 1979.

__ __ __, AR 702-3, Army Material Reliability, Availability, And Maintainability (RAM), November 1976.

__ __ __, AR 750-1, Army Material Maintenance Concepts And Policies, March 1979.

___, TM 38-710, Integrated Logistic Support Implementation Guide For DOD Systems And Equipments, March 1972.

___, TRADOC Reg 700-1, Integrated Logistic Support (ILS), July 1977.

___, DARCOM TRADOC, Material Acquisition Handbook, January 1980.

___, DARCOM-P 750-16, DARCOM Guide To Logistics Support Analysis, January 1979.

___, DARCOM-R 70-16, Management Of Computer Resources In Battlefield Automated Systems, July 1979.

___, DARCOM Primer, ILS Integrated Logistic Support.

Manpower Personnel And Training

___, MIL-M-63035 (TM), Manuals, Technical: Front End Analysis, May 1977.

___, MIL-M-63036A (TM), Military Specification Manuals, Technical: Operators, Preparation Of, April 1980.

___, MIL-M-63040 (TM), Manuals, Technical: Extension Training Materials For Integrated Technical Documentation And Training (ITDT), May 1977.

___, MIL-STD-XYZ, Task Analysis: Requirements For The Use And Application Of Task Analysis (Draft), January 1980.

_ _ _, Army DI-H-1300, Personnel and Training Requirements, December 1969.

_ _ _, AR 108-2, Army Training And Audiovisual Support, July 1976.

_ _ _, AR 350-1, Army Training, May 1978.

_ _ _, AR 600-4, Integrated Personnel Support (IPS), June 1978.

_ _ _, AR 611-1, Military Occupational Classification Structure Development And Implementation, April 1976.

_ _ _, TRADOC Reg 71-12, Total System Management-TRADOC System Manager (TSM), September 1978.

_ _ _, TRADOC Reg 310-2, Development, Preparation, And Management Of Training And Evaluation Program (ARTEP), December 1979.

_ _ _, TRADOC Reg 351-4, Job & Task Analysis, March 1979.

_ _ _, TRADOC Reg 350-7 (Draft), A Systems Approach to Training.

_ _ _, TRADOC Cir 70-1, Training Device Development, February 1979.

_ _ _, TRADOC Cir 350-2, Officer Job/Task Analysis And Training Development, March 1979.

_ _ _, TRADOC Cir 350-3, Individual/Collective Training And Development Glossary, December 1979.

_ _ _, TRADOC Cir 351-1, Common Job And Task Management, January 1980.

_ _ _, TRADOC Cir 351-2, Army Correspondence Coarse Program: Subcourses, June 1980.

_ _ _, TRADOC Cir 351-3, Training Requirements Analysis System (TRAS)/Individual Training Plan (ITP), December 1979.

_ _ _, TRADOC Cir 351-7, Job Training Program (JTP), April 1980.

_ _ _, TRADOC Cir 351-8, Individual And Collective Training Plan For Developing Systems Policy And Procedures, May 1980.

_ _ _, TRADOC Cir 351-12, Format For Programs Of Instruction (PDI), April 1980.

_ _ _, TRADOC Cir 351-28, Soldier's Manuals, Commander's Manual And Job Books Policy And Procedures, December 1978.

_ _ _, TRADOC PAM 71-8, Analyzing Training Effectiveness, February 1976.

_ _ _, TRADOC PAM 71-10, Cost & Training Effectiveness Analysis, (Draft).

_ _ _, TRADOC PAM 310-8, Collective Front-End Analysis (CFEA) For Development Of Army Training And Evaluation Program (ARTEP), (Draft).

___ ___ ___, TRADOC PAM 350-30, Interservice Procedures For Instructional Systems Development; Executive Summary & Model, August 1975.

___ ___ ___, TRADOC PAM 350-30, Interservice Procedures For Instructional Systems Development: Phase I Analyze, August 1975.

___ ___ ___, TRADOC PAM 350-30, Interservice Procedures For Instructional Systems Devleopment: Phase II Design, August 1975.

___ ___ ___, TRADOC PAM 350-30, Interservice Procedures For Instructional Systems Development: Phase III Develop, August 1975.

___ ___ ___, TRADOC PAM 350-30, Interservice Procedures For Instructional Systems Development: Phase IV & V Implement & Control, August 1975.

___ ___ ___, TRADOC PAM 351-4, Job And Task Analysis Handbook. August 1979.

___ ___ ___, TRADOC PAM 351-6, Job Performance Aid: Job And Task Analysis,

___ ___ ___, USASC - FG Pamphlet 350-8, Skill Performance Aids, February 1979.

Bonder, S. Review Of Army Force Modernization And Associated Manpower, Personnel, And Training Process, Alexandria, VA: US Army Research Institute for the Behavioral and Social Sciences, January 1981.

Fink, D. C., Carswell, W. A. Integrated Personnel And Training Information For TRADOC System Manager (TSM): Technological Gaps, Alexandria, VA" US Army Research Institute for the Behavioral and Social Sciences, February 1980.

Hanson, V. L., Parifoy, G. R., Jr. TSM Guide To Training Development And Acquisition For Major Systems, Alexandria, VA: US Army Research Institute for the Behavioral and Social Sciences, March 1978.

Lenzychi, H. R., Finley, D. L. How To Determine Training Device Requirements And Characteristics: A Handbook For Training Developers, Alexandria, VA: IS Army Research Institute for the Behavioral and Social Sciences, May 1980.

Matlick, R. K., Berger, D. C., Knerr, C. M., Chiorini, J. R. Cost And Training Effectiveness Analysis In The Army Life Cycle Systems Management Model, Alexandria, VA: US Army Research Institute for the Behavioral and Social Sciences, September 1980.

Matlick, R. K., Rosen, M. H., Berger, D. C. Cost And Training Effectiveness Analysis Performance Guide, Alexandria, VA: US Army Research Institute for the Behavioral and Social Sciences, September 1980.

Rhode, A. S., Skinner, B. B., Mullin, J. L., Friedman, F. L., Franco, M. M. Manpower, Personnel And Training Requirements For Material System Acquisition, Alexandria, VA: US Army Research Institute for the Behavioral and Social Sciences, February 1980. , AR 5-5, The Army Study System, June 1978.

Other

___, AR 10-38, United States Army Concepts Analysis Agency (Short Title: CAA), November 1978.

___, AR 10-41, Organization And Functions United States Army Training And Doctrine Command, July 1973.

___, AR 11-18, Army Programs The Cost Analysis Program, March 1976.

___, AR 71-2, Basis Of Issue Plan, April 1976.

___, AR 310-31, Management Systems For Tables Of Organization And Equipment (The TOE System), October 1974.

___, AR381-11, Threat Analysis, August 1974.

___, AR 611-3, Army Occupational Survey Program (ADSP), November 1977.

___, DA PAM 5-5, Guidance For Study Sponsors And Study Advisory Groups, October 1976.

___, FM 770-78, System Engineering, April 1979.

___, TRADOC Reg 11-5, Cost Analysis Program (MOS Training Costs) RCSATRM - 159 (RI), November 1977.

___, TRADOC Reg 11-7, Operational Concepts And Army Doctrine, December 1980.

___, TRADOC Reg 11-8, Cost And Operational Effectiveness Analysis In The Material Acquisition Process, March 1977.

___, TRADOC Reg 71-4, TRADOC Standard Scenarios For Combat Developments, March 1977.

___, TRADOC, Handbook: Mission Area Analysis, December 1979.

___, TRADOC Reg 71-5, Scenario Oriented Recurring Evaluation System (Scores, November 1977.

___, TRADOC Reg 71-10, Integration Of The TOE Development Process And The Scenario Oriented Recurring Evaluation System, September 1977.

Murry, G. B., Vinger, R. F. C., Bobbitt, J. R., III, Carman, J. W., Van Newkirk, W. K. Handbook For The Integration Of Countermeasures/Counter-Countermeasure Considerations Into The Material Development/Acquisition Life Cycle, Department of the Army United States Army Material Development And Readiness Command United States Army Electronics Research And Development Command Countermeasures/Counter-Countermeasures Directorate, March 1980.

Price, H. E., Feorello, M., Lowry, J. C., Smith, M. G., Kidd, J. S. The Contrabution Of Human Factors In Military System Development: Methodological Considerations, Alexandria, VA: US Army Research Institute for the Behavioral and Social Sciences, July 1980.

APPENDIX B

EXAMPLE OUTPUTS FOR FUNCTIONAL REQUIREMENTS AND DESIGN CONCEPTS

This appendix provides examples of the types of outputs associated with the functional requirements and design concept areas. These outputs are only designed to demonstrate the basic types of information which will be required. The exact format of the outputs cannot be determined until specific output mechanisms have been identified.

FIGURE B-1 TOP LEVEL FUNCTIONAL REQUIREMENTS

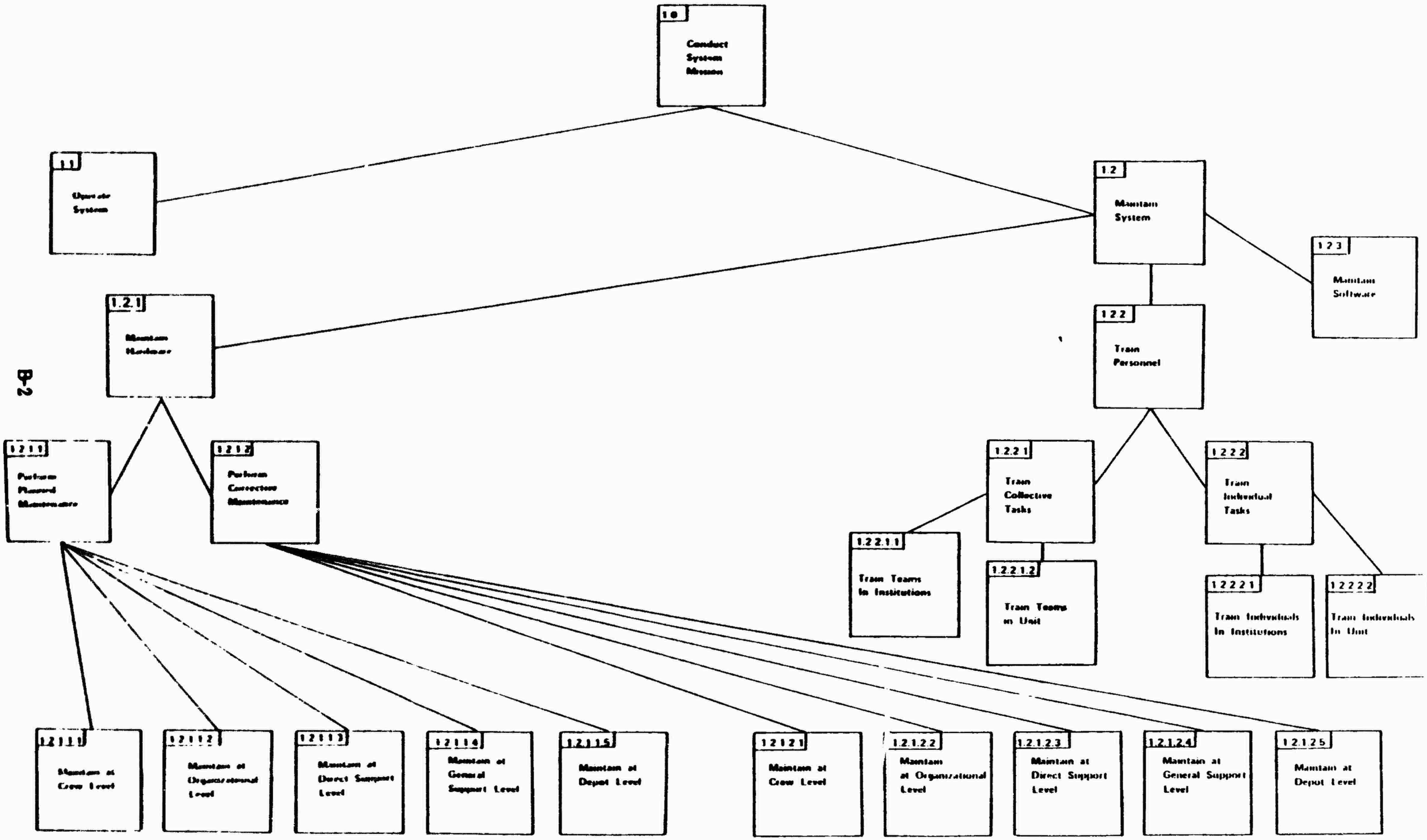


FIGURE B-2 FUNCTIONAL HIERARCHY OPERATIONS

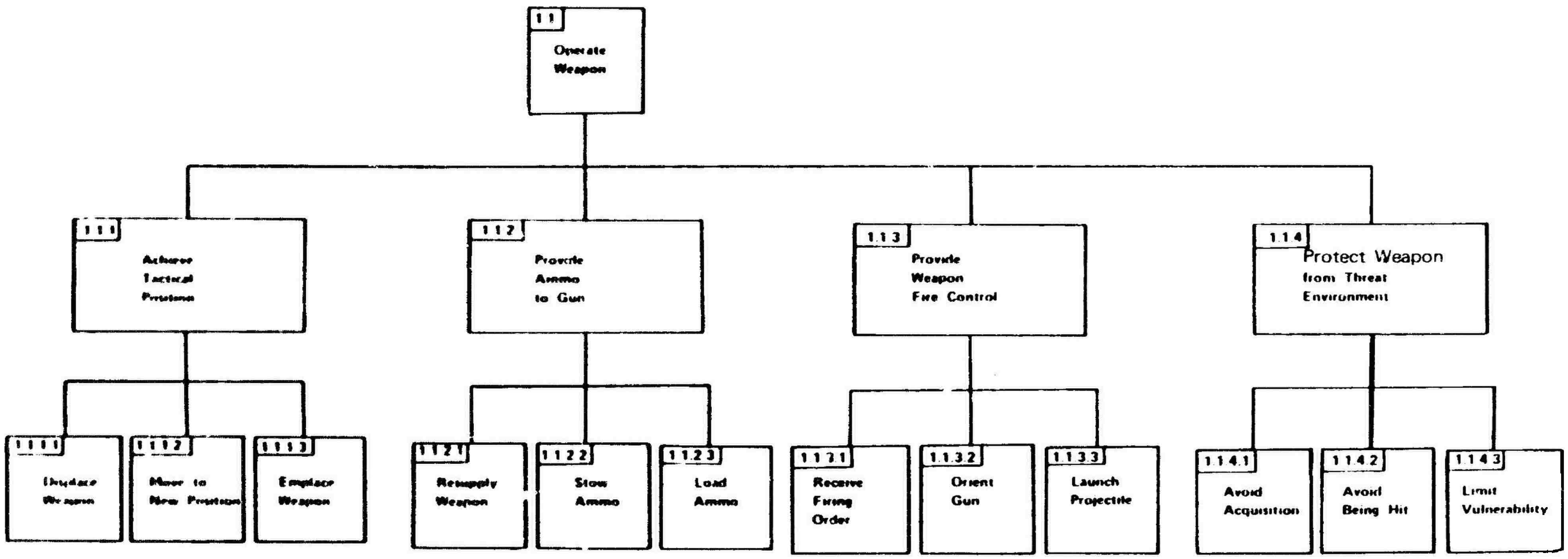


FIGURE B-3 ACTIVITY FLOW

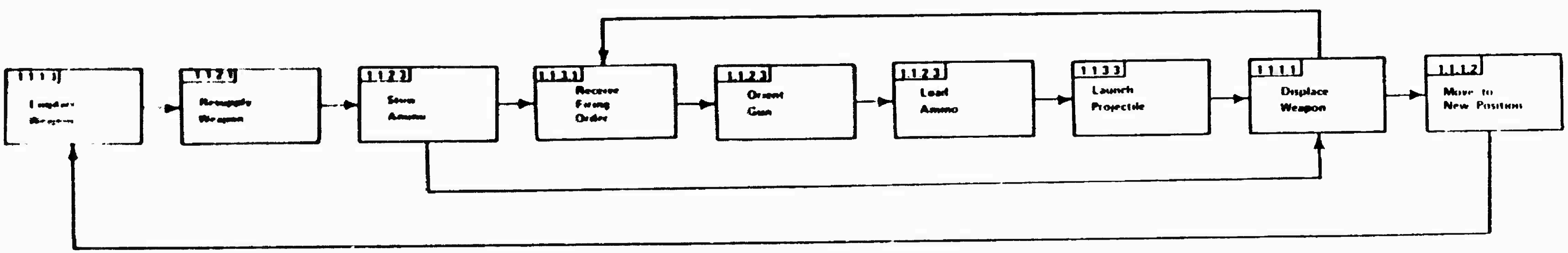


TABLE B-1 PERFORMANCE GOALS BY FUNCTION

Function No.	Function Title	Goal	Reference
1.1.1.1	Displace Weapon	o Displacement time must not exceed 5 min.	
1.1.1.2	Move to New Position	o Must be able to travel at least 10 km per hour	MENS
1.1.1.3	Emplace Weapon	o Emplacement time must not exceed 10 min.	
1.1.2.1	Resupply Weapon	o Must be able to receive 200 rounds in 20 min.	MENS
1.1.2.2	Stow Ammo.	o Must be able to store 400 rounds of ammo.	
1.1.2.3	Load Ammo	o Must be able to load 1 round per minute	MENS
1.1.3.1	Receive Firing Order	o Must be able to receive 1 order per minute	MENS
1.1.3.2	Orient Gun	o Projectile must be capable of hitting within 10 feet of targer 90% of time	2
1.2.1.1	Perform Preventative Maintenance	o Total preventative maintenance requirements must not exceed 4 hours per week	2
1.2.1.2	Perform Corrective Maintenance	o MTBF must not be less than 100 hours o System availability must not be less than 95%	2
1.1.4.3	Limit Vulnerability	o System must be survivable in NBC environment as specified in scores	2 Scores

TABLE B-2 TERRAIN IMPACTS ON FUNCTION

Function No.	Function	Environmental Variable	Variables Impacted	Reference(s)
1.1.1.1	Displace Weapon	Terrain	Displacement Procedures	1
		Weather	Displacement Procedures	
1.1.1.2	Move to New Position	Terrain	Movement Procedures	1
		Weather	Movement Procedures	
1.1.1.3	Emplace Weapon	Terrain	Emplacement Procedures	2
		Weather	Emplacement Procedures	
1.1.3.2	Orient Gun	Weather	Orientation Procedures	1

TABLE B-3 THREAT IMPACTS ON FUNCTIONS

Function No.	Function	Threat Variable	Variables Impacted	References
1.1.2.3	Load Ammo.	Type of Target Target Density	Type of Ammo. Type of Ammo.	1
1.1.3.1	Receive Firing Order	Type of Target Target Density Target Position	Type of Fire Order Type of Fire Order Type of Fire Order	2
1.3.1	Avoid Acquisition	Type of Threat	Acquisition Means to be Avoided	3
1.3.2	Avoid Being Hit	Type of Threat Threat Density	Fire to be Avoided	4
1.1.4.3	Limit Vulnerability	Type of Threat Threat Density Threat Position	Fire to be Avoided	5

TABLE B-4 MISSION PROFILE IMPACTS ON FUNCTIONS

Funct. No.	Function	Impact Variables	Ref.	Var. No.	Submissions Impacted	Variables Impacted	Ref.	% Time Spent in Submission
1.1.3.1	Receive Firing Order	Firing Order Data	1	1	Target Servicing	1,2,3	3	75%
					Counterfire	1,2,3	3	15%
					Air Defense Suppression	1,2,3	3	10%
1.1.3.2	Orient Gun	Orientation Procedures	1	2				
1.1.2.3	Load Ammo.	Type of Ammo.	1	3				

FIGURE B-4 GENERIC EQUIPMENT HIERARCHICAL STRUCTURE

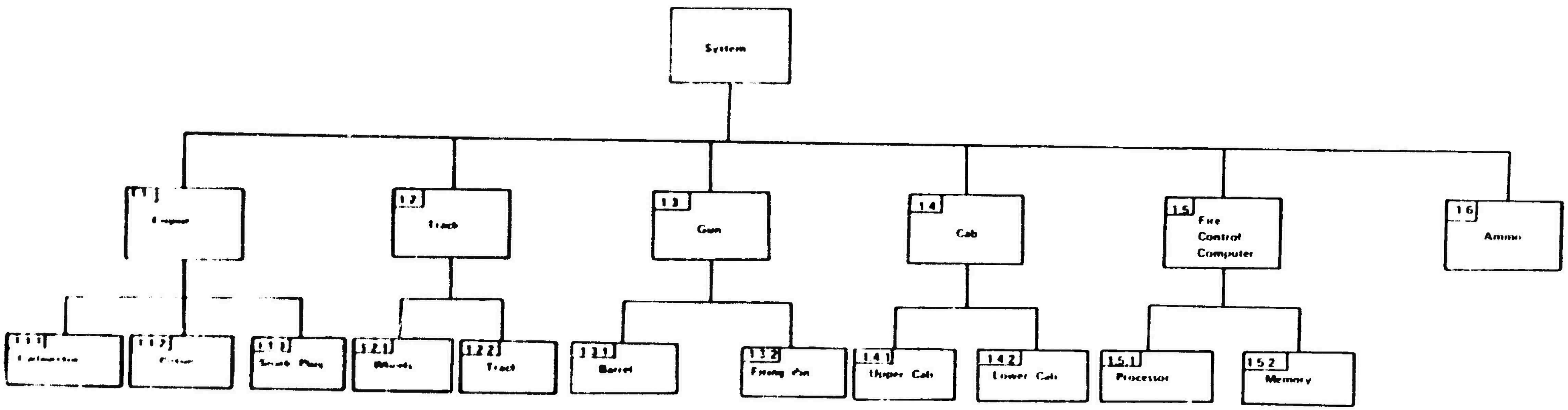


TABLE B-5 GENERIC EQUIPMENT HIERARCHY

APPROVED DESIGN CONCEPTS AND COMPARABLE EXISTING EQUIPMENT

Generic Equipment Number	Generic Equipment Title	Approved Generic Struct.	Related Functions Number	Related Functions Title	Relevant Existing Equipment	Degree of Difference	Software Reqs.
1.1	Engine	x	1.1.1.2	Move to new position	M100 engine	None	
1.1.1	Carburator		–	–	–		
1.1.2	Piston		–	–	–		
1.1.3	Spark Plug		–	–	–		
1.2.1	Track	X	1.1.1.2	Move to new position	M100 track	very small	
1.2.1	Wheels		–	–	–		
1.2.2	Track		–	–	–		
1.3	Gun	X	1.1.3.2;1.1.3.3	Orient Gun; Launch Projectile	M100 gun	Small	
1.3.1	Barrel		–	–	–		
1.3.2	Firing Pin		–	–	–		
1.4	Cab	X	1.1,1.3.3	Operate System; Limit Vulnerability	M100 cab	Small	
1.4.1	Upper Cab	X	1.1.1.1;1.1.3.1	Displace Weapon; Emplace Weapon	–		
1.4.2	Lower Cab	X	1.1.1.2;1.1.2.2	Move to new position; Stow	–	None	
1.5	Fire Control Computer	X	1.1.3.2	Orient gun	Fire Control Computer		
15.1	Processor		–	–	–		
15.2	Memory		–	–	–		
1.6	A [illegible] vition		–	–	–		

TABLE B-6 DESIGN ALTERNATIVES

Function	Alternative
1.5 Fire Control Computer	Company A Fire Control Computer A
	Company B Fire Control Computer B

*Two companies have submitted proposals. They propose essentially the same designs, they only differ in the fire control computer they propose to utilize.

FIGURE B-5

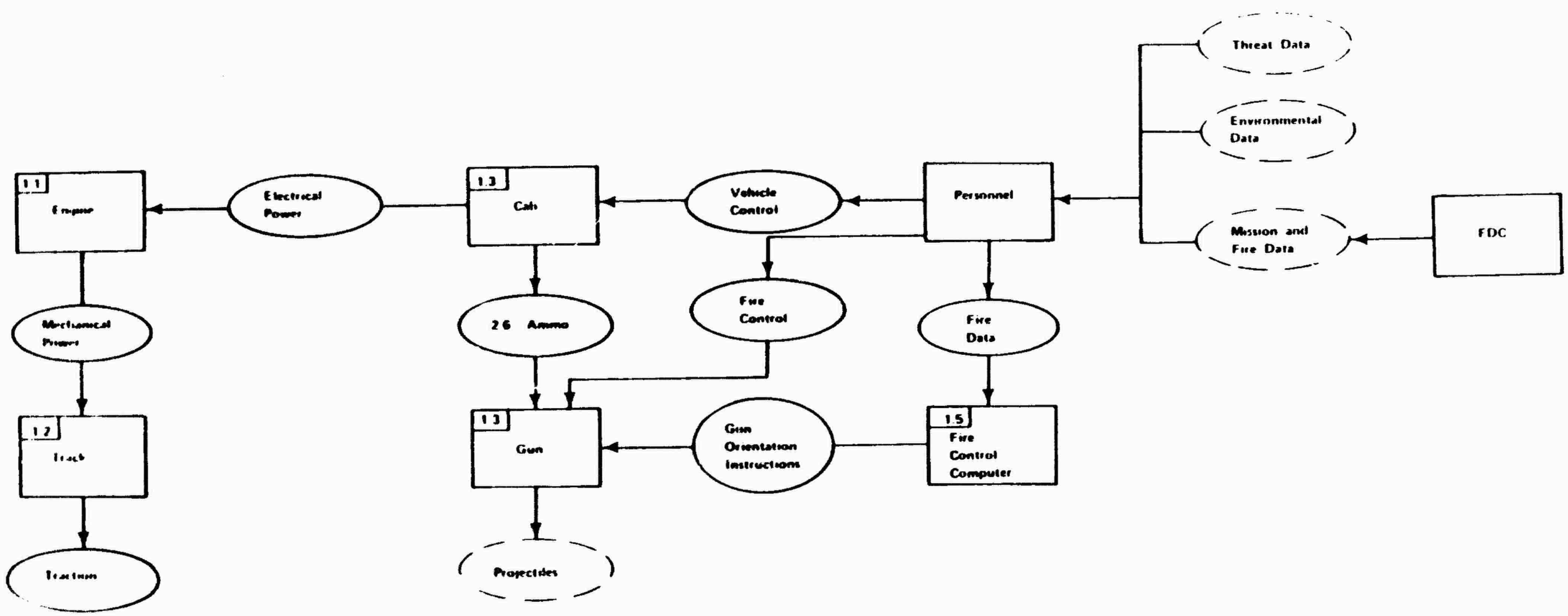
EQUIPMENT HIERARCHICAL STRUCTURE – DESIGN ALTERNATIVES

Diagram would be similar to Figure B-4, only with specific equipment replacing generic equipment.

TABLE B-7 EQUIPMENT HIERARCHY AND COMPARABLE EXISTING EQUIPMENT FOR DESIGN X

Table would be similar to Table 3-5 except that specific equipment would replace generic equipment in second column.

FIGURE B-6 INFORMATION FLOW



APPENDIX C

REVIEW OF PSYCHOLOGICAL RESEARCH RELATED TO DESIGN

This Appendix is a review of psychological research related to the construction of the ETES SDT. It is divided into three sections. The first section reviews the role of the SDT in the engineering design process and indicates what general areas of psychological research are related to this role. The second section reviews past psychological research relating to engineering and software design and the implications that this research has for the SDT. The third section is a bibliography of the psychological research related to design.

C.1 REVIEW OF DESIGN PROCESS

A detailed description of the overall weapon system design process was presented in Appendix A and the role of the SDT was detailed in Section 2. Hence, only a brief review is presented here. The SDT will essentially be a data base management system for describing, storing, updating, and communicating the general system elements listed in Table C-1. (A more detailed breakdown of these general elements is presented in Section 2.) The SDT will have the capability of describing these system elements for several different design alternatives and for describing estimated elements when actual data is not available.

Appendix D discusses psychological research relating to the engineering design process--as opposed to the more general weapon system design process. It describes research relating to the more general weapon system development process and its associated information flow problems. One

Table C-1

SYSTEM ELEMENTS CONTAINED IN SDT*

General Elements Described by SDT	Description Required by Design Alternative	Description Required for Estimated as well as Actual Data
Functional Requirements	No	No
Design Concepts	Yes	Yes
Equipment-Task Interface	Yes	Yes
Tasks	Yes	Yes
Skills and Knowledges	Yes	Yes
Training Program Elements	Yes	Yes

*More detailed description of SDT elements is presented in Section 2.

can distinguish between these two processes by noting that the engineering design process refers to the activities of a single design engineer while the weapon system development process refers to the activities of a large number of different individuals in a variety of different technical disciplines. Thus, psychological research relating to engineering design is primarily concerned with individual cognitive processes while psychological research relating to weapon system development is concerned with information flow and man-computer interactions.

C.2 REVIEW OF PSYCHOLOGICAL RESEARCH RELATING TO THE ENGINEERING DESIGN PROCESS[1]

Numerous studies (see Thomas and Hankins, 1980, and Clemson University, 1979, for a review) have shown that while human resources data can and should play a significant role in the systems design process, designers do not always utilize the full potential of the data. There are a number of reasons for this. Certainly it is partly due to the past training and experience of the designer. Another reason may be that human resources data is often presented in a form that is incompatible with the cognitive processes the designer employs during design. One important consideration, then, would be to present human resources data in a form that is compatible with the design process used by designers. But to do this, it would seem appropriate to characterize the design process itself in cognitive terms. This section attempts to identify the cognitive processes underlying the

---

1 Section C.2 was prepared by Drs. Paul Ronco and Jack Hansen under a subcontract to DRC. They are now part-time DRC employees.

engineering design process, to identify and classify potential tools and aids related to this design process, and to assess the applicability of these aids to the ETES SDT. In support of this section, an annotated bibliography of the major references in this area has also been developed. This bibliography is contained in Section C.3.

C.2.1 The Engineering Design Process

Miller (1969) has suggested that the design process can be broken down into essentially three fundamental underlying processes -- namely, goal elaboration, design generation and design evaluation.

1. Goal Elaboration - This process is initiated by a statement of the problem and an examination of the goals. It involves goal decomposition and sub-goal selection until sub-goals are specific enough to be considered as functional requirements. The designer may or may not be involved in the specification of the higher level goals. Often the designer is simply presented with a statement of the problem and the operational requirements of the system. However, whether the designer is directly involved in the goal elaboration process or not, he needs to be supplied with goal-related information including the following:

   - systems and equipment requirements (e.g., equipment must be portable.)

   - functional analyses

- applicable constraints

- conclusions drawn from previous data analyses and inputs

- design criteria, specified in the forms of specifications and the designer's own accumulated knowledge and experience. (The latter may not be immediately available to the designer because of problems associated with information retrieval from memory. That is, often the designer possesses relevant knowledge which is not spontaneously accessed. However, if properly cued, recall and recognition of this data is possible.)

In analyzing the problem and establishing sub-goals, the designer will require certain inputs. The data he selects for input will depend on how he construed the problem. His interpretation of the problem will in turn depend on his own past experience with similar problems as well as any more analytically based knowledge he may possess.

2. Design Generation - The various inputs described above contain information which the designer will utilize to resolve the problem. The input, however, is different from the information it contains. The information is a product of the designer's interpretation of the implications and application of the inputs to the problem situation. This process of deriving the correct design implications from an input is the most

difficult part of the design process. Some implications are relatively obvious (e.g., the need for portability) whereas others can be quite obscure.

Design generation, then, starts with functional requirements and attempts to come up with a design organization which meets the functional requirements. In one sense, the design process can be thought of as an effort to organize design elements and characteristics into a whole. It is not simply a treatment of isolated parts, even though sub-problems may be worked on in isolation. It is a "holistic" process. This is a problem in and of itself. The design elements and characteristics developed in the achievement of various sub-goals may assume different values as a function of the total "Gestalt." Unfortunately, because of human memory limitations, it is often difficult for the designer to keep track of all the elements that enter into and affect the design and systems operations, and he thus may require some assistance in keeping track of the elements and relationships in the particular design problem.

3. Design Evaluation - The designer must be continuously evaluating how well a proposed design meets the stated characteristics and requirements of the system. An important feature of this process is that it may uncover new requirements. The whole process is continuously evolving. New inputs which arrive, usually sequentially, are

integrated, analyzed and accepted or rejected. The impact is continually changing as a function of evaluation.

Given this description of the engineering design process, an attempt was made to identify a psychological paradigm which closely paralleled this process with the hope that this paradigm would provide a systematic framework for organizing the psychological research related to design. Fortunately, several past investigators of the design process (e.g., Miller, 1969; Atwood, et al, 1979) have identified such a paradigm: namely, problem solving behavior. Design is obviously (at least in part) a form of problem solving.[2]

In the subsection which follows, the problem solving literature relevant to the design process is reviewed.

C.2.2 Problem Solving Behavior and Its Implication for Design

Several different psychologists have come up with schemes for categorizing the different phases in problem solving behavior. However, two of these schemes are especially relevant to the engineering process. One scheme, developed

---

2 It must be emphasized that the terms design and "design process" refer to the process that an individual engineer might go through in developing a design. They do not refer to the more general acquisition process and the overall design of the system which would be accomplished by a large number of different people. However, Appendix D does deal with research related to this more general process.

by Miller, 1969, is specifically concerned with the hardware design process and is closely linked to the "classical" psychological literature in problem solving. The second scheme was developed by Ramsey and Atwood, 1979, to describe the software design process. Actually both schemes are compatible with one another. Table C-2 displays these two schemes and indicates how they can be directly related to one another, to the phases of the design process described in Section C.2, and to the activities in the weapon acquisition process. A more detailed discussion of these two schemes is provided in the sections which follow.

C.2.2.1 Problem Solving and Design: Implications from Classical Problem Solving Research

In many of the classical discussions of problem solving activity, problem solving is described as the process of finding a connection between the known and the unknown. In other words, problem solving refers to the process of "generating" a connection between the known and unknown. This means that tasks in which the principal activity involves noticing that some event occurs or is present, are recognition tasks and not problem solving tasks. (See Wickelgren, 1979, for a discussion of problem solving behavior and its relationship to other cognitive processes such as recognition.) Utilizing Miller's (1969) framework, the problem solving process can be broken down into four basic phases:

1. Problem Definition - Any problem requires the problem solver to cognitively represent the information in some way. He usually utilizes auxiliary memory devices that range from paper and

Table C-2

SCHEMES FOR DESCRIBING PROBLEM SOLVING BEHAVIOR

Problem Solving Stages		Processes in Design	Periods in Acquisition Process
Problem Recognition	Problem Recognition	————	Ongoing Mission Area Analysis
Problem Definition	Problem Definition	Goal Elaboration	Requirements Analysis
Goal Definition	Functional Analysis	Goal Elaboration	Requirements Analysis
Strategy Collection	Organization	Goal Elaboration	Requirements Analysis
Alternative Generation	Invention of Mechanism	Design Generation	System Design and Development
Alternative Evaluation	Invention of Mechanism	Design Evaluation	System Design and Development
Alternative Selection and Execution	Invention of Mechanism	Design Evaluation	System Design and Development

pencil, drawn diagrams, and written equations to complex computer programs. Satisfactory problem solving performance requires the problem solver to develop a concept of the problem -- to form an appropriate representation of the problem. There seems to be unanimous agreement that this is one of the crucial steps in problem solving. "Problem representation" refers not merely to formal notation (syntactics) but encompasses more specifically the designer's perception of the logical structure of the problem. This representation, of course, includes an understanding of the goals (see Klein and Weitzenfield, 1979, or a review of procedures for improving goal understanding). If the goals are not clear, it is difficult (if not impossible) for the problem solver to generate procedures that will accomplish the goals. Thus, the first phase in problem solving -- problem definition -- is extremely critical.

2. <u>Organization</u> - The first element, "Problem Definition," blends imperceptibly into the second element, which might be labeled "Organization," the construction of some type of conceptual model. Typically in design, the model is constructed on the basis of analogy. The designer sees the current problem as in a class of other problems and he analyzes the problem as being similar to some prior problem. The prior instances, the analogous ones, serve as a vehicle for thought and the construction of a conceptual model. For example, if the task is to design a

training device to accomplish a given training requirement, the first step typically is to assess the training devices previously designed to accomplish the same or similar training requirements. These serve as analogies and provide a variety of candidate features to include in the new device.

3. Function Analysis - The development of a conceptual model also involves the analysis of function. This involves getting an idea of what kinds of things must happen for the system under design to operate and perform its function.

4. Invention of a Mechanism - Finally, the designer must get an idea of the physical structure and operation whereby the goals and mission of the system can be accomplished. As mentioned previously, he often gets to this point through the use of analogical reasoning.

In addition to the multi-stage approach toward describing the problem solving process, another approach to conceptualizing the design process has been to describe problem-solving behavior in terms of an "analytic/synthetic" dimension (see Ramsey and Atwood, 1979, for a more detailed description of this dimension). Under the analytic/synthetic conceptualization the design process is viewed as a two-step process, in which the individual components of the problem are first identified. This analysis is then followed by the synthesis of a solution. The main distinction between the analytic/synthetic and the multi-stage conceptualizations of the design process is that analytic/synthetic approach does not emphasize the explicit search for alternative solutions. Instead, a solution is "synthesized" based upon pattern recognition and the use of components from past solutions to other problems.

Analysis usually consists of decomposition or problem reduction. The overall problem is decomposed into subproblems which are more manageable and easier to solve. The subproblems consist of getting from givens to the subgoal and then from the subgoal to the goal. If one defines subgoals that have a high probability of being on a solution path in the problem tree, the search is greatly reduced. This calls for intelligent problem solving methods that entail trying out the likelier possibilities first, even if such heuristic methods do not always work (see Klein and Weitzenfeld, 1979). This is, of course, what intelligent human or computer problem solvers do. They guide search by using search reduction methods which prune large problem trees in clever ways. They also use representative methods which code or recode problems so as to replace large problem trees with small ones that are nevertheless equivalent to the large trees with respect to solving the given problem. Problem reduction methods, however, themselves raise problems -- namely, keeping track of the various goals and subgoals. Keeping track of active goals appears to be a principal source of information load in design. The human problem solver has fairly severe cognitive limitations (especially short term memory) within which to operate. When a problem reduction strategy is used on a highly complex problem, it is very likely that the problem solver will have difficulty in recalling and utilizing the global information required to deal with complex subproblem interdependencies. Thus, in the design process the identification of points involving high information load should help in the determination of appropriate automatic aids to reduce this load (see Atwood, et al, 1979).

While we can classify the problem solving or reasoning processes used in design as "analogical reasoning" or "analytic/synthetic," this is not meant to imply that all designers utilize the same strategies or same mode of thinking. There are individual differences between designers as to the mode of cognitive activity they use. For example, Greeno (1973) makes a distinction between formal and informal reasoning. Formal reasoning involves the use of syntactic information, formal languages and relatively mechanical procedures. Informal reasoning involves semantic models. The reasoning processes differ considerably between these two classes. Larkin presents data which suggest that very experienced physicists may adopt predominantly semantic (intuitive) approaches to the solution of physics problems. Relatively inexperienced physics students, on the other hand, tend to proceed immediately to the use and solution of mathematical equations and thus employ formal reasoning. Presumably, approaching the problem with informal reasoning would allow the problem solver to make much greater use of his knowledge of the problem domain and experience with conceptually related problems. It is possible that the very formal syntactic approach may well deprive the designer of the ability to use problem relevant knowledge to resolve difficulties which arise in design. Thus, algorithmic strategies may limit the advantageous use of relevant knowledge by experienced designers. On the other hand, they may be quite advantageous if used by inexperienced designers for appropriate problems.

- Implications for SDT

Given the above descriptions of the design process described above, it is possible to identify some general human limitations which have direct impact on the ETES SDT. Table C-3 shows some of the major human limitations surrounding problem solving behavior and the general requirements they generate for the SDT.

C.2.2.2 Problem Solving and Design: Implications from Software Design Research

Recently there has been a good deal of research on the psychological processes underlying the software design process. Because the ETES SDT will essentially be a data base management system, an attempt was made to systematically review the literature in this area. Fortunately, a comprehensive review of the literature in this area has been completed by Ramsey and Atwood (1978). The details of this review are not repeated here. Section C.3 summarizes the problem solving stages that were identified by Atwood and Ramsey and their comments on the major human limitations surrounding each stage.

- Implications for ETES

In addition to identifying relevant stages in the design process and some of the major limitations of each stage, Atwood and Ramsey also identified some aiding mechanisms which could be used to overcome these limitations. Table C-4 displays these mechanisms and the implications these mechanisms have for the SDT.

Table C-3

HUMAN LIMITATIONS RELATED TO ETES

Human Limitation Relevant to ETES	Relevant Literature	SDT Requirements
Memory Limitations		
• Short term memory capacity is limited • Access of items in long term memory is limited due to storage/retrieval limitations		As much information as possible should be stored in systematic data bases which are congruent with existing human storage/retrieval frameworks.
Attention		
• Channel capacity is limited and subject to interference		SDT must have a systematic framework which users can utilize to selectively search for relevant information.
Generation and Evaluation of Alternatives		
• Deductive reasoning capability is limited		Detailed procedures must be supplied for "walking" the user through the use of SDT.
• Capacity for processing quantitative information is limited		
• Capability for making multi-attribute or multidimensional comparisons is limited		The SDT must be capable of interfacing with the training aids and simulation models which can assist in the detailed evaluative and quantitative tasks.

Table C-4 PROBLEM-SOLVING AIDS RELATED TO ETES*

Aiding Mechanism	Description	Comments	Principal References	Implications for SDT
Alternative Evaluation	These aids may either automate the user's evaluation criteria, require the user to use established criteria, or simulate the results of actions that do not have well established evaluation criteria.	Except for aids that automate the user's evaluation criteria, these task aids are task-specific. Most useful if the task is not well-defined or if a large number of evaluation criteria need be considered.	Brown et al (1975) Hormann (1967) Rapp (1972) Smith, H. T. and Crabtree (1975)	SDT must be able to feed into these alternative evaluation aids.
Alternative Generation	These aids are primarily used to generate alternatives that the user would not normally consider or, for extremely well-defined tasks, to present algorithmically determined alternatives.	Except for well-defined task domains, where they may have very little impact, they are difficult to construct. Can be cost-effective for training applications, but generally are of limited use in complex problem-solving tasks.	Baldwin & Siklossy (1977) Gagliardi et al (1965)	SDT must generate data base information options for user whenever it is possible to do so. However, SDT will not be directly involved in generating design options.
Automatic Action Execution	Such aids permit the user to name the desired action without explicitly carrying out the steps involved in its execution.	Most useful when the results of applying an action do not impact subsequent problem-solving actions. If this is the case, the user may need sophisticated alternative evaluation heuristics.	Carlson & Hodgson (1977) Hanes & Gebhard (1966) Pulfer (1971)	SDT must be powerful enough to allow for automatic action executions of information storage and retrieval commands.
Automatic Takeover	This type of aid functions as an automated decision maker that is able to select alternative actions on the basis of prior observations of the human decision maker's behavior. Although allocation of control to this aid occurs automatically, whenever some criterion of correspondence between predicted and observed human behavior is reached, voluntary turnover of control is also possible.	Although demonstrated to be effective in some contexts (e.g., control tasks), the range of tasks in which this is appropriate is not well understood. User acceptance may be low and should be carefully examined.	Freedy et al (1972) Steeb & Freedy (1976)	SDT must be capable of generating automatic "help" queries when user makes major errors.
Backtracking	Such an aid allows the problem solver to "undo" the effects of recent actions and return to an earlier state of the problem-solving process without actually starting over.	Useful in tasks where it is possible to "undo" recent actions. Can improve performance at relatively little development cost.	Carlson & Hodgson (1977) Michie et al (1968) Teitelman (1972)	SDT must have capability of providing separate work spaces for each user.
Better Weighting of Unreliable Data	This aid re-codes low-fidelity data into a form that is more readily usable by the problem solver.	Depends on the ability to accurately recode low-fidelity data.	Topmiller (1968) Howell & Gettys (1968)	SDT must clearly distinguish between estimated and actual data.
Change of Problem Representation	Typical implementations of this aid present problems as isomorphic variations of more standard problem representations. It is intended that this will aid the problem solver in selecting an appropriate and efficient problem formulation.	Most useful in well-understood tasks. An inappropriate representation may seriously degrade performance.	Chesler & Turn (1967) Smith, H. T. (1974) Newsted & Wynne (1976)	SDT must be capable of accessing selected sets of information by a variety of different means.
Decision Consistency Improvement	This type of aid assists the users applying their own decision strategies consistently in cases in which these strategies are complex.	Useful for expert problem solvers in well-defined tasks. Including sufficient versatility to adapt to individual users may be difficult.	Davis et al (1975) Freedy et al (1976)	SDT must have systematic procedures for "walking" the user through its use.
Decision Strategy Improvement	Such aids assist the user in applying problem-solving techniques that would not normally be considered or known.	Useful in well-defined tasks in which optimal, or near optimal, problem-solving techniques are known, or in tasks in which general heuristics, such as problem reduction, are applicable. Requires detailed knowledge of the task.	Caruso (1970) Gagliardi et al (1965) Rogers et al (1964) Wolde (1969)	Not applicable to SDT
Decomposition and Recomposition	This type of aid allows the user to divide the original problem into sub-problems. The solutions of the various subproblems are then combined into a solution to the original, larger problem.	Useful only if a task can be decomposed into independent subproblems. Requires a good understanding of the task.	Krolak (1971)	SDT must have built in hierarchical structure for data items within selected files to permit decomposition and recomposition.
Disruption of Psychological Set	Such an aid is intended to disrupt any bias or "sets" that the user may employ and, thereby, stimulate more creative or novel problem-solving attempts.	Potentially useful, but may disrupt an appropriate "set."	Stewart (1976)	Not applicable to SDT

*Table derived from Ramsey and Atwood.

Table C-4 (continued)

Aiding Mechanism	Description	Comments	Principal Reference	Implications for SDT
Extended Memory	This aid allows the user to store and retrieve problem-relevant information. This information may initially be generated by the user or by other problem-solving aids, such as aids for alternative generation and evaluation.	Very useful in almost all tasks. Success is related to the ease of retrieval from external memory.	Balzer & Shirey (1968) Newsted & Wynne (1976) Smith, H. T. & Crabtree (1975)	SDT, being a data base management system, must have extensive extended memory capabilities.
Lockout	In an interactive problem-solving situation, this technique restricts the problem solver's access to the computer for some period of time after the presentation of the results from the current request for information.	Although demonstrated effective in some contexts user acceptance was low. The tradeoff between user acceptance should be carefully considered.	Boehm et al (1971) Seven et al (1971)	May be necessary in cases where two or more uses attempt to access and modify the same data simultaneously.
Rapid Trial-and-Error	This aid allows the user to rapidly and easily examine the consequences of alternative action by simulating their application.	Easily implemented in well defined tasks. May offset inadequacies in decision strategy improvement aids.	Balzer & Shirey (1968) Carlson & Hodgson (1977) Rapp (1972) Wilde (1969)	SDT must have sufficient diagnostics to allow inexperienced user some degree of trial and error

*Table derived from Ramsey and Atwood.

C.3 BIBLIOGRAPHY OF PSYCHOLOGICAL RESEARCH RELATED TO DESIGN

Tables C-5 and C-6 are tables taken from a report on human factors in computer systems by Ramsey and Atwood. The listed references can be found in a separately published annotated bibliography (Ramsey, Atwood & Kirschbaum, 1978). References marked with a single asterisk indicate reports of surveys, questionnaires or summarized data. A double asterisk indicates a report of performance data or detailed results of experimental studies. It should be kept in mind that the references are, for the most part, concerned with problem solving behavior as it relates to software design.

Table C-5
Basic Subtasks or Phases Involved in Problem Solving

Subtask or Phase	Description	Comments	Principal References
Problem Recognition	The first stage in problem solving is to recognize that a problem exists. People are frequently slow to recognize, or at least react to, problems. This is especially true in situations in which a person must monitor the current state of the environment and detect or react to critical changes.	A primary need is for an aid that alerts the problem solver to "relevant" changes in the environment. The relevant variables for a given task can be difficult to define. Current status displays, historical displays, and aids for dealing with degraded data can be useful. If the relevant variables are identified, coding techniques can be very useful.	Booth et al (1968) Chesler & Turn (1967) Scanlan (1975) Smith, R.L. et al (1972) Topmiller (196 Wylie et al (1975)
Problem Definition	After a problem is recognized, the problem solver must determine how to formulate, or represent, the problem. In most cases, there are several alternative formulations for a given problem. The overall success of problem solving strongly depends on selecting an appropriate formulation.	Aids that provide a change in problem representation (e.g., graphical displays, isomorphic representation) can be extremely useful. Developing alternative representations requires a thorough understanding of the specific problem and the problem-solving processes that are most appropriate. Allowing the problem solver to decompose the problem into subtasks and recombine these subtasks in various ways can be useful in problems with relatively independent tasks. This type of aid is less difficult to develop than changes in problem representation, but it is also less general.	Balzer & Shire (1968) Cushman (1972) Krolak et al (1971) Newsted & Wynn (1976)* Smith, H.T. (1974)* Stewart (1974)

Subtask or Phase	Description	Comments	Principal References
Goal Definition	In some cases, the goal to be achieved is predefined. In other cases (e.g., tactical planning), the problem solver must select an appropriate goal. A selected goal must be not only appropriate, but also attainable.	The primary difficulty is that a selected goal may not be attainable. It may be useful to aid the problem solver in generating several alternative, logically consistent goal structures and to delay selecting a specific goal until later in the problem-solving process. Research on goal definition is lacking.	None
Strategy Selection	Strategy selection is is concerned with determining the general approach that will be used in problem solving. In some cases, a certain strategy is dictated by the problem representation that is selected. In general, strategy selection is based on previous experience with a given class of related problems.	The majority of strategy-selection aids are concerned with specific problem domains. This is appropriate since strategy selection is strongly driven by experience in a given domain. In domains in which problems can be decomposed into fairly independent subproblems, aids that allow the user to select strategies for these subproblems independently before combining them into an overall strategy can be very useful. Additional research is needed on the nature of specific problem-solving tasks and the strategy selection heuristics used by expert problem solvers. This would enable the construction of techniques to aid the less experienced user in this phase of problem solving.	Bennett (1971) Caruso (1970)* Wilde (1969)*

Subtask or Phase	Description	Comments	Principal References
Alternative Generation	In well-defined tasks, the problem solver can usually generate all alternative actions that may be appropriate. If there is a large number of alternatives, however, the problem solver may not be able to retain all alternatives in memory for later evaluation. If the task is not well-defined, the problem solver may not be able to generate appropriate alternative actions.	Aids that store a large number of user-generated alternatives can easily be developed and can also be effective. The principal need is for aids to suggest alternatives that the user is unable to generate. Such aids have been developed for training applications and for cases in which the computer has been programmed to generate optimal solutions without explicit user interaction. For ill-defined task environments, aids that suggest hypotheses to be tested may aid in alternative generation. Although potentially very useful, such aids could be difficult to construct.	Brown et al (1974) Carlson & Hodgson (1977)* Gagliardi et a[l] (1965)* Hormann (1967)
Alternative Evaluation	Problem solvers are generally very good at evaluating alternatives in a manner consistent with their perception of the problem and the goal to be achieved. If the alternatives have far-reaching consequences or if they must be evaluated with respect to a large number of factors, the problem solver's memory and processing limitations may be exceeded.	In extremely well-defined task environments, aids have been developed that allow the user to simulate the consequences of various alternatives. Although they are very useful in specific cases, such aids have limited generality. Aids that capture the user's evaluation heuristics and then filter information to be consistent with these heuristics and sometimes even present alternatives considered to be optimal are especially useful when a large number of evaluation heuristics must be applied. This type of aid is both effective and general, but it requires a great deal of effort to implement.	Balzer & Shire[y] (1968) Brown et al (1975) Davis et al (1975)** Doutriaux (197[illegible] Freedy et al (1976)** Michie et al (1968)** Rapp (1972)* Smith, H.T. & Crabtree (1975)** Yntema & Clem (1965)*

Subtask or Phase	Description	Comments	Principal References
Alternative Selection & Execution	The last phase of problem solving is concerned with implementing the solution.	Automatic execution of user-specified actions can aid the user in interacting with the problem-solving environment. Aids that automatically take over the problem-solving process may also be useful, but they should be used with caution.	Bursky et al (1968) Freedy et a (1976)** Hanes & Gebha (1976)* Pulfer (1971)

Table C-6
Types of Problem-Solving Aids

Aiding Mechanism	Description	Comments	Principal References
Alternative Evaluation	These aids may either automate the user's evaluation criteria, require the user to use established criteria, or simulate the results of actions that do not have well established evaluation criteria.	Except for aids that automate the user's evaluation criteria, these task aids are task-specific. Most useful if the task is not well-defined or if a large number of evaluation criteria need be considered.	Brown et al (1975) Hormann (1967) Rapp (1972)* Smith, H.T. & Crabtree (1975)**
Alternative Generation	These aids are primarily used to generate alternatives that the user would not normally consider or, for extremely well-defined tasks, to present algorithmically determined alternatives.	Except for well-defined task domains, where they may have very little impact, they are difficult to construct. Can be cost-effective for training applications, but generally are of limited use in complex problem-solving tasks.	Baldwin & Siklossy (1977) Gagliardi et al (1965)**
Automatic Action Execution	Such aids permit the user to name the desired action without explicitly carrying out the steps involved in its execution.	Most useful when the results of applying an action do not impact subsequent problem-solving actions. If this is the case, the user may need sophisticated alternative evaluation heuristics.	Carlson & Hodgson (197[illegible] Hanes & Gebhard[illegible] (1966)* Pulfer (1971)
Automatic Takeover	This type of aid functions as an automated decision maker that is able to select alternative actions on the basis of prior observations of the human decision maker's behavior. Although allocation of control to this aid occurs automatically whenever some criterion of correspondence between predicted and observed human behavior is reached, voluntary turnover of control is also possible.	Although demonstrated to be effective in some contexts (e.g., control tasks), the range of tasks in which this is appropriate is not well understood. User acceptance may be low and should be carefully examined.	Freedy et al (1972) Steeb & Freedy (1976)*

Aiding Mechanism	Description	Comments	Principal References
Back-tracking	Such an aid allows the problem solver to "undo" the effects of recent actions and return to an earlier state of the problem-solving process without actually starting over.	Useful in tasks where it is possible to "undo" recent actions. Can improve performance at relatively little development cost.	Carlson & Hodgson (1977) Michie et al (1968)* Teitelman (1972)
Better Weighting of Unreliable Data	This aid re-codes low-fidelity data into a form that is more readily useable by the problem solver.	Depends on the ability to accurately recode low-fidelity data.	Topmiller (1968)** Howell & Gettys (1968)*
Change of Problem Representation	Typical implementations of this aid present problems as isomorphic variations of more standard problem representations. It is intended that this will aid the problem solver in selecting an appropriate and efficient problem formulation.	Most useful in well-understood tasks. An inappropriate representation may seriously degrade performance.	Chesler & Turn (1967) Smith, H.T. (1974)** Newsted & Wynne (1976)*
Decision Consistency Improvement	This type of aid assists the users applying their own decision strategies consistently in cases in which these strategies are complex.	Useful for expert problem solvers in well-defined tasks. Including sufficient versatility to adapt to individual users may be difficult.	Davis et al (1975)** Freedy et al (1976)**
Decision Strategy Improvement	Such aids assist the user in applying problem-solving techniques that would not normally be considered or known.	Useful in well-defined tasks in which optimal, or near optimal, problem-solving techniques are known, or in tasks in which general heuristics, such as problem reduction, are applicable. Requires detailed knowledge of the task.	Caruso (1970)* Gagliardi et al (1965)** Rogers et al (1964) Wilde (1969)*

Aiding Mechanism	Description	Comments	Principal References
Decomposition and Recomposition	This type of aid allows the user to divide the original problem into subproblems. The solutions of the various subproblems are then combined into a solution to the original, larger problem.	Useful only if a task can be decomposed into independent subproblems. Requires a good understanding of the task.	Krolak (1971)*
Disruption of Psychological Set	Such an aid is intended to disrupt any bias or "sets" that the user may employ and, thereby stimulate more creative or novel problem-solving attempts.	Potentially useful, but may disrupt an appropriate "set".	Stewart (1976)
Extended Memory	This aid allows the user to store and retrieve problem-relevant information. This information may initially be generated by the user or by other problem-solving aids, such as aids for alternative generation and evaluation.	Very useful in almost all tasks. Success is related to the ease of retrieval from external memory.	Balzer & Shirey (1968) Newsted & Wynne (1976)* Smith, H.T. & Crabtree (1975)**
Lockout	In an interactive problem-solving situation, this technique restricts the problem solver's access to the computer for some period of time after the presentation of the results from the current request for information.	Although demonstrated effective in some contexts user acceptance was low. The tradeoff between user performance and user acceptance should be carefully considered.	Boehm et al (1971)** Seven et al (1971)**
Rapid Trial-and-Error	This aid allows the user to rapidly and easily examine the consequences of alternative action by simulating their application.	Easily implemented in well defined tasks. May offset inadequacies in decision strategy improvement aids.	Balzer & Shire (1968) Carlson & Hodgson (197 Rapp (1972)* Wilde (1969)*

Aiding Mechanism	Description	Comments	Principal References
Strategy Capture	These aids attempt to model and predict the user's behavior. Strategy capture is generally used in conjunction with other aids, such as automatic takeover or alternative evaluation.	A prerequisite for developing automatic takeover aids. Best suited to tasks that allow algorithmic, rather than heuristic, strategies.	Doutriaux (1973)* Freedy et al (1976)**

BIBLIOGRAPHY & REFERENCES

The items listed in Part I are those that were used as specific and general references for the main report. Those listed in Part II are the ones cited in Tables C-5 and C-6 in the Appendix of the main report. Annotations for these can be found in the report, "Annotated bibliography on human factors in software development" by Atwood et al.

Bibliography & References
Part I

1. Amkreutz, J.H.A.E. Cybernetic model of the design process. Computer Aided Design, 1976, 8(3), 187-192.

Computer aided design can be seen as a process in which a careful integration of the specific characteristics of man and machine is taking place. This integration may lead to better design results, quantitatively and/or qualitatively. The realization of this integration in CAD systems necessitates a renewed and profound analysis of the design process. A model of the design process is developed against the background of the evolution of this process. To this end, cybernetics is used as a meta-language. The development of CAD systems based on this model is discussed.

2. Askren, W.B. Human resources as engineering design criteria. AFHRL-TR-76-1, Brooks AFB, Texas: AF Human Resources Laboratory, March 1976.

Summarizes the results of a number of studies which have been performed in an attempt to develop a technology for using human resources data as criteria in engineering design studies. Eight investigations conducted during the period 1966-1975 are briefly described. The results of the eight studies are integrated around the six topics of: feasibility and practicality of using human resources data as criteria in engineering design, methods for using the data in design studies, effect on the system of using the data as design criteria, types of human resources data most relevant for use as design criteria, methods for generating human resources data for use in design studies, and nature of the engineering design process.

3. Atwood, M.E. et al. Annotated bibliography on human factors in software development. ARI Tech Rep P-79-1. Englewood, Colorado: Science Applications, Inc., June 1979.

As part of a larger Army Research Institute effort to survey, synthesize, and evaluate the state of the art in the area of human factors as applied to software development, a fairly extensive literature survey was conducted. This resulting bibliography contains citations of 478 articles or reports pertaining to the behavioral aspects of software design, programming, coding, debugging, testing, evaluation, and maintenance. Most citations are accompanied by descriptive abstracts, and all are indexed by author, publication source, institutional affiliation, and subject. To help the user unfamiliar with the area, the bibliography contains brief, basic reference lists in the areas of software engineering, the psychology of software development, the Structured Programming Series, and the DoD software program. Coverage is exhaustive through 1977 with a few references from 1978.

4. Atwood, M.E. et al. An exploratory study of the cognitive structures underlying the comprehension of software design problems. Tech Rep 392, Englewood, Colorado: Science Applications, Inc., July 1979.

An experiment was conducted to evaluate a framework for the study of software complexity and comprehension. Basic to this framework is the concept that a person's knowledge of, and experience with, software design affects that person's ability to comprehend a software problem and its potential solutions. Past research on software complexity and comprehensibility has largely been based on the assumption that complexity is a function of surface properties, such as variable names and flow of control. Such measures, however, ignore the effects of experience.

Research on expert-novice differences in problem-solving suggests that experts possess a large number of previously developed knowledge structures, or schemata, that can be used to understand or solve the current problem. Research on text comprehension provides theoretical concepts and experimental paradigms that are useful in determining the structure and content of these experience-related schemata.

An experiment examined the knowledge structures used by participants at differing levels of experience in comprehending software system specifications. Six participants, at each of five levels, studied a software system specification and then summarized both the presented specification and the probable form of the corresponding software design. The results indicate that software designers use previously learned schemata in understanding a software design problem and in actually constructing a design, and that these schemata differ as a function of experience. In addition, the structure and content of these schemata were investigated. It is suggested that by determining the structure and content of such schemata, software complexity and comprehensibility can be considered in a more meaningful manner.

5. Bruce, B.C. Case systems for natural language. BBN Rep No. 3010. Cambridge, MA: Bolt, Beranek and Newman, Inc., April 1975.

In many languages (e.g., Latin, Greek, Russian, Turkish, German) the relationship of a noun phrase to the rest of a sentence is indicated by altered forms of the noun. The possible relationships are called (surface) "cases". Because (1) it is difficult to specify semantic-free selection rules for the cases, and (2) related phenomena based on prepositions or word order appear in apparently case-less languages, many have argued that studies of cases should focus on meaning, i.e. on "deep cases".

Deep cases bear a close relationship to the modifiers of a concept. In fact, one could consider a deep case to be a special or distinguishing modifier. Several criteria for recognizing deep cases are considered here in the context of the problem of describing an event. Unfortunately, none of the criteria serves as a completely adequate decision procedure. A notion based on the context-dependent "importance" of a relation appears as useful as any rule for selecting deep cases.

A representative sample of proposed case systems is examined. Issues such as surface *versus* deep *versus* conceptual levels of cases, and the efficiency of the representations implicit in case systems are also discussed.

6. Dzida, W.; Herda, S. & Itzfeldt, W.D. User-perceived quality of interactive systems. *IEEE Transactions on software engineering*, 1978, SE4(4), 270-276.

User-perceived quality of interactive systems is defined in terms of statistically nonoverlapping categories, so-called dimensions or factors. Categories are identified by factor analysis and represent a dimensional concept of the quality of interactive systems as perceived by its users. Each category describes essential user requirements.

This paper reports on a method and some initial results in the analysis of user-perceived quality of interactive systems. It is based on research work described in more detail elsewhere.

The methodology of approach is suitable for software requirements definition and human factors engineering.

7. Foley, J.D. & Wallace, V.L. The art of natural graphic man-machine conversation. In *Proceedings of the IEEE*, April 1974, 62(4), 462-471.

The design of interactive graphic systems whose aim is good symbiosis between man and machine involves numerous factors. Many of those factors can be judged from the perspective of natural spoken conversation between two people.

Guiding rules and principles for design of such systems are presented as a framework for a survey of design techniques for man-machine conversation. Attention is especially focused on ideas of action syntax structuring, logical equivalences among action devices, and avoidance of psychological blocks to communication.

8. Gannon, J.D. An experiment for the evaluation of language features. *Int. J. man-machine studies*, 1976, 8, 61-73.

Recently a number of experiments have been performed whose aim was to compare programming language features to determine which programming language features programmers found difficult to use. This paper examines these experiments in light of the evidence that programming language designers would find most useful. A new experiment is described and applied to the problem of whether the assignment operation should be defined as an operator or a statement designator. Empirical evidence in the form of errors made by students programming solutions to two good-sized problems is presented favoring the use of assignment as a statement. Finally, the shortcomings of the new experiment are discussed.

9. Greeno, J.G. The structure of memory and the process of solving problems. In R.L. Solo (Ed.), Contemporary issues in cognitive psychology: The Loyola Symposium. Washington: Winston/Wiley, 1973, 103-133.

This paper explores the role of knowledge structures and memory in problem solving. A brief review of past approaches to problem solving and the degree to which the approaches involved the role of memory is presented. Problem solving is viewed as the process of transforming the initial situation (or given variables) into the desired situation (or unknown variables). Both givens and the desired situation can differ from problem to problem with regard to how well they are specified. Productive thinking in problem solving is discussed in terms of a 3-factor theory and some related research studies are briefly reviewed.

10. Halpern, M. Foundations of the case for natural-language programming. IEEE Spectrum, March 1967, 4(3), 140-149.

The purpose of this paper is to clarify some of the misconceptions that impede useful discussion of the question of the suitability of natural language for programming. It is argued that: (1) Natural-language programming is an attempt to put nonprogrammers in a closer relation with the computer, (2) A natural programming language must be able to be written easily, not just read easily, (3) Processing natural language is qualitatively different from (and faster than) translating one language to another, (4) The redundancy of natural language is an advantage rather than a disadvantage, and (5) Natural language programming will help bridge the man-machine communication gap.

11. Hill, I.D. Wouldn't it be nice if we could write computer programs in ordinary English - or would it? The Computer Bulletin, June 1972, 306-312.

One argument that is frequently made in favor of natural-language programming is that people should be able to communicate with computers in the same way that they communicate with each other. While it is desirable to have a common mode of communication, this does not imply that we need to teach computers English; an alternative is to teach people to communicate with each other through unambiguous instructions. This paper will consider the intricacies in natural English that prohibit natural language and simultaneously illustrate the need for people to use programming languages in their interactions with others.

12. Hoc, J.M. Role of mental representation in learning a programming language. Int. J. man-machine studies, 1977, 9, 87-105.

A theoretical framework has been defined to elucidate the problems raised in the training of analyst-programmers, and a beginning made in validating it in a preliminary experiment. This experiment showed that a programming language is progressively interiorized by a subject in the form of a "Systeme de Representation et de Traitement" (S.R.T.) or "Representation and Processing System", in which the experienced programmer can analyze problems. Prior to this, however, he must have made his analysis in other S.R.T. that are more or less compatible with the programming language concerned. Nineteen subjects of various levels of training were made to construct a COBOL flowchart of a Metro ticket-machine control problem. An analysis of errors was made and the strategies used described with the aid of 22 variables in order to determine the three principal steps involved in learning a programming language.

13. Klein, G.A. & Weitzenfeld, J. Improvement of skills for solving ill-defined problems. AFHRL-TR-78-31, Brooks AFB, Texas: AF Human Resources Laboratory, March 1979.

To develop effective programs for training people to solve general, commonly encountered, problems, it is necessary to recognize that such problems are typically ill-defined and require additional goal specification. Most current training programs have developed from information processing or from Deweyan theories of problem solving. However, none of these theories has provision for dealing with ill-defined problems. Current programs are therefore limited in their applicability. Solving ill-defined problems can be described in terms of two interacting processes: identifying the properties of the goal, and simultaneously attempting to find procedures for accomplishing the goal. Within this framework, goal specification is supported by the inference of goal properties from analogous problems, and by the use of unsuccessful procedures for inferring goal properties. This description of how people solve ill-defined problems was used to develop a number of implications for training programs aimed at improving problem solving abilities, such as the need to train personnel to specify goal properties initially and also continually throughout the process, special opportunities for using unsuccessful hypotheses as a source of goal properties, and the value of analogies for suggesting goal properties.

14. Meister, D. & Farr, D.E. The methodology of control panel design. AMRL-TR-66-28, Wright Patterson AFB, Ohio: Aerospace Medical Research Laboratories, Sept. 1966.

Nine control panel drawings were developed by designers using standard design criteria from a designer's guide. The drawings were then evaluated by five experts representing the disciplines of human factors, industrial design, maintainability and reliability engineering. Sample panels were mocked up and subjects were tested in operational use of these panels. The major results of the overall study were that (1) Designers manifest a high degree of variability in developing control panel drawings even when presented with a standard package of design information; (2) human engineering design criteria appear to be significant only in relation to anticipated operator performance characteristics, and difficulties in applying these criteria stem from lack of empirical knowledge of these relationships; (3) a major source of difficulty in securing the application of human engineering design criteria by designers is the latters' lack of a system-behavioral approach to design. The major need in the control panel design area is empirical research to refine and standardize simple and quickly applied evaluation techniques. More information is needed concerning the manner in which designers utilize human factors and other design inputs.

15. Meldman, J.A. A new technique for modeling the behavior of man-machine information systems. Sloan Management Review, Spring 1977, 29-46.

A serious problem in understanding or designing man-machine systems is the lack of powerful formal techniques for modeling or describing man-machine interactions. This paper focuses on man-machine interactions in management information systems. A management information system has four crucial characteristics that complicate modeling -- a large number of interacting subsystems, highly parallel behavior, asynchronous coordination of subsystems, and alternative behavior of subsystems. It is suggested that Petri Nets offer a technique for modeling that is formal and explicit, highly modular, and comprehensive, and can aid in better understanding man-machine interactions.

16. Miller, G.A.; Galanter, E. & Pribram, K.H. Plans and the structure of Behavior. Holt, Rinehart & Winston, 1960.

This book is considered a landmark in the evolution of the study of cognition and behavior. It essentially discusses the notion that some more organized control than simple stimulus-induced activation is necessary to account for information processing capacities. It discusses the cognitive control processes - plans - that guide behavior. The book presents a discussion of plans - schemata - for all aspects of behavior, such as motor skills and habits, remembering, speaking, etc.

17. Miller, L.A. Programming by non-programmers. Int. J. man-machine studies, 1974, 6, 237-260.

Non-programmers were asked to organize natural English commands of a laboratory programming language into programs for solving name-sorting problems. The problems differed in the sort concept to be programmed (conjunction vs. disjunction) and in the form of expression of the letter tests to be made on the names (affirmation vs. negation).

Programming performance was found to be impaired with disjunctive concepts and with letter tests involving negation. Different classes of program structure were identified and were associated with certain problem conditions and error measures. An influence of prior experience with procedures on performance was suggested. Program debugging and testing performance was characterized.

18. Miller, R.B. Psychology for a man-machine problem-solving system. TR 00.1246, Poughkeepsie, NY: IBM Corp., Feb., 1965.

This paper deals with the use of computer capabilities to extend human capabilities for invention and discovery. A programmatic route will be proposed for development. The first stage in this route will be an analytic enumeration of human abilities and liabilities as a problem-solving mechanism. The second stage will deal with an analysis of human information-handling tasks. These two stages should illuminate system objectives, while at the same time options for designing the man-machine problem-solving entity become clarified. The result will be an intelligence-retrieval system combined with logical and extraordinary display capabilities. The principal design issues will be revealed as indexing content and structure and display symbologies. An important (and neglected) dimension in system design is the human's ability to learn and think in new languages and symbologies.

19. Miller, R.B. Archetypes in man-computer problem solving. Ergonomics, 1969, 12(4), 559-581.

Information systems applied to operational environments have meaning only in what they do for humans performing tasks, whether clerical, technical or managerial. Each person's job-position entails interaction with a limited set of categories of variable data. By "limited" is meant less than several thousand, and more likely several hundred, categories. A category set associated with a collection of tasks performed by an individual or an organization may be called a category domain. This concept makes possible a practicable (in size) data base responsive to support human tasks in human (psychological) time.

An analysis of human problem-solving tasks reveals the following types: simple inquiry and update, status inquiry, briefing, exception detection, diagnosis, planning/choosing,

evaluating/optimizing, constructing (designing), and discovery. There is no compulsive ordering of these on a complexity scale. The information processing structure of each is examined: some common denominators among this set reveal five underlying archetypes of interaction. By making these archetypes explicit and consistent with concepts of domain, application disciplines and system design can move in parallel and generate a simple, well-defined language structure between system and human user.

20. Mitchell, T.R. Uncertainty and decision making. Tech Rep 79-19, Seattle, WA: Univ. of Washington (Dept. of Psychology), Sept. 1979.

This final report on 3+ years of research reviews studies in the causes of uncertainty, theoretical developments in uncertainty, the consequences of uncertainty and the applications of theory to uncertain situations. Further, work on a contingency model for selection of decision strategies is outlined and related research is described.

21. Norman, D.A. Memory, knowledge and the answering of questions. In Solso, R. (Ed.) *Contemporary issues in cognitive psychology: The Loyola Symposium*. Winston/Wiley, 1973, 135-165.

Discusses the nature of memory, concluding that the stored representation of knowledge cannot be separated from the uses to which the knowledge is put. Consideration is given to the learning process, emphasizing the necessary interaction between learner and instructor. The learner must be questioned to see what information is lacking; the information is then provided; and then requestioning must occur to evaluate success of the desired information transfer. This, with related considerations, leads to the formulation of a theory of instruction which highlights the importance of properly connecting new material into the framework provided by previous information storage. A formal structure for representing semantic information is presented, with examples, and a test of the structure utilizing simulation on a digital computer is described.

22. Posner, M.I. Cognition: Natural and artificial. In Solso, R. (Ed.) *Contemporary issues in cognitive psychology: The Loyola Symposium*. Winston/Wiley, 1973, 167-174.

This paper outlines some of the relationships among the papers included in the Solso edited volume, and compares current cognitive psychology with that of 100 years ago. It is pointed out that whereas psychologists tend to split the world into perceptual and linguistic domains, scientists working with artificial intelligence rely upon similar programs to handle both. This is seen as possibly suggesting a fundamental difference between artificial and natural intelligence.

23. Ramsey, H.R. & Atwood, M.E. Human factors in computer systems: a review of the literature. Tech Rep SAI-79-111-DEN. Englewood, Colorado: Science Applications, Inc., Sept. 1979.

Based on an extensive literature survey, this document presents a description and critical analysis of the state of the art in the area of human factors in computer systems. This review is concerned both with the status of human factors research in the area of user-computer interaction and with the current state of user-computer interaction technology and practices. The primary purpose of the review is to determine whether research and practice in this area have evolved sufficiently to support the development of a human factors guide to computer system design. It is concluded that insufficient data exist for the development of a "quantitative reference handbook" in this area, but that a "human factors design guide" -- which discusses issues, alternatives, and methods in the context of the design process -- is both feasible and needed.

24. Ramsey, H.R.; Atwood, M.E. & Campbell, G.D. An analysis of software design methodologies. Tech Rep 401. Englewood, Colorado: Science Applications, Inc., Aug. 1979.

Four formal software design methodologies were described and briefly analyzed: (1) Structured Design, (2) Jackson's Methodology, (3) Integrated Software Development System (Higher Order Software), and (4) Warnier's "Logical Construction of Programs." Relative strengths and weaknesses and commonalities among the methods were identified, and human factors problem areas were analyzed.

Several major human factors deficiencies and problems were identified. Formal software design methods differ in terms of: Applicability to problems of different types, size or complexity; susceptibility to design errors; and constraints and limitations imposed on the software designer. Various methods limit the designer's ability to select an appropriate problem representation, prevent the designer from utilizing relevant knowledge and experience, or impose potentially significant information loads on the designer. Improvements in design methodologies require a better understanding of the problem-solving behavior of software designers; potential research topics in this area were identified.

25. Ramsey, H.R.; Atwood, M.E. & Kirshbaum, P.J. A critically annotated bibliography of the literature of human factors in computer systems. Tech Rep SAI-78-070-DEN. Englewood, Colorado: Science Applications, Inc., May 1978.

A very broad survey of the literature dealing with human factors in computer systems was performed. Included in the survey were books, journal articles, proceedings papers and institutional publications from the literatures of

psychology, human factors, and computer science. From the resulting list, 564 references were selected for inclusion in this bibliography. The references selected deal primarily with the human factors aspects of interactive computer systems, including hardware, software and procedures. The selection of references emphasizes experimental studies, but the bibliography also includes relevant descriptions of dialogue techniques, user requirements analysis methods, guidelines, and a variety of other relevant topics.

For each reference, a citation is previously included together with sufficient information to allow the reader to obtain a copy, a descriptive abstract and a critical annotation. An extensive subject index, as well as an author index and browsing aids, allow the users to locate those articles in which they are interested.

26. Ramsey, H.R.; Atwood, M.E. & Willoughby, J.K. Paper simulation techniques in user requirements analysis for interactive computer systems. In Proceedings of the Human Factors Society 23rd Annual Meeting. Santa Monica, CA: Human Factors Society, 1979.

This paper describes the use of a technique called "paper simulation" in the analysis of user requirements for interactive computer systems. In a paper simulation, the user solves the problems with the aid of a "computer", as in normal man-in-the-loop simulation. In this procedure, though, the computer does not exist, but is simulated by the experimenters. This allows simulated problem solving early in the design effort, and allows the properties and degree of structure of the system and its dialogue to be varied. The technique, and a method of analyzing the results, are illustrated with examples from a recent paper simulation exercise involving a Space Shuttle flight design task.

27. Schrenk, L.P. Aiding the decision maker - A decision process model. Ergonomics, 1969, 12(4), 543-557.

Despite an increasing capacity for automating various tasks, there continues to be a requirement for man to serve as the decision element in many complex systems. The complexity and far-reaching consequences of many decisions impels a concern for improving decision-making performance in man-machine systems. In this paper current knowledge regarding human decision behavior and methods for aiding this behavior are briefly reviewed. A tentative conceptual model of an idealized process of decision making is presented. This model, which is based on both empirical and theoretical research, contains three phases. These are (1) problem recognition, (2) problem diagnosis, and (3) action selection. The model is intended primarily to provide a guide to system designers in structuring decision tasks and a framework for organizing knowledge about decision-making behavior.

28. Shneiderman, B. & Shapiro, S.C. Toward a theory of encoded data structures and data translation. Int. J. of computer and information sciences, 1976, 5(1), 33-43.

Several models of data base systems have distinguished levels of abstraction ranging from the high-level entity set model down to the low-level physical device level. This paper presents a model for describing data encodings, an intermediate level which focuses on the relationship among data items as demonstrated by contiguity or by pointer connections. Multiple data encodings for a file are shown and transformation functions that describe the translation between data encodings are discussed.

29. Smith, S.L. Requirements definition and design guidelines for the man-machine interface in C^3 system acquisition. Rep M80-10. Bedford, MA: MITRE Corp., April 15, 1980.

This report is both a review of the state-of-the-art of man-machine interface (MMI) in C^3 systems and a proposal for the exploration of potential development and application of MMI design guidelines in Air Force C^3 systems acquisition. The report discusses requirements definition, including consideration of user/operator characteristics, the information handling requirements of people's jobs, and the functional capabilities that can be provided in the MMI. It then discusses the problem of developing specifications that will communicate functional requirements to the systems designer, giving a sample set of guidelines. The report further discusses the specific documentation of MMI design that will be needed.

30. Whalen, G.V. & Askren, W.B. Impact of design trade studies on system human resources. AFHRL-TR-74-89, Brooks AFB, Texas: AF Human Resources Laboratory, Dec. 1974.

This study was undertaken to accomplish two objectives. The first objective was to identify and classify the characteristics of conceptual design trade studies that have high potential impact on human resource requirements of Air Force weapon systems. The approach used was a case history review and analysis of 129 F-15 aircraft design trade studies. The analysis indicated that the avionics system demonstrated the greatest potential impact on human resources. It was also found that trade studies dealing with design alternatives that encompass widely different technologies have substantial impact on human resources. The types of human resources data (HRD) most influenced by alternative design options were maintenance task times and personnel costs. The second study objective was to determine the accuracy of using subjective estimates as a technique for deriving HRD impact of trade study options. Using only engineering information for six avionics subsystems from the conceptual design phase, Air

Force maintenance technicians made subjective estimates of the impact of the designs on selected HRD items. It was found that technicians made highly accurate estimates of the amount of time, the Air Force occupational specialty, the level of technical skill, and the number of personnel needed to perform field maintenance tasks.

31. Wickelgren, W.A. Cognitive psychology. Englewood Cliffs, NJ: Prentice-Hall, Inc., 1979.

The book attempts to summarize and synthesize knowledge about how the mind works within an integrated theoretical framework. The approach is a consistent combination of what the author considers to be the best parts of associative, structural-linguistic, and information processing approaches to the mind. The chapters cover such topics as: visual perception; spatial cognition and imagery; auditory perception; speech and reading; attention; short-term memory; associative long-term memory; retrieval; recall and recognition; semantic memory coding: concepts, propositions and schemata; semantic memory processes: retrieval, learning and understanding; and thinking: plans, inference, problem solving and creativity.

32. Woods, W.A. What's in a link: Foundations for semantic networks. In Bobrow, D.G. & Collins, A. (Eds.) Representation and understanding: Studies in cognitive science. Academic Press, 1975.

This paper explores the topic of semantic networks and the degree to which they are capable of representing knowledge. The chapter is organized into 3 major sections. The first discusses the concept of semantics, including misconceptions about semantics and the semantics of programming languages. The second section discusses semantic networks presenting requirements for semantic representation, and the approaches such as links and case representations. The third section explores various problems in knowledge representation.

Baldwin, J.T. & Siklossy, L. An unobtrusive computer monitor for multi-step problem solving. Int. J. of man-machine studies, 1977, 7, 349-362.

Balzer, R.M. & Shirey, R.W. The on-line firing squad simulator. Tech Rep RM-55-73-ARPA. Santa Monica, CA: Rand Corp., Aug. 1968.

Bennett, J.L. Spatial concepts as an organizing principle for interactive bibliographic search. In D.E. Walker (Ed.) Interactive bibliographic search: The user/computer interface. Montvale, NJ: AFIPS Press, 1971, 67-82.

Boehm, B.W. et al. Interactive problem-solving: An experimental study of "lockout" effects. AFIPS Conference Proceedings, 1971, 38, 205-210.

Booth, T.L. et al. Experimental investigations of man-machine processing of information (Vol. III). Tech Rep U417-68-098. Groton, CT: General Dynamics Corp., Oct. 1968.

Brown, J.S. et al. A sophisticated instructional environment. Tech Rep AFHRL-TR-74-93. Brooks AFB, Texas: Air Force Human Resources Laboratory, Dec. 1974.

Brown, R.V. et al. Decision analysis as an element in an operational decision aiding system (Phase II). Tech Rep 75-13. McLean, VA: Decisions and Designs, Inc., Nov. 1975.

Bursky, P. et al. A man-machine competitive game: A naval duel. Tech Rep 68-34. Philadelphia, PA: Univ. of Pennsylvania, Moore School of Electrical Eng., May 1968.

Carlson, P. & Hodgson, T.J. An interactive heuristic approach for scheduling a multi-resource constrained system. Rep No. 77-10. Gainesville, FL, Sept. 1977.

Caruso, D.E. Tutorial programs for operation of on-line retrieval systems. J. chem documentation, 1970, 10, 98-105.

Chesler, L. & Turn, R. Some aspects of man-computer communication in active monitoring of automated checkout. Tech Rep P-3522. Santa Monica, CA: Rand Corp., March 1967.

Cushman, R.H. TOFT: A method for electronic doodling and a first step towards the use of computers on ill-defined problems. In Proceedings of the 1972 International Conference on Cybernetics and Society. NY: Inst. of Electrical and Electronics Engineers, 1972, 157-162.

Davis, K.B. et al. Adaptive computer aiding in dynamic decision processes: An experimental study of aiding effectiveness. Tech Rep PIR-1016-75-70. Woodland Hills, CA: Perceptronics, Inc., May 1975.

Doutriaux, J. Human-computer process control: Better training and better performance for the printing industry. In Proceedings of the 17th Annual Meeting of the Human Factors Society. Santa Monica, CA: Human Factors Society, 1973, 111-115.

Freedy, A. et al. Interactive aspects of a man/learning system control team. In Proceedings of the 1972 International Conference on Cybernetics and Society. NY: IEEE, 1972, 135-140.

Freedy, A. et al. Adaptive computer aiding in dynamic decision processes: Methodology, evaluation, and applications. Tech Rep PFTR-1016-76-8/30. Woodland Hills, CA: Perceptronics, Inc., Aug. 1976.

Gagliardi, U.O. et al. Man-computer interactions in idealized tactical problem solving. Final Report. Contract NONR-3602(00). Darien, CT: Dunlap & Associates, Inc., May 1965.

Hanes, R.M. & Gebhard, J.W. The computer's role in command decision. U.S. Naval Institute Proceedings, Sept. 1966, 92(9), 60-68.

Hormann, A. Problem solving and learning by man-machine teams -- progress and planned investigations. Tech Rep SDC-TM-2311/008/00. Santa Monica, CA: System Development, Corp., July 1967.

Howell, W.C. & Gettys, C.F. Some principles for design of decision systems: A review of the final phase of research on a command-control system simulation. Tech Rep AMRL-TR-68-158. Wright Patterson AFB, Ohio: Aerospace Medical Research Laboratories, Nov. 1968.

Krolak, P. et al. A man-machine approach toward solving the traveling salesman problem. Communications of the ACM, 1971, 14, 327-334.

Michie, D. et al. A comparison of heuristic, interactive and unaided methods of solving a shortest-route problem. In D. Michie (Ed.) Machine Intelligence (Vol. 3). NY: American Elsevier, 1968, 245-255.

Newsted, P.R. & Wynne, B.E. Augmenting man's judgment with interactive computer systems. Int. J. of man-machine studies, 1976, 8, 29-59.

Pulfer, J.K. Man-machine interaction in creative applications. Int. J. man-machine studies, 1971, 3, 1-11.

Rapp, M.H. Man-machine interactive transit system planning. Socio-Economic sciences, 1972, 6, 95-123.

Rogers, C.A. et al. Computers, physicians and the diagnostic decision-making process. Human factors, 1964, 6, 459-464.

Scanlan, L.A. Visual time compression: Spatial and temporal cues. Human factors, 1975, 17, 337-345.

Seven, M.J. et al. A study of user behavior in problem-solving with an interactive computer. Rep No R-513-NASA. Santa Monica, CA: Rand Corp., April 1971.

Smith, H.T. Man-computer collaboration in the design process. Memo No 59. Sheffield, England: Dept. of Psychology, MRC Social and Applied Psychology Unit, Sept. 1974.

Smith, H.T. & Crabtree, R.G. Interactive planning: A study of computer aiding in the execution of a simulated scheduling task. Int. J. man-machine studies, 1975, 7, 213-231.

Smith, R.L. et al. Development of graphic area displays for ASW attack management simulation (2 vols.). Tech Rep C-1003. Burbank, CA: Ocean Technology, Inc., 1972.

Steeb, R. & Freedy, A. Man-machine interaction in adaptive remote systems. In Proceedings, IEEE International Conference on Cybernetics and Society, Nov. 1976. NY: IEEE, 1976, 727-731.

Stewart, T.F.M. Ergonomic aspects of man-computer problem solving. Applied ergonomics, 1974, 5, 209-212.

Stewart, T.F.M. Displays and the software interface. Applied ergonomics, 1976, 7, 137-146.

Teitelman, W. "Do what I mean": The programmer's assistant. Computers and automation, April 1972, 21(4), 8-11.

Topmiller, D.A. Mathematical models of human performance in man-machine systems. Tech Rep AMRL-TR-68-22. Wright Patterson AFB, Ohio: Aerospace Medical Research Laboratories, May 1968.

Wilde, D.U. Iterative strategy design. Amer. documentation, 1969, 20, 90-91.

Wylie, C.D. et al. Toward a methodology for man-machine function allocation in the automation of surveillance systems. Vol I.: Summary. Tech Rep 1722-F, Vol. I. Goleta, CA: Human Factors Research, Inc., July 1975.

Yntema, D.B. & Clem, L. Telling a computer how to evaluate multidimensional situations. IEEE Transactions on Human Factors in Electronics, 1965, HFE-6, 3-13.

APPENDIX D

REVIEW OF RESEARCH RELATED TO HUMAN-COMPUTER INTERACTIONS

Appendix C reviewed psychological literature related to the engineering design process -- that is, the process that an individual engineer might go through in developing design concepts. In doing so the review focussed on the cognitive processes of the individual designer in developing system concepts. Appendix D reviews research relating to the weapon system design process, which is a more general process relating to the overall development of the system. The weapon system design process involves a large number of different individuals in several different disciplines who are scattered across many different Army and contractor organizations. As was noted, one of the major problems surrounding the weapon system design process is the communication and flow of information among the participants in this more general process. The ETES SDT is specifically designed to deal with these communication problems by providing a centralized, automated data base for describing and updating emerging system concepts and providing direct access to this data base to all participants in the weapon system design process. Thus, the SDT provides a systematic vehicle through which the participants in the weapon system design may communicate (see Figure D-1). To provide a foundation for SDT development this appendix reviews research relating to the SDT's role as a data base management system and communication tool.

The appendix is divided into two major sections. The first section describes the SDT requirements related to human-

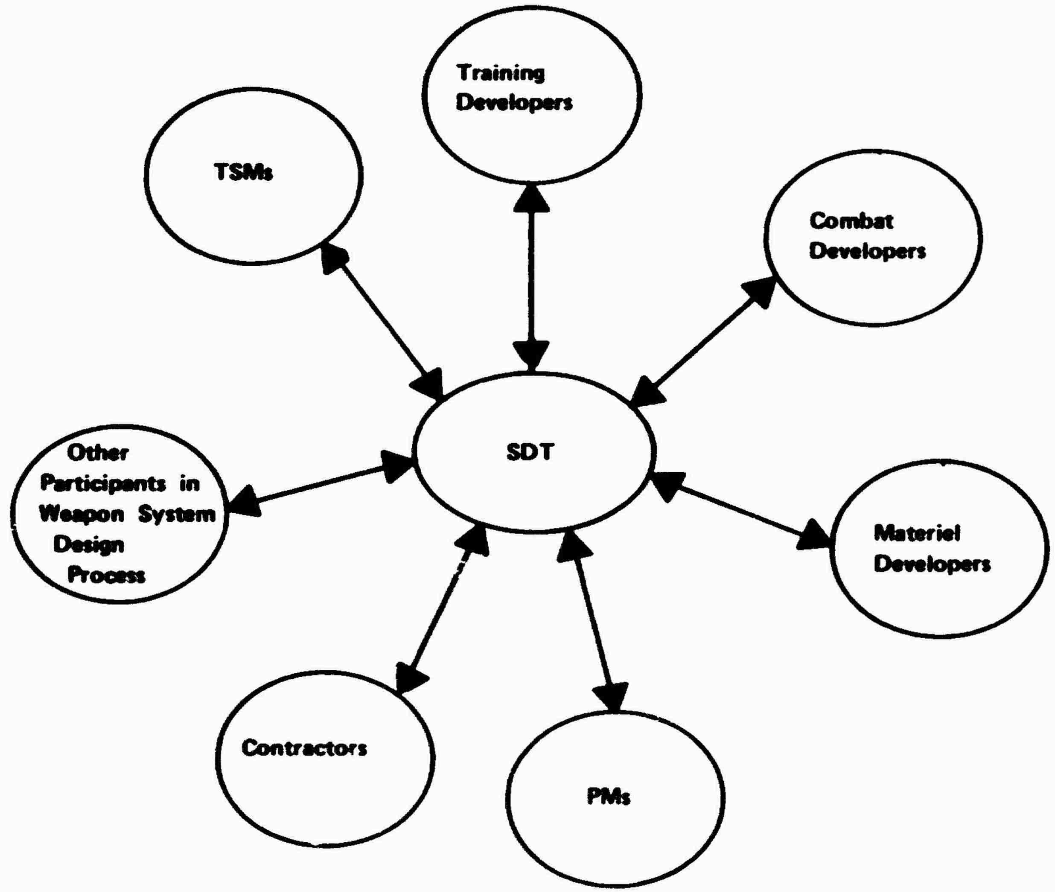


Figure D-1 CENTRAL COMMUNICATION ROLE OF SDT computer interactions. The second section reviews psychological research relating to the process of human-computer interactions and summarizes the implications that this literature has for the SDT.

D.1 SDT REQUIREMENTS RELATED TO HUMAN-COMPUTER INTERACTIONS

Section 1.0 reviewed the problems in information flow and communication which led to the initiation of the ETES project. Briefly, it was pointed out that training developers and other participants in the acquisition process are not systematically receiving information on early system concepts and are not being kept abreast of system changes and updates in a timely and systematic fashion. This lack of systematic communication makes it difficult, if not impossible, to effectively assess training and other human resources, and this lack of assessment, in turn, makes it difficult to effectively manage and control these resources.

In order for the SDT to fill these communication deficiencies, the SDT must itself be designed to facilitate easy and rapid communication with the personnel who will use it. Reviewing Appendix A, it is clear that the primary users of the SDT will be personnel from the staff of the training developers, combat developers, and materiel developers. These personnel are likely to have had little, if any, experience in utilizing computers or computerized data bases. Interviews with current commanders in these organizations indicates that there is also likely to be very little time or resources to train these personnel on ETES-related activities. Thus, it is imperative that the SDT be designed to (1) be utilized by uninitiated users who have no background in the use of computers and computer languages and (2) have minimal training requirements. Fortunately, in recent years there has been a growing body of literature on human-computer interactions and the types of interactions which are appropriate for uninitiated, minimally trained users. This literature is reviewed in the subsections which follow.

Before discussing this literature, it is important to point out that the systematic study of human-computer interactions is a relatively new area of research. Consequently, many of the guidelines discussed in the next section are only rational schemes for dealing with human-computer interactions -- empirical research to support these guidelines is generally not available. However, the conceptual schemes which have been developed do appear to have a high degree of face validity and should provide the necessary framework for the development of the SDT.

D.2 MAN-DATA BASE INTERACTION LITERATURE REVIEW

There have been four major efforts to survey and categorize literature relating to human-data base interactions.[1] More details on these four efforts is presented in the subsections which follow.

D.2.1 - Martin's Work on Interactive Dialogues

The first comprehensive work in human-computer interactions was conducted by Martin (1973) who documented his work in a

---

1 This research area is also described as the "man-machine interface" (MMI), as well as the human-computer interface or interaction.

book on human-computer dialogues. Martin's book was the first attempt to provide a systematic set of guidelines or human-computer interactions. Earlier work had focussed on the development of guidelines for computer input devices or output devices or computer programming practices but had not systematically covered dialogue or process-related questions.

Martin's basic approach toward conceptualizing the human-computer interactions was to divide human-computer interations into 18 basic dialogue types and to outline the advantages and disadvantages of each type in terms of types of users and information characteristics.[2] Table D-1 displays Martin's dialogues types and the estimated applicability to the SDT based upon Martin description of their advantages and disadvantages. As Table D-1 indicates, the most likely dialogues types for inclusion in the SDT are menu selection dialogues, form-filling, and question and answer dialogues. These were the dialogues which Martin indicated were (1) most appropriate for uninitiated users and (2) would not require extensive development costs or special terminals.

Martin also provides a series of guidelines to consider in selecting input and output devices. Table D-2 lists the input and output devices covered by Martin. To reduce implementation costs, it is desirable that the SDT utilize equipment that is currently available in ETES-related

---

2 Martin presents little empirical evidence to support his concepts (very little work has been done in this area). However, his concepts appear logical and seem to have a high degree of "face validity."

Table D-1

MARTIN'S DIALOGUE TYPES AND THEIR APPLICABILITY TO SDT*

1. Programming languages

2. English-language dialogue

3. Limit English input

*4. Question and answer dialogues (in which the computer asks the operator a series of questions)

5. Dialogue using mnemonics

6. Dialogue with programming-like statements

7. Computer-initiated dialogues (in which the operator responds to the computer rather than the computer responding to the operator)

*8. Form-filling (in which the operator fills out a "form" on a visual display)

*9. Menu-selection dialogues

10. Build dialogue features into special terminal hardware

11. Dialogues with a light pen for input (or other means of pointing to the screen)

12. Fixed-panel responses (in which the computer responds with one of a standard set of panels)

13. Modifiable-panel dialogues (in which the panels can be modified by the programs)

14. Graphics using chart displays

15. Graphics using symbol manipulation

16. Dialogues with photographic frames

17. Voice answerback dialogues

18. Dialogue via a third party

---

*Items applicable to SDT.

Table D-2

MARTIN'S CATEGORIZATION OF INPUT-OUTPUT DEVICES AND THEIR APPLICABILITY TO THE SDT

Input	Output
*Keyboard	*Typewriter or printer
Lever set or	Alphanumeric screen
Rotary switches	*Graphics screen
Push buttons	Screen displaying film frames
Light pen for point at screen	Light panel
Finger pointing at screen	Graph piotter
Stylus for drawing	Dials
Plate reader	Voice answerback
Badge reader	Facsimile machine

*Applicable to SDT.

organization. This requires that the SDT interactive input device mechanisms be restricted to a keyboard and output devices mechanisms be restricted to a printer and a graphics screen.

Finally, it is interesting to note that Martin suggests that in the future, "most dialogue programs will be generated with dialogue program generators rather than being programmed in a conventional language." These dialogue program generators are currently being used in computer-assisted instruction and are desirable because they reduce the programming that is needed for user-friendly dialogue. These types of programming techniques should also be considered during construction of the SDT.

D.2.2 - Ramsey and Atwood's Work on Human Factors in Computer Systems

Another major effort in systematically assessing human-computer interactions was directed by H. Rudy Ramsey and Michael Atwood in work sponsored by the Engineering Psychology Programs of the Office of Naval Research. Ramsey and Atwood (1979) have developed a conceptual scheme for classifying different areas of research relating to human-computer interactions. Table D-3 presents this scheme and also indicates which Atwood and Ramsey categories are most relevant to the SDT development. It is important to point out that input and output device questions are less relevant to the SDT because the SDT will utilize existing input/output devices and thus the interactive input device for the SDT is likely to be a keyboard and the interactive output devices are likely to be a printer and terminal

Table D-3

RAMSEY AND ATWOOD'S SCHEME FOR CLASSIFYING HUMAN-COMPUTER INTERACTION INFORMATION*

Description of Ramsey and Atwood Categories	Applicability to SDT
Users: their behavior in general; how to determine the properties of a particular user population; the implications of those properties for the interactive system.	x
Tasks: what tasks users perform; how to determine tasks involved in an application.	x
Requirements analysis: how to analyze information requirements; how to select appropriate types of problem-solving, clerical and user support aids; allocation of basic tasks to user or computer; modeling of user-system interactions; evaluation of basic design.	
Interactive dialogue: properties of different dialogue types; selection of appropriate dialogue type(s); detailed design of command language, system access structures, tutorial aids, etc.	x
Output devices and techniques: properties of display devices; implications of dialogue method for display device selection; selection or design of display device(s); detailed display design, formatting, coding techniques, etc.	x
Input devices and techniques: properties of input devices; implications of dialogue methods for input device selection; selection or design of input device(s).	
Evaluation of system performance: use of subjective evaluations, objective performance measures.	

*Table derived from Ramsey and Atwood (1979) display. In the sections which follow, more details on these categories of research are reviewed.

- User Characteristics

In discussing user characteristics Ramsey and Atwood review several past articles which have dealt with the requirements and/or capabilities of uninitiated users. (e.g., Card et al, 1974; Eason, et al 1975; Evans, 1976; Martin, 1973; Nickerson and Pew, 1971; and Thompson, 1971). Atwood and Ramsey indicate that interactions by these users can be facilitated if the computer-initiated or natural language dialogues are used, and they point out that natural language dialogues are very expensive to develop.

Atwood and Ramsey also discuss a procedure developed by Nawrocki, et al (1973) for conducting an automated error analysis. The error analysis can provide the means for improving system performance and might be useful during SDT implementation.[3]

- Tasks

Ramsey and Atwood's discussion of human-computer tasks centers on the development of task taxonomies which can be applied to computer-related tasks. This issue can be subsumed under the more general issue of the development of a task taxonomy applicable to all weapon system behaviors. This issue will be examined in later phases of SDT

---

3 However, as Strub (1975) has pointed out many errors are not detectable by such automated techniques.

development. Hence Ramsey and Atwood review of this area, which actually does little in the way of suggesting a possible taxonomy, is not reviewed here.

- Requirements Analysis

A discussion of requirements analysis tools is presented in Section 3. Hence, Ramsey and Atwood's discussion is not repeated here.

- Interactive Dialogue

As was noted earlier, Ramsey and Atwood indicate that computer-initiated dialogue would seem to a much more effective means of communication with uninitiated users, who are exactly the type of users which will utilize the SDT.

Ramsey and Atwood point out that computer-initiated dialogue has several advantages. First, this approach to dialogue allows the system to rely on the passive vocabulary of the user (the set of words which the user can recognize and understand), which is typically much larger than the user's active vocabulary (words which the user can generate and use without prompting). Second, it allows the designer to implicitly convey to the user a "mental model" of the system's dialogue structure. The major disadvantage of computer-initiated dialogue is the frequent delay it may produce for experienced users.

Ramsey and Atwood also present a scheme for classifying different types of interactive dialogues, and list the advantages and disadvantages of each type. A summary of this discussion is presented in Table D-4. Two types of

Table D-4

RAMSEY AND ATWOOD'S SCHEME FOR CLASSIFYING DIALOGUE TYPES*

Dialogue Type	Description	Appropriateness for SDT
Question-and-Answer	Computer asks questions to which user responds.	Recommended for naive user. Hence, appears to be applicable to SDT.
Form-filling	Computer presents form with blanks. User fills in blanks.	Recommended for naive user. Hence, appears to be applicable to SDT. Faster than question-and-answer dialogue.
Menu Selection	Computer presents list of alternatives, and user selects one or more.	Recommended for naive user. Hence, appears to be applicable to SDT.
Function Keys	User indicates desired action by depressing keys, each of which represents a command, command modifier, or parameter value.	Not recommended for SDT since SDT will be required to use existing terminals.
User-initiated Command Language	User types commands, perhaps using mnemonic abbreviations.	Not acceptable because of emphasis on well-trained users.
Query Language	User inputs questions or data base access procedures to a data base system. System produces response or report.	May be applicable for several SDT functions relating to data retrieval.
Natural Language	Dialogue is conducted in user's natural language (e.g., English).	Not recommended for SDT because of extremely high cost involved and wide range of users using system.
Interactive Graphics	Generation of pictorial displays, ability of user to select displayed entities and spatial locations by pointing or similar nonverbal means.	Not recommended for SDT because of cost and equipment limitations.

*Table derived from Ramsey and Atwood (1979).

dialogue seem especially applicable to the SDT because of their emphasis on computer-initiated dialogue -- form-filling and menu-selection. Another dialogue type query languages-would appear to be relevant to the SDT because of its close ties to data base management systems.

Form-filling is often used in situations in which the user's input is dominated by parameter values, rather than commands. Many attributes involve this type of data. Hence, form-filling would appear to be particularly useful as a data input mechanism for attributes.

Menu selection is described by Ramsey and Atwood as the "archetype of computer-initiated dialogue." Unlike question and answer dialogue or form-filling, all of the items to be selected appear on the screen, and thus the user need only recognize the desired action. Also, a simple menu-selection dialogue ordinarily requires only one user input (on the keyboard or screen), rather than, for example, the series of keystrokes required to type a whole word. Redsdale (1970) reports a study which documented the effectiveness of menu-selection with naive users. The study indicated that menu selection was especially effective when used as means of obtaining answers to a set of branching questions.

It is especially important to note that menu selection is a highly effective dialogue method for hierarchic search because of its reliance on the user's passive vocabulary and recognition memory. Hence, menu selection is particularly applicable to information retrieval (see Thompson, 1969, 1979 for a more extensive discussion of menu selection as a data retrieval mechanism).

Query languages are used to access an existing data base. According to Ramsey and Atwood, studies of specific query languages have generally concluded that existing query languages can be utilized by untrained users, particularly if the system in which the language is embedded emphasizes a computer-initiated approach. However, they point out that the error rate with query languages can be high. To reduce error, they suggest (1) utilizing a layered or portioned query language (that is, a query language which utilizes a lot of computer-initiated dialogue) (2) restating the user's command before execution.

- Input/Output Devices and Techniques

As noted above, the restriction of SDT hardware to computer equipment currently utilized by probable ETES users requires that SDT input devices be limited to the keyboards of existing terminals and SDT output devices be limited to printers and the display capabilities of existing terminals. Because of these restrictions, most of Ramsey and Atwood's discussion of input/output devices and techniques is not relevant. However, one area of research that is relevant is the discussion of techniques for coding information on CRT screens. The literature in this area is quite extensive and the interested reader should consult Ramsey and Atwood for a review. It should be noted that the requirement that the SDT be usable on a range of existing terminals severely restricts the type of coding that can be applied. Only very simple coding techniques (underlining, character size control, etc.) may be able to be usable on a wide range of terminals.

- Evaluation of System Performance

Ramsey and Atwood's discussion of the evaluation of human-computer systems is more related to implementation phase of the SDT than to the developmental phase. Hence, it is not reviewed here.

D.2.3 Smith's Work on Man-Machine Interface

Another major effort related to the assessment of human-computer interactions has been Sidney Smith's (1980) work on the development of guidelines for the man-machine interface in C^3 systems. This work was sponsored by the Air Force Electronic Systems Command.

Like Ramsey and Atwood, Smith (1980) has developed a scheme for categorizing topic areas related to human-computer interactions. Table D-5 displays Smith's schemes and the categories of Smith's work which have the most applicability to the SDT. Selected aspects of Smith's (1980) major report in this area which are relevant to ETES are reviewed below.

- Dialogue-Types

Smith, like many other investigators in this area, notes that computer-initiated dialogue types (e.g., form-filling, menu-selection) are more appropriate for uninitiated users than are user-initiated dialogues (e.g. programming languages). Table D-6 displays Smith's categorization of the different dialogue types and his estimation of user training and response time associated with each type. Based upon Smith's estimates, question and answer, form-filling and menu selection would seem to be the most appropriate

Table D-5

SMITH'S SCHEME FOR CLASSIFYING HUMAN-COMPUTER INTERACTION INFORMATION*

1.0 DIALOGUE TYPE
1.1 Question and Answer
1.2 Form Filling
1.3 Menu Selection
1.4 Function Keys
1.5 Command Language
1.6 Query Language
1.7 Natural Language
1.8 Graphic Interaction

2.0 DATA ENTRY/INPUT
2.1 Position Designation
2.2 Direction Designation
2.3 Data Type
2.4 Entry Formats
2.5 Data Validation
2.6 Data Processing

3.0 DATA DISPLAY/OUTPUT
3.1 Data Type
3.2 Data Density
3.3 Data Aggregation
3.4 Data Coding
3.5 Display Partitioning
3.6 Display Selection
3.7 Data Coverage
3.8 Display Update
3.9 Data Selection

4.0 SEQUENCE CONTROL
4.1 Transaction Selection
4.2 Interrupt
4.3 Context Definition
4.4 Error Management
4.5 Alarms

5.0 USER GUIDANCE
5.1 Status Information
5.2 Routine Feedback
5.3 Error Feedback
5.4 Instructional Aids

6.0 DATA TRANSMISSION/ COMMUNICATION
6.1 Data Transfer
6.2 Data Type
6.3 Transmission Control

*Table derived from Smith (1980).

Table D-6

SMITH'S SCHEME FOR CLASSIFYING DIALOGUE TYPES*

Dialogue Type	Required User Training	System Response Time
Question and Answer	Little/None	Moderate
Form Filling	Moderate/Little	Slow
Menu Selection	Little/None	Very Fast
Function Keys with Command Language	High/Moderate	Fast
User-Initiated Command Language	High	Fast
Query Languages	High/Moderate	Moderate
Natural-Language Dialogues	Moderate (potentially little)	Fast
Interactive Graphics	High	Very Fast

*Table taken from Smith (1980)

dialogue formats for the types of uninitiated users who will use the SDT (response time is not a critical issue for the SDT).

- Data Entry/Input

In discussing data entry/input topics, Smith mentions some general guidelines that should be considered in the construction of data entry mechanisms.[4] First, an operator should seldom be required to enter the same data twice or enter a data item already entered by another operator. He suggests that the way to do this is to program the computer to maintain context (that is, the computer should be able to access all data related to the user's input). Second, computer systems should be flexible. This means that the user should be able to set his own pace, cancel incomplete transactions, order inputs including temporary omission of unknown items, etc.

- Data Display/Output

Much of the material covered in Smith's discussion on display/output is redundant with Ramsey and Atwood's work. Hence, it is not repeated here.

- Sequence Control

Smith suggests that menu selection might be used as an effective means of providing sequence control for

---

4 He also has an appendix of more specific guidelines which merit careful consideration.

uninitiated users. He also indicates that flexibility is important in sequence control, particularly in interactions which involve the modification of stored data.

- User Guidance

Smith suggests that the fundamental rule in the area of user guidance is that for every action by the user there should be a response by the machine. Such feedback helps maintain user orientation.

D.2.4 Sidorsky and Parrish Work on Battlefield Automated Systems

A comprehensive assessment of human-computer interaction is currently being developed by Sidorsky and Parrish (1980) in work conducted for the Army Research Institute. The goal of the Sidorsky and Parrish work is to develop general guidelines for describing and designing battlefield automated systems so that ultimately the interoperability of such systems can be increased, and, consequently, user errors can be decreased. Like Smith (1979) and Ramsey and Atwood, Sidorsky and Parrish (1980) have developed a conceptual scheme for classifying information related to human-computer interactions (see Table D-7). They have also developed a method for systematically assessing the human-computer transactions which currently occur in battlefield systems.

The study by Sidorsky and Parrish was only begun recently, hence they have only developed guidelines in a few selected areas (e.g., data entry). Thus, their work provides little current guidance for the SDT development. However, Sidorsky

Table D-7

SIDORSKY'S AND PARRISH'S SCHEME FOR CLASSIFYING HUMAN-COMPUTER INTERACTION INFORMATION*

1. CONTROL METHODS
1.1 Command Languages
1.2 Menus
1.3 Function Keys
1.4 Hybrid Methods
1.5 Prompt/HELP

2. DISPLAY FORMAT
2.1 Fixed Alphanumeric Displays
2.2 Variable-Length Alphanumeric Displays
2.3 Graphic Displays
2.4 Highlighting

3. DATA ENTRY ASSISTANCE
3.1 Information on Legal Entries
3.2 Unburdening of Input
3.3 Interrupts and Work Recovery

4. MESSAGE COMPOSITION AIDS
4.1 System Design Features
4.2 Format for Alphanumeric Messages
4.3 Graphic Messages

5. DATA RETRIEVAL ASSISTANCE
5.1 Query Method
5.2 Query Structure

6. GLOSSARIES
6.1 Standard Terms
6.2 Character Sets and Labels
6.3 Glossary Availability and Use
6.4 Abbreviation and Coding

7. ERROR HANDLING
7.1 Prevention
7.2 Detection
7.3 Feedback
7.4 Correction/Recovery

8. USER/OPERATOR CONFIGURATIONS
8.1 Operator(s) Only
8.2 Operator(s) and User(s)
8.3 Combined Operator/User
8.4 Operator and User Chains

*Table derived from Sidorsky and Parrish (1980).

and Parrish's work will be closely monitored as the SDT is developed since it focusses directly on Army systems, and relevant guidelines developed in this study will be incorporated into the SDT.

One of the more developed areas of Sidorsky and Parrish which is relevant to the SDT work is the identification of techniques for highlighting (or coding) information on display terminals. Table D-8 lists the highlighting methods identified by Sidorsky and Parrish and indicates which of these highlighting methods is likely to be appropriate for the SDT given the general restriction that the SDT highlighting techniques must be applicable to a wide range of existing terminals.

D.3 SUMMARY OF IMPLICATIONS FOR SDT

Reviewing the literature related to man-computer interactions, it is possible to identify four general guidelines for the construction of the SDT.

First, in selecting the type of dialogue which is appropriate for the SDT, it is clear that some form of computer-initiated dialogue should be utilized given the types of uninitiated users who can be expected to employ the SDT. More specifically, it would appear that three dialogue types -- question and answer, form-filling and menu selection seem most appropriate for the SDT. A fourth type of dialogue, query languages, might be selectively used by the SDT data base maintainers and other more sophisticated users but probably is not adequate for general use.

Table D-8

SIDORSKY'S AND PARRISH'S SCHEME FOR HIGHLIGHTING INFORMATION

Highlighting Method	Methods Applicable to SDT
Brightness Control	
Character Size Control	x
All Upper Case	x
Reverse Display	
Underlining	x
Different Font	
Color Control	
Blinking, Pulsating	
Boxing	x
Arrowing	x
Symbolic Tagging	x
Alphanumeric Tagging	x
Position Displacement	x

Second, because most of the users of the SDT are not likely to have the resources to purchase ETES-specific equipment, the SDT must be compatible with the input and output devices which are currently available in the organizations which will utilize the SDT. This means that the SDT interactive input device probably must be restricted to a keyboard (via existing terminals) and the SDT output devices must be restricted to printers and display terminals which are also available on these terminals.

Third, coding of information on the SDT displays should be developed in accordance with the most current guidelines in Smith (1979), Sidorsky and Parrish (1980), and Ramsey and Atwood (1980). However, any coding schemes which are developed must be general enough to apply to a wide range of terminals. This would restrict coding to a small number of available techniques (i.e., underlining, italics, character size control, position displacement, alphanumeric tagging, symbolic tagging, and arrowing).

Fourth, it is recommended that the SDT study team continue to review the three on-going efforts of Smith, Sidorsky and Parrish, and Ramsey and Atwood so that newly developed guidelines can be incorporated into the SDT whenever it is possible to do so.