

R-115

IMPLEMENTATION OF COMPUTER SOFTWARE TECHNIQUES TO HUMAN FACTORS TASK DATA HANDLING PROBLEMS

A. T. TULLEY
G. R. MEYER

SYSTEM DEVELOPMENT CORPORATION

SEPTEMBER 1967

JOINT NASA/USAF STUDY



Distribution of this document is unlimited. It may be released to the Clearinghouse, Department of Commerce, for sale to the general public.

AEROSPACE MEDICAL RESEARCH LABORATORIES
AEROSPACE MEDICAL DIVISION
AIR FORCE SYSTEMS COMMAND
WRIGHT-PATTERSON AIR FORCE BASE, OHIO

N68-11855

(ACCESSION NUMBER)	(THRU)
98	1
(PAGES)	(CODE)
OK 49525	05
(NASA CR OR TMX OR AD NUMBER)	(CATEGORY)

GPO PRICE \$	
CFSTI PRICE(S) \$	
Hard copy (HC)	2.50
Microfiche (MF)	1.65

NOTICES

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

Federal Government agencies and their contractors registered with Defense Documentation Center (DDC) should direct requests for copies of this report to:

DDC
Cameron Station
Alexandria, Virginia 22314

Non-DDC users may purchase copies of this report from:

Chief, Storage and Dissemination Section
Clearinghouse for Federal Scientific & Technical Information (CFSTI)
Sills Building
5285 Port Royal Road
Springfield, Virginia 22151

Organizations and individuals receiving reports via the Aerospace Medical Research Laboratories' automatic mailing lists should submit the addressograph plate stamp on the report envelope or refer to the code number when corresponding about change of address or cancellation.

Do not return this copy. Retain or destroy.

**IMPLEMENTATION OF COMPUTER SOFTWARE
TECHNIQUES TO HUMAN FACTORS TASK
DATA HANDLING PROBLEMS**

A. T. TULLEY

G. R. MEYER

Distribution of this document is unlimited. It may be released to the Clearinghouse, Department of Commerce, for sale to the general public.

FOREWORD

This report presents the findings of a research effort sponsored jointly by the United States Air Force and the National Aeronautics and Space Administration. The report was prepared by the Dayton, Ohio facility of System Development Corporation for the Behavioral Sciences Laboratory, Aerospace Medical Research Laboratories, Wright-Patterson Air Force Base, Ohio. Mr. A. T. Tulley was the principal investigator. The research was conducted under Contract F33615-67-C-1036 during the period of 21 June 1966 through 30 June 1967.

This research was conducted in support of Project 1710, "Human Factors in the Design of Training Systems," and Task 171006, "Personnel, Training, and Manning Factors in the Conception and Design of Aerospace Systems." Dr. Gordon Eckstrand was the Project Scientist and Mr. Melvin Snyder was the Task Scientist. Mr. Lawrence E. Reed served as the contract technical monitor.

Appreciation is extended to Dr. Stanley Deutsch, National Aeronautics and Space Administration, Washington, D. C., who served as the NASA initiator and monitor, for reviewing the preliminary draft manuscript.

The authors wish to thank the many government and contractor personnel who contributed valuable suggestions throughout the course of this research and who helped review the draft manuscript.

Special thanks are due Mr. Melvin Snyder and Dr. Gordon Eckstrand, who assisted in the review of the manuscript and Mrs. Joan C. Robinette for her suggestions and editorial assistance. Acknowledgement is also made of other System Development Corporation personnel: Mr. Robert G. Oller, Misses Sue E. Reardon and Phyllis J. Mitchell who worked on the research during the period covered, and Mr. Donald E. Blair who edited this report.

This technical report has been reviewed and is approved.

WALTER F. GREYER, PhD
Technical Director
Behavioral Sciences Laboratory
Aerospace Medical Research Laboratories

ABSTRACT

Research leading to the implementation of computer software techniques for handling human factors task data generated in support of aerospace system development programs is discussed. Techniques being explored in this research program are based on the assumption that a user-oriented computerized data system will help draw human factors specialists closer to needed data. The application of such a system will reduce the problem of data accessibility and allow more effective use of data in the system engineering process. Preliminary research leading to proposed data handling techniques is discussed. A computerized data handling system to store, retrieve, and process human factors task data is initially implemented through a pilot study. A discussion of the pilot study specification is followed by a presentation of the design specification for a computerized experimental system. The experimental system, referred to as the pilot study experimental system, provides the primary means for demonstrating and evaluating the research results against the original research goals. Computer software descriptions are presented for implementing the pilot study experimental system in a user-oriented environment in terms of information needs of human factors specialists.

TABLE OF CONTENTS

	<u>Page</u>
Section I: Research Process -----	1
Overview of the Problem -----	1
Research Goals -----	2
The Research Approach -----	2
Preliminary Research -----	2
Pilot Study -----	4
Derive Operational System Requirements -----	5
Derive Operational System Concept -----	6
Define Pilot Study Requirements -----	9
Selection of Research Areas -----	9
Conduct Research -----	10
Pilot Study Design, Test, and Evaluation -----	10
Develop Pilot Study Design Specifications -----	11
Identify Pilot Study Experimental System Design -----	11
Conduct Tests -----	11
Evaluation of Test Results and Preparation of Recommendations -----	11
Experimental Data Vehicle -----	13
Summary -----	13
Section II: Pilot Study Design Specifications -----	14
Organization of the Specifications -----	14
Design Specifications -----	16
Input -----	16
Generated Data -----	16
Data Categorization -----	16
Preparation of Data -----	17
Current Awareness -----	18
Retrieval -----	20
Output -----	20
Response to User Queries -----	20
Error Messages -----	21
Security/Proprietary Constraints -----	21
Output Mode -----	22
Current Awareness -----	23
Update -----	23
Add New Data to Current Data Base -----	24
Modify User Profiles -----	24
Purging of Data Base -----	25
Add New Data Elements of Items to Data Base -----	25
Add to or Modify Pilot Study Software -----	26
Analysis and Simulation Capabilities -----	27
Thesaurus -----	28

TABLE OF CONTENTS (CONT'D)

	<u>Page</u>
Environment -----	30
Computer Hardware -----	30
Computer Software -----	30
Other Software -----	33
Interface - Other Systems-----	33
Interface - Data Management -----	34
Research Testing and Evaluation-----	35
Guides-----	37
User's Guide-----	37
Controller's Guide -----	38
 Section III: Pilot Study Experimental System -----	 39
Introduction -----	39
Pilot Study Experimental System Design Specifications -----	41
Information Storage and Retrieval -----	43
Data Preparation-----	43
File Establishment -----	47
File Maintenance -----	47
Batch Updating -----	48
On-Line Updating -----	49
Data Retrieval-----	51
Current Awareness-----	51
Profile Preparation -----	52
Profile Maintenance-----	52
Data Notification -----	53
Analysis-----	54
 Section IV: Functional Description of LUCID System as Applied to Human	
Factors Task Data -----	56
Introduction -----	56
System Overview-----	57
Describing the Data Base-----	57
Loading the Data Base -----	59
Querying the Data Base -----	60
Updating the Data Base -----	61
TSS-LUCID Program Components -----	62
Translator Program-----	62
Load Program -----	64
Retrieval Program-----	66
Update Program -----	71
Merge Program -----	74
 Section V: Summary-----	 75

TABLE OF CONTENTS (CONT'D)

	<u>Page</u>
Appendix I: Definition of Data Elements -----	79
Appendix II: Definition of Data Items-----	80
Appendix III: Guidelines for Data Extraction (ALCC) -----	87
Appendix IV: Standard Input Data Form-----	89
References -----	91

LIST OF ILLUSTRATIONS

Figure	<u>Page</u>
1 Research Approach -----	3
2 Conceptualized Operating -----	7
3 Operation of Data Exchange Center -----	8
4 Total Processes and Products of a Computer System-----	12
5 Pilot Study Design Process -----	15
6 AN/FSQ-32 Equipment Configuration -----	31
7 Pilot Study Experimental System -----	40
8 Pilot Study Experimental System Design Areas -----	42
9 File Establishment Process -----	48
10 Batch Update Process -----	49
11 Retrieval and On-Line Updating Process-----	50
12 Current Awareness Process -----	55
13 TSS-LUCID Concordance -----	67
14 Sample Retrieval Request-----	72

LIST OF TABLES

Table I: Characteristics of AN/FSQ-32 and Related Equipment -----	32
---	----

SECTION I

RESEARCH PROCESS

OVERVIEW OF THE PROBLEM

Before the complex systems--Atlas ICBM System; SAGE, an air defense system; and F105, a tactical fighter system--that grew out of the developmental revolution spawned by World War II, man's capabilities or limitations were seldom considered in the design of an equipment item or assembly of integrated equipment items now called a system. When machine tool designers found that an operator couldn't reach the buttons or levers of the machine, they frequently told the operator to get a push-rod or tie a stick on the handle. The machine had been built, and man would have to adapt to the machine. When the airplane pilot found he couldn't step up into the aircraft, the designers gave him a step stool. When the pilot couldn't get into the cockpit with a parachute, the designers hung the parachute on the side of the plane. The machine was triumphant over man and man had to adapt.

With World War II, machines became much more complex and required more cooperative attention. If the machine's designers ignored man, the machine would not perform accurately or adequately. Man was finally recognized and specialists in man's capabilities or limitations--human factors specialists--were included in system design teams.

With the ever-increasing demands that modern aerospace systems place on man, the need for highly skilled personnel becomes critical. This need has been accompanied by an increasing demand for more and better data by system specialists. Concomitantly, the size, complexity, and compressed developmental schedules of modern systems have resulted in the generation of increasingly large amounts of data. This problem has led to: (1) decreased effectiveness of data in system design and development, (2) increased reliance on expertise when existing data are not known to exist or are inaccessible, (3) the generation of inadvertent duplication of data, and (4) the scattering of costly information.

In reporting the results of Project Forecast, General Schriever (Armed Forces Management, May 1965) emphasized that computer costs and effectiveness dictate that for the best use of current and future hardware, user-oriented languages and means for time sharing must be developed. By devising means to make more effective, direct use of the computer--through decreased dependency on the intermediary between the users and the machine--the user and generator of information will be drawn closer to the data. The problem of data accessibility presently plaguing the government and industry will be reduced and the data will be more effectively used in the design and development of

present and future aerospace systems. These statements govern the philosophy of the research discussed in this report.

RESEARCH GOALS

Means for alleviating the problems discussed in the preceding paragraphs are sought in response to the following research goals:

- Improve communications--Provide a means of having information where and when it is needed--This will include a current awareness service to the user.
- Reduce redundancy--Alleviate the problem of duplicative effort through a continuous exchange of information within a system development team and between systems
- Improve accessibility--Provide a common store of data from which the user may retrieve selectively
- Provide basic tools--Provide the user with a pool of analytic and simulation tools
- Provide dynamic data--Provide a data base which is current and frequently updated
- Standardization--Provide standardization of the language and procedures

THE RESEARCH APPROACH

Figure 1 is an illustration of research activities leading to the application and implementation of computer software techniques for handling human factors task data generated in support of aerospace system development programs. The blocks within the dashed lines represent the major research activities. These activities are grouped into three areas of study--preliminary research, pilot study, and pilot study design, test, and evaluation.

Preliminary Research

The objective of this research was to identify problem areas and to determine the feasibility of developing improved methods for handling human factors task information. On the basis of information gathered from generators and users of human factors information in system development programs, it was recommended that a computerized system be capable of performing the following functions:

- Supply data, including task analysis, manning and training requirements, for any part of a system which has been duplicated in past systems or on an experimental basis--This recommendation simply means that a need

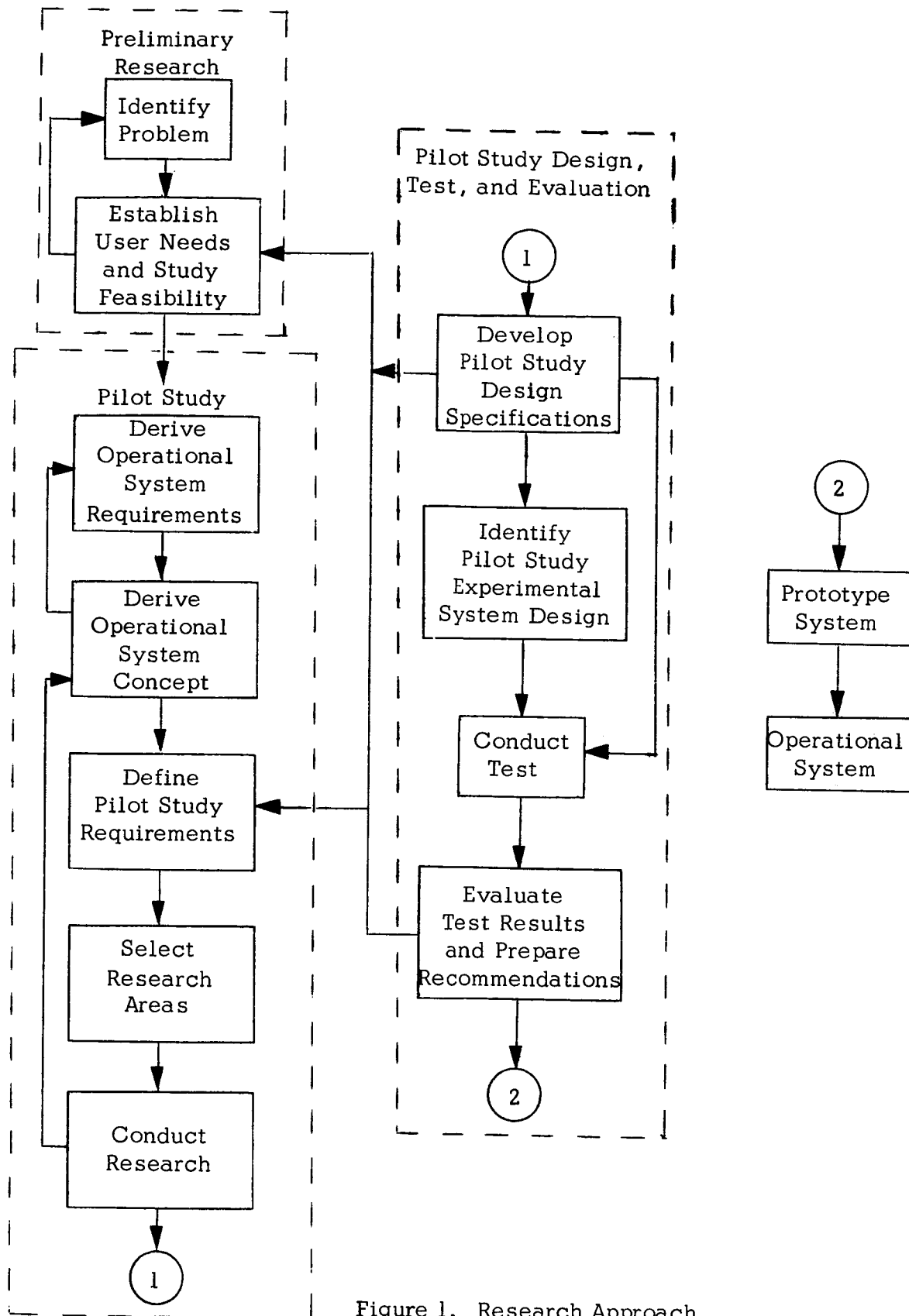


Figure 1. Research Approach

exists to provide data on past systems for design and development decisions in new systems.

- Provide rapid access to information concerning facilities, training aids, aerospace ground equipment, and trained personnel necessary to design, develop, operate, and maintain a system--This function should be available at any time during the life cycle of a system.
- Simulate any proposed system, or portion thereof, at any time during the system life cycle and at various levels of detail
- Be amenable to frequent updating in order to provide the most recent information to the user
- Generate replies to specific queries in the event that the required data are not available in handbooks or when the latest data are required--Also, the data system should provide the user with basic analytical tools to aid in the solution of system-specific problems.
- Provide data, throughout system development, in support of personnel subsystem requirements, such as QQPRI, TEPI, and TOs

The results of the preliminary research are reported in:

Hannah, L. D., Boldovici, J. A., Altman, J. W., and Manion, R. C. The Role of Human Factors Task Data in Aerospace System Design and Development, AMRL-TR-65-131.

Hannah, L. D. and Reed, L. E. Basic Human Factors Task Data Relationships in Aerospace Systems Design and Development, AMRL-TR-65-231.

Whiteman, I. R. The Role of Computers in Handling Aerospace Systems Human Factors Task Data, AMRL-TR-65-206.

Pilot Study

The objective of the pilot study was to define requirements and develop techniques for handling human factors task data. The results of the pilot study research are reported in:

Potter, K. W., Tulley, A. T., and Reed, L. E. Development and Application of Computer Software Techniques to Human Factors Task Data Handling Problems, AMRL-TR-66-200.

The functions of the pilot study were divided into five distinct groups, summarized as follows:

Derive Operational System Requirements

An extensive review of the recommendations, information, and experience resulting from the preliminary research led to the determination of fourteen requirements necessary for the development of an operational data handling system. The requirements provided the guidelines for the pilot study research. These requirements are:

- The system must be oriented to user requirements. It must satisfy the needs of aerospace scientists and engineers as well as serve management needs for data.
- The system must provide for the storage, updating, and retrieval of human factors task data. The data should be indexed to permit retrieval based on several reference points.
- The data system must be responsive to current Air Force and NASA system and data management concepts. Compatibility, rather than dependability, with the system engineering process is emphasized.
- The system must provide simplicity of use by accepting and outputting data in a form approaching user terminology. This means that all inputs and outputs must be immediately interpretable by the user. This includes all data, whether they are qualitative or quantitative in nature.
- The user of the data system must have easy access to the stored data through the use of a user-oriented query language. The terminology interpreted by the system must be compatible with the language employed by the user in his system-specific activity.
- Provisions must be made for external storage of data that cannot be coded economically for computer storage. Where applicable, cross-indexed data should be stored in the computer for referencing information filed externally to the computer, e.g., documents, pictures, and graphs.
- The data bank structure must be flexible enough to allow for future expansion and inclusion of additional data elements--categories of information. This flexibility will allow for changes in data concepts and new system requirements and avoid major changes in the structure of the data base.
- The data bank must be capable of frequent updating while retaining selected data for such uses as design trend analysis. The system must be capable of purging unwanted historical information. The updating capability of the system should allow for the storage of information generated in support of on-going phases of an aerospace system life cycle.

- The data system must be capable of retrieving similar information generated in support of different aerospace systems. This capability will allow maximum use of data in making design decisions for new systems.
- The data system must be capable of selectively retrieving data elements by qualifying them with other data elements. The user is permitted to receive only the data he needs and nothing else.
- The data system must have the capability of protecting data having security classification and/or proprietary status.
- The data system must provide the capability of processing retrieved data through the use of analytic programs and simulation models with a minimum of human intervention.
- The system must provide the user freedom in specifying the format of outputs.
- The system must provide for current awareness notifications to qualified users, in response to interest profiles. A notification is defined as a statement that data meeting the requirements of a qualified user now exist in storage.

Derive Operational System Concept

A concept of an operational system for handling and processing aerospace system human factors data in a government/industry environment was developed to lend a realistic context to the research (Reed and Wise, 1965). The concept was derived from the operational system requirements discussed above and was used to assist in the identification of problems for research. A brief description of the concept is presented below.

Figure 2 is an illustration of the conceptualized operating system in an Air Force environment. The Data Exchange Center is the heart of the system. It consists of computer hardware, computer programs, data storage, and a data control group.

For purposes of discussion, it will be assumed that a data exchange center is located at an Air Force Systems Command (AFSC) Division. Subcontractors provide prime contractors with data. The prime contractor integrates these data with his own and sends them to the data exchange center in a format ready for input to the computer. It is envisioned that new data and updates will be entered in the Data Bank on a continuous basis and not just at design milestones. The data bank contains human factors data from several aerospace systems. This makes it possible for System Program Office (SPO) or SPO designated users to compare data longitudinally within a system and across systems.

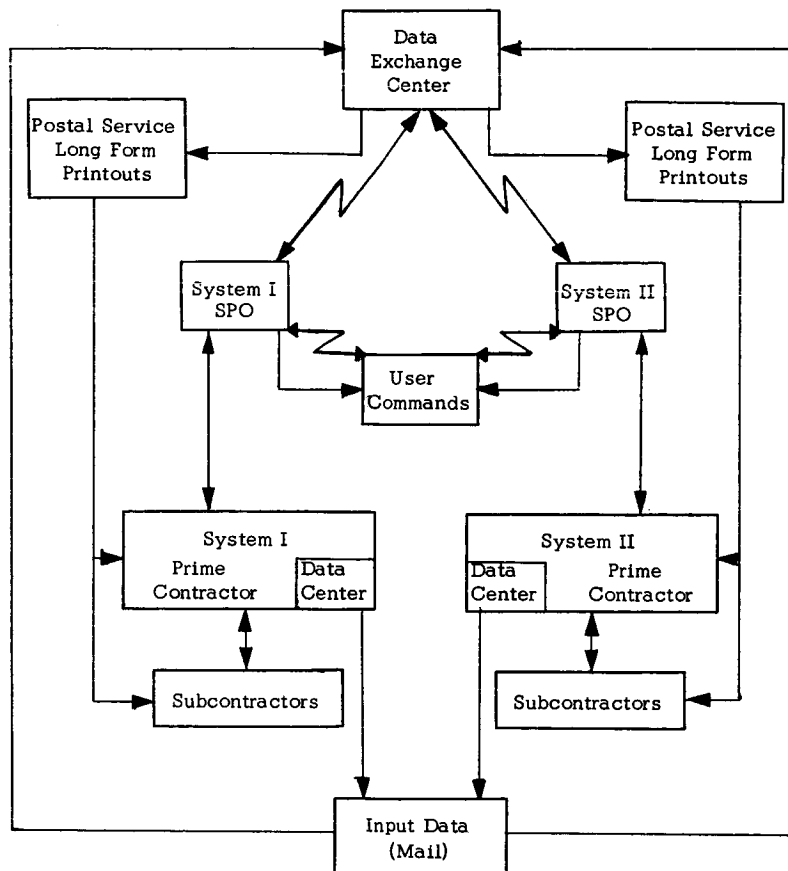


Figure 2. Conceptualized Operating System

All of the techniques and programs available at the data exchange center are available to contractors for use on their own equipment. If certain special techniques such as large-scale simulation cannot be conducted at the local center; the services of the data exchange center may be requested through the appropriate SPO.

Two modes of communication with the data exchange center are used. SPOs and user commands have direct communication with the data bank by means of teletypes on a time shared basis. Requests and brief responses to requests and current awareness notifications can be sent economically over teletype lines. Postal services are used to deliver large volume printouts. Others will use postal services or teletype to the SPO control group for all input and output communications.

The central processing unit (figure 3) must be a medium or large computer with features adaptable to time shared multiprocessing and multiprogramming operation. It must have an extremely flexible growth potential for increased processing power and storage capability.

The input/output (I/O) processor, shown separately, controls all data transfers. It allows the central processor to be free to continue basic processing while various priorities of data transfers are handled separately. These priorities include teleprocessing traffic, memory transfers, peripheral buffering, etc.

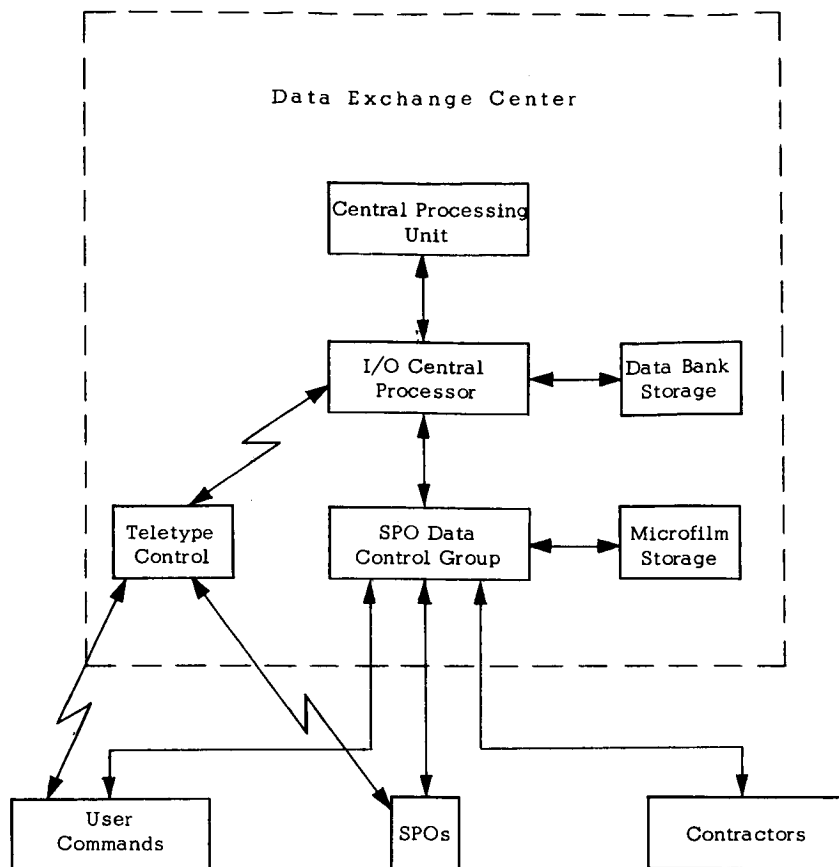


Figure 3. Operation of Data Exchange Center

The data bank storage consists of a variety of storage capabilities, ranging from direct random access to off-line cards and tapes. Because of the volume and growth rate of the data bank, additions to storage are required from time to time. Microfilm storage supplements the data bank. An index to the microfilm is maintained in the data bank. Thus, as data are retrieved from the computer storage, references are listed for supplemental retrieval from microfilm.

Teleprocessing capability is described here as a direct hook-up of remote and on-line teletype stations with the I/O central processor. These stations may be in the computer room, SPO offices in the same building, or across the country. A combination of hardware and software safeguards protect proprietary and classified information.

The SPO control group assists and monitors the data cycle within the exchange facility. This group would be an extension of each of the SPOs involved. The group would hopefully stimulate cross-talk between the SPO offices and provide an added measure of control on proprietary information. It is responsible for the collection, review, and verification of incoming data. It maintains the software; handles the reproduction of materials including data lists, charts, and reports; and monitors the releasability of all outputs.

Define Pilot Study Requirements

The pilot study requirements were developed to more clearly focus on the operational aspects of computer software functions. A concept of the computer software functions was developed to exemplify the inner working of an operational system and to narrow down the identification of research areas discussed below.

Selection of Research Areas

Five areas, considered fundamental from the standpoint of studying the problem and proposing solutions, were identified. These five areas are presented and discussed below.

- Analysis of Human Factors Task Data, Data Relationships, and Classification Schemes

One of the most important considerations in applying information system techniques to data handling problems is the determination of the characteristics of the data to be handled by the system. The factors involved in gaining a complete understanding of the data include its diversity, application, environment, content, life cycle, and significant phases in its generation and usage. The personnel involved in data handling are also important factors--their problems, response times, tools, scheduling, and products. Therefore, an analysis of the data, and of the people directly concerned with it, is needed.

- Vocabulary and Thesaurus Techniques Applied to Human Factors Task Data

A capability that increases the effectiveness of communication among man/machine/software functions is highly desirable in computer systems. Vocabularies in today's systems cannot be used as off-the-shelf components; they must be tailored to the environment of the system. In this research program, a special consideration--standardization--is apparent in studying task data terminology. Not only is the problem apparent for treating multisystem data, but also in providing for multiusers. A data vocabulary is considered essential to this information system and requires careful study in its applications.

- Computer Storing and Retrieval of Human Factors Task Data

Particular problems, identified in the first part of this research program (Hannah et al, 1965), that are of concern to users of task data are the handling of large amounts of data, dealing with scattered sources, and drawing from previous and current systems' experience. The recommendations (see page 2) call for a store of information, mutually available to those who require such information, and in a form that expedites the use of that information. These recommendations

lead to the conclusion that a factual storage and retrieval capability is needed which performs on an up-to-date basis.

- Analytical and Simulation Modeling Techniques Applied to Human Factors Task Data

Techniques employed by analysts in refining task data into useful products are, many times, the result of scientific analysis and modeling procedures. Such procedures are highly amenable to computer applications. It follows that if raw quantitative data are easily accessible in a computer store, that various processing techniques can be immediately and directly applied to refine the data into the required products.

- Current Awareness Techniques Applied to Human Factors Task Data

A special problem pointed out in the Hannah et al study (1965) involves the inability of analysts to keep up with the fast pace of data generation. This is particularly true when the data are scattered, or if channels providing awareness are inefficient. Inefficiency is almost always present when several separate organizations are involved. The requirement that the user of task data be immediately aware of its generation, when it is pertinent to his interests, can be facilitated by the functioning of a common data store. The problems of assuring awareness can be lessened by setting up a major automated control point that acts as a disseminator, providing notification of pertinent data to interested people. The operational system can act as the control point.

Conduct Research

The activity during this stage led to formulating an initial, yet detailed, design. This activity included describing particular problems and objectives for each research area, as well as planning and carrying out the objectives of the research. The results gained during this stage provided the basis for the pilot study design. The nature of the operational system concept and its man/machine functions became more clearly described as problems were explored. Modifications to the concept were made during the course of the study, as shown in the feedback loops in figure 1.

Pilot Study Design, Test, and Evaluation

The pilot study design, test, and evaluation encompasses all research areas which are carried to test and evaluation and which will ultimately result in recommendations for the development of an operational data handling system.

Develop Pilot Study Design Specifications

The pilot study design specifications are generated in response to the research results drawn from the five research areas (see Selection of Research Areas, page 9, and Conduct Research, page 10). The specifications describe and identify the requirements, techniques, functional interface, and approach necessary for the development of a data handling system. Since the objective of the research is not to develop a full operating data system, but rather to explore techniques necessary for such a system, the specifications also identify those areas which are carried into computer software and those which will be tested manually or given narrative treatment. The pilot study design specifications are presented in Section II of this report.

Figure 4 (Shaw and Gopoian, 1966) illustrates the total processes and products that are produced in development of any computer system. The pilot study design specifications must take all of these products into consideration.

Identify Pilot Study Experimental System Design

The design specifications for an experimental system satisfy those areas of the pilot study which are carried into the development of a computerized evaluative system for handling human factors task data. These specifications will result in the development of a pilot study experimental system to provide a means by which data handling techniques are tested, evaluated, and demonstrated. The pilot study experimental system design specifications are presented in Section III of this report. Section IV presents the operating environment of the pilot study experimental system.

Conduct Tests

The test program compares the total research results with the original research requirements (see Preliminary Research, page 2). The pilot study experimental system will be subjected to realistic situations, made up of real system data, as a means for running "live" queries. Human factors specialists engaged in aerospace system development programs, but who are unfamiliar with the pilot study experimental system, will be used as test subjects. Areas of the research which have not been carried into software development are to be tested manually.

Evaluation of Test Results and Preparation of Recommendations

Test results are evaluated for possible changes or recommendations for changes in the pilot study design. This approach provides an object basis for the feasibility of developing an operational system. Recommendations for the development of a prototype system or an operational data handling system will result from the evaluation of the test results, as shown in figure 1.

THE FOLLOWING PRODUCTS. . .				
Phase of Work	Relating to People	Relating to Data	Relating to Software	Relating to Hardware
System Analysis Phase	Human Action Requirements	Data Requirements	Program Requirements	Hardware Requirements
System Design Phase	Organization Specifications Job Specifications	Data Specifications • Definitions • Constraints	Program Specifications • General Description • Functions • Communications • System Test Specifications	Hardware Specifications (For Competitive Bids)
System Production Phase	Users Manuals, Operators Manuals, and Training Manuals	Data Organization • Formats • Storage and Allocation • Values • Collection and Conversion Procedures	Program Descriptions • Flow Diagrams • Set Used Listings • Program Listings • Relationships Among Programs	Equipment Manuals
System Test and Installation Phase	Trained People	Data Decks and Tapes Program Decks and Tapes	Test Results	Checked-Out Equipment
System Maintenance Phase		Operational Data Decks and Tapes Operational Program Decks and Tapes Corrections and Modifications to All Areas		

Documents

Other System Products

Figure 4. Total Processes and Products of a Computer System

Experimental Data Vehicle

The aerospace systems (object system) chosen as the data vehicle for pilot study development were the Airborne Launch Control Center (ALCC), the C-5A Heavy Transport Aircraft (C-5A), and the Saturn V Stage IC Space Booster Vehicle (SV-IC). These systems were chosen on the basis of the following criteria:

- The systems should be in their early stages of development in order that the evolutionary process of data generation usage can be given consideration.
- The systems should contain a personnel subsystem or life sciences program.
- The personnel subsystem or life sciences program requirement should be partially satisfied through a task analysis program.
- The system should be representative of those developed for the government, e.g., space systems, aircraft systems, command control systems, or missile systems.

SUMMARY

The research required to develop a computerized data system to store, retrieve, and process human factors task data will initially be implemented through a pilot study. The pilot study will demonstrate, in capsulated form, all the aspects of a total computer data system. Human factors task data were selected as the data system vehicle because task data contain a representative, cross-disciplinary body of data that can be found in most aerospace system development programs. The human factors data consist of combinations of alpha-numeric, numeric, and alphabetic information. These data are processed through a range of arithmetical or statistical procedures--simple arithmetic to complex statistical operations, mathematical modeling, and various types of simulations. These data are extremely heterogeneous, and require careful pruning and the gradual imposition of restrictive boundaries. Finally, it is hypothesized that techniques that work with human factors task data can be adapted to other groupings of aerospace system data, i.e., reliability, maintainability, safety, or engineering design data.

SECTION II

PILOT STUDY DESIGN SPECIFICATIONS

ORGANIZATION OF THE SPECIFICATIONS

Figure 5 illustrates the total pilot study design process. Beginning with the baseline established by the pilot study design specifications, three research products are developed: (1) the pilot study experimental system software specifications, (2) the nonsoftware related specifications, e.g., user's guides, and (3) the recommendations for future research. Detailed computer program descriptions are developed from the software design specifications. The pilot study test and evaluation program is a two-step process; (1) development test and evaluation that compares the results with the original design objectives (feedback to the pilot study design specifications provide a vehicle for error correction) and (2) field test and evaluation, using selected test subjects, which provides a vehicle for comparing the results with the original research objectives and for generating recommendations for future research. The final result of the pilot study design process is a set of tested and evaluated software and nonsoftware products and a set of recommendations for additional research.

The specifications establish the baseline for the design of the pilot study. Eight design areas are identified for the pilot study. They are:

- | | |
|--------------|----------------------------|
| 1. Input | 5. Analysis and Simulation |
| 2. Retrieval | 6. Thesaurus |
| 3. Output | 7. Environment |
| 4. Update | 8. User's Guides |

The specifications are organized as follow:

- A description of the design area
- Identification of the discrete parts--modules--contained in each design area
- Listing of specification requirements--these listings are divided into:
 - 1) The specifications required for the design of the pilot study
 - 2) The specifications required for the design of the pilot study experimental system

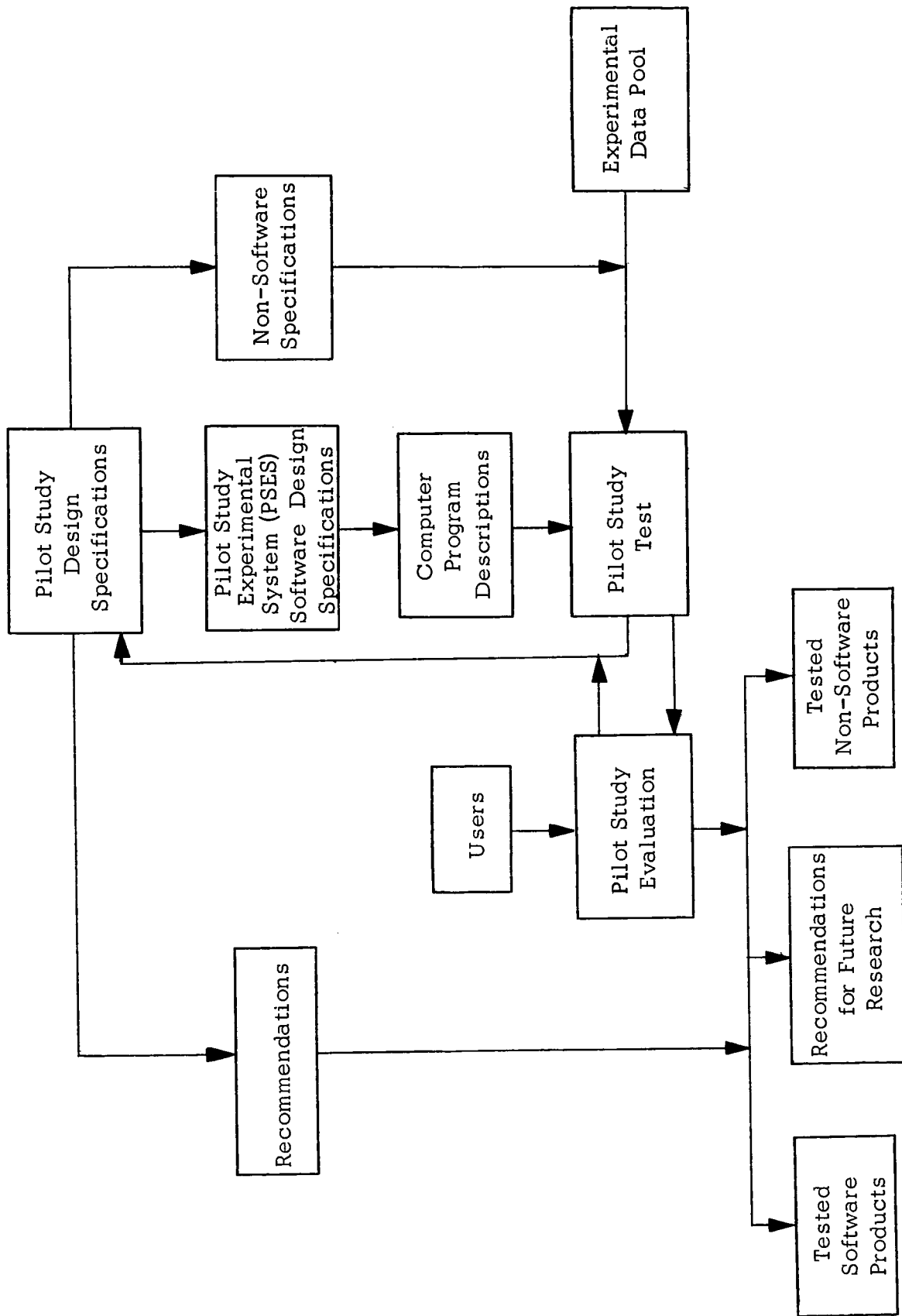


Figure 5 . Pilot Study Design Process

DESIGN SPECIFICATIONS

Input

Input is defined as those actions associated with the analysis of generated data and computer input preparation activities.

Generated Data

An analysis of the task data generated in support of three aerospace system development programs--ALCC, C5A, Saturn SV-IC--resulted in the formulation of ten data elements reported in Potter et al (1966). Appendix I contains the list and definition of these ten data elements. Preparation for computer storage was accomplished by isolating and placing all task data into their respective elements.

Pilot study specifications:

- Establish data element analysis rules and create finite definitions for the existing data elements
- Define data relationships for interdependencies that exist in generated data, coordinate these relationships with the data generators to assure that the defined relationships reflect the intent of the generators, revise relationships in accordance with the feedback received from the generators
- Establish data element rule application procedures to apply to newly isolated data elements
- During the data analysis process, assign security or proprietary data flags where indicated to maintain data integrity

Pilot study experimental system specifications:

- Identical to the pilot study specifications

Data Categorization

A categorization scheme for each data item is selected and classes of related data within each previously defined data element are isolated (see Appendix II for the list and definition of presently identified data items). The categorization scheme selected is determined by the data content of the data element. Thus, a variety of categorization schemes may be required to best organize data elements. This multiclassification structure exists because of the nature of the data. Some data are amenable to hierarchical arrangements--alphabetic or numeric--other data are best arranged by a key term index, and still other data are best grouped by an inverted subject heading scheme. Thus, it is evident that no single classification scheme is applicable to the retrieval of human

factors task data. A universal search technique, tying these categorization schemes together, must be developed as a means of integration and to provide a rational search criteria to discrete data values within data items.

Pilot study specifications:

- Assign a categorization scheme, after analysis, that best relates the data item content
- Apply data item analysis rules to the definition of data content for data item categories
- Establish procedures for creating new data item descriptions, as required
- Define data relationships for both inter- and intra-dependencies that exist in the generated data--coordinate these relationships with the generators and revise, as necessary, in accordance with the feedback received
- Define special "ripple effect" relationships that exist between data items in an inter-intra-data element environment
- Establish a rationale to permit the required cross-object system comparison of data item content

Pilot study experimental system specification:

- Identical to the pilot study specifications

Preparation of Data

Data may take a variety of forms and the methods of presentation or description vary, e.g., English or metric measurements, minutes or hundredths of hours. To provide a measure of conformity, standards must be established. The data must be converted into the established standards after receipt from data generators and before entry into the experimental data pool--data bank.

The variety of data formats and the requirement for standardization resulted in the design of an input data collection form (Potter et al, 1966, see Appendix IV). This form was designed as a means of: assisting in the construction of a data file structure, collecting--into a central location--similar data from various locations of the generator's input form, facilitating the conversion of data into standard units, and reducing the time required to convert the generated data into computer input data--keypunch.

Human factors specialists, as well as specialists in other areas, have a tendency to be nonselective in their use of words and technical terms. Attempts to standardize the use of terms have met with resistance from potential users. The problem of vocabulary standardization is well recognized among both

government and contractor personnel. Attempts to solve the problem have resulted in the publication of vocabulary lists by individual contractors and documents, such as, MIL-STD-721B, Definition of Effectiveness Terms for Reliability, Maintainability, Human Factors, and Safety or ASA Standard X3.12, American Standard Vocabulary for Information Processing. These lists or documents are intended to improve communication between the government and its contractors through the definition of commonly used words and terms. The development of standardized words and technical terms used in the system engineering process would be an overwhelming task. Even if standardization were possible, it would be difficult to expect specialists to use a dictionary containing thousands of words, terms, and specialized meanings, while conducting a system, task, reliability, maintainability, or safety analysis. How then, can a computer be used to assist in simplifying this task? Computers are, by design, idiot servants, not mechanical brains. Thus, the structure and definition of data must be highly standardized for consistent and efficient retrieval--to avoid or minimize the loss of data in retrieval through poorly defined terminology. For the pilot study, data preparation personnel will manually, by examination and substitution, perform the required vocabulary standardization before input to the computer.

Pilot study specifications:

- Prepare an analysis of the quantitative data values in the object systems and establish research pilot study standard conversion factors
- Convert generated data to computer input data in accordance with standardized criteria and the controlled vocabulary

Pilot study experimental system specifications:

- Identical to the pilot study specifications

Current Awareness

Current awareness is defined as that operational concept by which a system user will be notified that information has been received into the system. The notification process requires that the program module be developed to match input data against a series of user selected vocabulary terms--profile--that indicate, to the computer, the user's areas of interest. When a match is made between the new data and the profile, a notification of this is sent, via teletype to the user address. If the user needs immediate access to the data, he then makes a data request through the retrieval function. This limitation is imposed to insure that a user does not receive data for which he does not have immediate use. However, the data may have an indirect impact on his current problem and he can thus control the receipt of the data.

This research, for specification purposes only, has been divided into input and output requirements. During the computer input cycle, a comparison is made

between new data inputs and stored user interest profiles. Notification--output--will be sent to those users whose profiles indicate an interest in the new data.

The first consideration is one of new data quantity. A complete set of new data presents no difficulty. However, a problem exists in determining how much of an incomplete data input--data received to complete a data file in the computer where the original data input was incomplete--will constitute a requirement for notification. There is also the problem that a user may only be interested in a single data item, e.g., new AFSC designations or a new man-machine action assigned to AFSCs. It is necessary to determine a logical, justifiable cutoff point--level of data file detail--to give the most service to the greatest number of users. This problem will be resolved only after protracted contact with users.

The second consideration is the locus of the comparison between new data and user profiles. This locus may exist at two points in time: (1) before initial storage or update, and (2) after storage or update which includes those files called into memory for updating or separating in preparation for the introduction of a new data file.

Pilot study specifications:

- Provide a factual baseline for the establishment of a current awareness information cutoff level
- Design, code, and check out a computer program to perform: (1) the identification of "new" data, (2) a match between new data and computer stored user profiles, and (3) storage for output of user notifications of data availability
- Provide a guide of instructions for the initial preparation and maintenance of a list of standardized terms--vocabulary/thesaurus--that describe a user's specific interest in human factors data contained in the experimental data pool
- Design, code, and check out a computer program to establish and maintain user profiles
- Provide the user with an option of receiving data concurrent with the data notification--This option would not be the normal mode of operation but will be available to the user for a single or a limited number of data requests

Pilot study experimental system specifications:

- Identical to the pilot study specifications

Retrieval

Retrieval is defined as the computer process of obtaining specific data from the experimental data pool in response to a user query. Retrieval is of two types: (1) simple--retrieval of data or data sets from one or multiobject system(s) in immediate response to a user query, and (2) processing--retrieval of data or data sets from one or multiobject system(s) for user specified processing--mathematical/statistical or simulation manipulation. The data from a simple retrieval are retained in the original file. The data from a processing retrieval are retained in the original file, but the results obtained from the manipulation of the data are not retained in any section of the original files. The computational results may be retained, at user's option, in a unique user file that is completely separate from the data files. The user may then perform simple or processing retrievals of data from either or both the experimental data pool and the user unique computational file.

Pilot study specifications:

- Design, code, and check out computer program(s), processing modules, or both, to provide a processing retrieval capability
- Design, code, and check out computer program(s) to provide a simple retrieval capability

NOTE: If off-the-shelf generalized data retrieval software capabilities are available, their use is preferred to the development of specialized single-purpose programs or the generation of new generalized data retrieval capabilities.

Pilot study experimental system specifications:

- Identical to the pilot study specifications

Output

Output is defined as any communication, formatted or unformatted, from the computer to the user. Within this broad context, the following are included: response to user's queries, error messages, security/proprietary constraints, output mode, and current awareness considerations.

Response to User Queries

Simple or processed retrieved data will be converted into a standard response format. However, provision should be made to permit limited user options requesting that the data be displayed in a specialized format--double spaced, tabulated, etc.

Pilot study specifications:

- Provide a standard response format
- Permit user specification of the response format--This option will be exercised only at designated points within the man/computer dialog process.

Pilot study experimental system specifications:

- Identical to the pilot study specifications

Error Messages

Error messages, whether generated by a procedural or data error, should be in natural language. Symbolic or coded error messages defeat the intended purpose of the user-in-the-loop concept. The error message should be as descriptive as possible in identifying the problem. Wherever possible, the error message should provide an avenue to instructions, either computer contained or within the user guide, that will permit recovery to normal operation.

Pilot study specifications:

- Provide for error message generation in natural language only
- Error messages are to be as descriptive as possible, avoiding ambiguities
- Error messages are to be directed toward providing assistance--computer-contained or reference to user handbook--to facilitate user recovery

Pilot study experimental system specifications:

- Identical to the pilot study specifications

Security/Proprietary Constraints

Requirements identified by Potter et al (1966) established the necessity for screening output data for user access to the system. User access to the system must also be considered from the security and proprietary aspects of the data. Access to the system by users must be rigidly controlled to enforce the provisions of security regulations and to protect the contractor's proprietary data. The output computer program should:

(1) Insure that the data are not classified

- If classified, input flags set, ascertain both the user's level of clearance and need-to-know

- If classified, input flags set, insure that the data are not transmitted over unsecured circuits
- If each item of data is unclassified, insure that specific combinations do not become classified. Guidance will be obtained from each object system's security guideline.

(2) Insure that the data are not proprietary

- If proprietary, input flags set, ascertain the requestor's right to receive the data
- If proprietary and the requestor is not on the access list for data release, the SPO representative can secure from the contractor an authorization to release the proprietary data. This authorization would not imply continuing access to the data, but would be on a one-time-only basis.

Pilot study specifications:

- Provide a computer-generated security check on all data before output
- Inhibit the transmission of all data containing classified information until released by a SPO control group representative
- Provide a computer-generated proprietary check on all data before output
- Inhibit the transmission of data failing the proprietary check until released by a SPO control group representative

Pilot study experimental system specification

- Recommendation as to how these specifications could be implemented in an operational situation will be presented.

Output Mode

Determine the logical decisions required of the software design in order to display the data on one of the following equipment items:

- Teletype or dataphone circuit
- Cathode ray tube, plotter, or an optional user requested hard copy of information displayed
- An on-line or off-line printer--This capability would be for large volumes of data or data requiring security control for subsequent mailing to requester

and provide an option for making multiple copies of data output for mailing to a number of users.

Pilot study specification:

- Design output mode selector logic to include various types of output devices

Pilot study experimental system specification:

- Recommendations will be provided as to how these specifications could be implemented. The pilot study experimental system will be restricted to the use of teletype or dataphone communication links.

Current Awareness

Matches between user profiles and new input data result in a requirement to send notifications to interested users. A notification message is then composed, containing the user's identification code and the matched profile descriptor. The output software will contain the teletype addresses of the users to be notified. The notification message is then sent to the user's teletype address.

Pilot study specifications:

- Construct a standard notification format and provide a means of accumulating notification messages by user identification code
- Collect all notification messages for each user's identification code
- Match each user's identification code with his address and transmit notifications

Pilot study experimental system specifications:

- Identical to the pilot study specifications

Update

Update is defined as those pilot study procedures required to maintain the pilot study experimental system and the data. Maintenance in this context takes a diversity of meanings: to replace current data with new input data; to add totally new data to the data base; to modify a user's profile to reflect changes in his duty assignment or areas of interest; to purge the data base in order to eliminate data not used or to reassign infrequently used data from the current file to a standby file; add new data elements, data items or both for object systems in the current file, or add complete new object systems data; and add or modify pilot study or pilot study experimental system computer programs, data items, or both that are to be retained in a historical data file.

Pilot study specifications:

- Provide the computer software logic to replace computer data values in the file structures with new data values provided by data generators
- Perform an analysis of the existing data base's file structure of computer items to determine the data values to be retained in a historical file
- Provide the computer software logic to create an historical data base and add additional historical data values as these values are replaced in the current file

Pilot study experimental system specification:

- Restricted to the replacement of data values in the file structure

Add New Data to Current Data Base

During an object system's evolutionary life cycle, in addition to modifying existing data values as a result of system development, data generators create new data. The creation of the new data is often dependent upon the design of hardware or the reconfiguration of existing man-machine interactions. Therefore, provision must be developed to introduce complete functions or tasks, or both, into the existing data base.

Pilot study specification:

- Provide the computer software logic to add new functions, tasks, or both, to an object system's data base

Pilot study experimental system specification:

- Identical to the pilot study specification

Modify User Profiles

User interest in data content will vary. This variance will depend upon the status of the object system's development and the changes in the users' needs. These changes of interest will be accompanied by requests for changes in the users' profiles. Additionally, a user may discover, after familiarity with the pilot study experimental system's operations, that a different profile term or the addition of a further qualifier can result in information of more significance.

Pilot study specifications:

- Provide the computer software logic to permit a modification of users' profiles

- Provide the capability for a periodic review of each user's profile to assure that it reflects the current areas of the user's interest

Pilot study experimental system specifications:

- Identical to the pilot study specifications

Purging of Data Base

As a result of the dynamic nature of system development, information generated in response to current development activities may not be in the proper format or organization to satisfy future development activities. This situation may result in the creation of several data bases for the same object system. If the analogy were to be applied to a conceptual system, the result would be a collection of huge static data bases after the system had reached operational status. The utility of such a system would be far from optimum. To obtain optimum utility, procedures must be devised to provide substantive decision criteria for purging. For example, data generated during system development which are considered to be of little utility during the operational phase of systems or for making decisions on new system requirements, should be purged. There are two general methods of purging: (1) the complete elimination of the data to hard copy store, and (2) the removal--purging--of the data from the active file to an inactive file. A method of obtaining confirming facts about computer accession may be to have the computer retain a frequency count of each data accession.

Pilot study specifications:

- Require that a quantity count of data accessions be maintained
- Convene a purge panel, representatives of using organizations, to establish purge procedures
- Remove data to be purged from the current file; convert data to be retained into an inactive file; convert remaining data into hard copy for storage

Pilot study experimental system specifications:

- Identical to the pilot study specifications except for a manual count of data accessions during the test and evaluation phase of the pilot study design process to provide initial purge determining data

Add New Data Elements or Items to Data Base

As an object system evolves, data are created. Some of these data may require the creation of new descriptions--data elements or data items. Thus, the requirement for the new data element or item may be generated by an object system development program, by a special user request for new data or by a

reconfiguration of existing data elements or items. The requirement for adding a new object system would be generated by the SPO control group.

Pilot study specifications:

- Perform an analysis of the new data element or item to identify and name the required computer elements
- Provide the computer software logic to permit the addition of new data elements, items or both to the existing data base for the object system involved
- Perform an analysis of the new object system to identify and name the computer elements required to describe and access the object system's data elements and items
- Provide the computer software logic to add the new object system data to the computer storage

Pilot study experimental system specifications:

- Identical to the pilot study specifications

Add to or Modify Pilot Study Software

A requirement has been identified (Potter et al, 1966) to update the pilot study by adding new or modified computer programs and program modules. The researcher must remain cognizant of new developments in the state-of-the-art of programming tools and computer hardware.

Pilot study specifications:

- Identify new developments in programming tools and computer hardware applicable to the extension of the pilot study
- Prepare recommendations identifying applicable computer programs or hardware for addition to the pilot study
- Perform necessary program interface design, coding, and checkout to implement recommendations approved by the SPO control group

Pilot study experimental system specifications:

- Identical to the pilot study specifications

Analysis and Simulation Capabilities

Potter et al (1966) cited several requirements for including an analysis and simulation capability. The analysis of data for storage and retrieval is essentially a content analysis. This analysis is only the starting point of an analysis of data to determine the analytical quantities contained in the data, i.e., the scalar types and values for each data element and item. When these analytical quantities are presented on a comparison basis in the same numeric value system e.g., the metric or English systems; a time scale such as hundredths of an hour or seconds, minutes, hours; centigrade or fahrenheit temperature. The usual system analysis approach to the identification of requirements is to ask the user what he intends to do with the data. The computerized processing capabilities necessary to implement the requirements identified are then provided. However, Whiteman (1965) and Hannah et al (1965) have shown that potential users of data systems are poor judges of their own requirements and needs. Additionally, needs for data reduction vary to such an extent that they are almost impossible to predict.

An incremental process is recommended that begins with simple arithmetical operations--counting, addition, subtraction, multiplication, and division--and then progresses to more complex operations, e.g., mean, median, mode, and standard deviation. Further mathematical capabilities are identified as the user interacts with the data.

The simulation requirement will be provided in a three step process. The first step requires the design of a simple simulation capability to be included as part of the computer software. The second step permits a user to input his own simulation program, providing the simulation language is compatible. He will then be able to use either data from the experimental data pool, his own data, or some combination of data from these two sources in operating the simulation program. The third step provides a capability of including, as part of the operational system software, the large, complex, multifaceted simulation systems.

Pilot study specifications:

- Provide the computer software logic to perform arithmetic computations
- Provide the computer software logic to perform simple statistical computations
- Provide the computer software logic to perform complex statistical computations
- Provide the capability for a user to perform elementary simulations
- Provide the capability to receive, from a user, a simulation program with a compatible computer language

- Provide a means of accepting user-generated data to be used in the operation of a user's simulation program
- Provide the computer software logic to accept large, complex, simulation programs to be retained in the operational system

Pilot study experimental system specifications:

- Analyze the data to determine the applicable analysis units or analytical capabilities of the data
- Provide computer software logic to perform arithmetic computations
- Provide the capability for a user to perform an elementary simulation within the constraints of the data contained in the experimental data pool

Thesaurus

The object system data contained in the experimental data pool will be made up of a wide range of subject matter. To facilitate the transfer of information, it is necessary to implement techniques that will impose an acceptable degree of standardization on the meaning and usage of words, as well as to keep the number of terms within workable limits. The controlled vocabulary is an effective technique for managing word usage. Basically, a controlled vocabulary consists of predetermined lists or thesauri of terms to control the use of synonyms and to reduce proliferation of vocabulary. Accompanying and complementing the word lists are rules regulating usage, a glossary of acceptable terms, and a cross-reference index of acceptable terms and their more apparent synonyms. As the ability to express action is basic to describing task information, it will be necessary to develop the above mentioned regulatory mechanisms for verbs. Rules covering other parts of speech and nomenclature will also be developed as required. Standardization of terms will minimize grammatical, syntactical, and semantic problems; increase system efficiency and avoid uncontrolled vocabulary; and facilitate the retrieval capability by having a concept represented by one acceptable term, rather than by numerous terms.

Analysts have an unlimited choice of words and terms even to the use of improper or divergently defined words for emphasis. The use of a thesaurus is equally applicable to technical analysis writing and document or data systems. One purpose for developing a thesaurus during the pilot study is to test its applicability in an operational context. Although the thesaurus will not be automated for the pilot study experimental system, manual operations--examination of a procedural step, a man-machine action, and the substitution of the standardized vocabulary terms--will provide a means of determining the use and value of a standardized vocabulary. To demonstrate the automated operation, a small subset of the thesaurus vocabulary verbs will be automated to test the concept, establish the operating conditions and limitations, and demonstrate the application to human

factors data as well as to other types of system engineering data, e.g., reliability or maintainability data, equipment nomenclature, or engineering design data.

In addition, the controlled vocabulary will supply a standard series of terms that will be used as inputs to the development of a faceted classification system. The purpose of the faceted classification scheme will be to provide a method of retrieving data from any or all of the systems in the data base. A faceted classification system is basically one in which terms are grouped by conceptual categories and ordered to display their generic relationships. There are two general types of conceptual categories. The fixed categories consist of such things as processes, phenomena, materials, equipment, living organisms, economic factors, and places. The general categories may consist of properties, operating conditions, and common attributes. The general categories act as modifiers to the fixed categories, e.g., equipment-aircraft radio (fixed), operating condition-altitude above 50,000 feet (general). The facets become a means of linking conceptual categories. Essentially, the type of faceted classification proposed for the experimental data pool is one with conceptual categories replacing traditional main classes; insists on the homogeneity of concepts within the same category, facet or subdivision; emphasizes the intent to which terms are free to combine with another to express more complex terms and phrases; and maintains a unique place for each term. A faceted system that makes use of a category framework is able to uniquely place terms within schedules. That is, each term in the framework is unique to the concept being expressed. The same term may be used within another classification framework for expressing an entirely different concept. Therefore, concepts can be classified according to their basic characteristics. Also, the unique placement of terms means that the classification scheme can be simpler and briefer than hierarchical classification schemes.

Pilot study specifications:

- Develop a preliminary thesaurus of human factors terminology
- Test the feasibility of automating a human factors data thesaurus
- Establish rules for vocabulary standardization techniques and procedures
- Develop lists of precisely defined verbs, objects, and modifiers; coordinate these lists with data generators, potential system users, human factors personnel, and lexicographers; add approved terms to the preliminary thesaurus
- Research the applicability of a faceted vocabulary technique for the retrieval of human factors task data across systems

Pilot study experimental system specifications:

- Identical to the pilot study specifications

Environment

Environment is defined as the operational conditions and constraints within which the pilot study experimental system, prototype, and operating systems will function. These conditions and constraints include: hardware, computer software, other software systems, interface with other Air Force and NASA systems, interface with Air Force and NASA data management regulations and programs, and provision for the system test and evaluation.

Computer Hardware

Due to the size of the data base, a relatively large computer complex is required. Processing speed is not a necessity for system operation. However, to achieve optimum use in providing user services, a time-sharing environment is a necessity (Potter et al, 1966). Since time-sharing operation is a previously unexplored capability in this kind of application, the necessity of demonstrating operability in the research is a specific requirement. This will require either a special purpose or large, third generation computer such as the IBM 360/50, RCA Spectra 70/46, GE 645, or other large computer systems. The pilot study will use a minimum of input/output (I/O) devices. These I/O devices will include random access storage devices, an off-line printer, and teletype equipment used as remote consoles.

Pilot study specification:

- Provide a computer complex with the following characteristics: a large random access storage capability, capability for time-sharing, and adequate flexibility to permit addition of a variety of experimental I/O devices

NOTE: For the pilot study experimental system, the IBM, AN/FSQ-32, ARPA funded, time-shared research computer complex at the System Development Corporation, Santa Monica, California, will be used to provide research and development capabilities. Figure 6 presents an overall view of the equipment configuration of the AN/FSQ-32, indicating its on-line peripheral equipment and its relationship to the Digital Equipment Company, PDP-1 Computer, which handles all interactive input/output requirements between users and the time sharing system, and the IBM 1401 computer, which handles off-line requirements. Table 1 provides the general characteristics of the equipment.

Computer Software

Computer software is defined as including both the program or programming logic and the operating language. The programs to be developed or used off-the-shelf shall include:

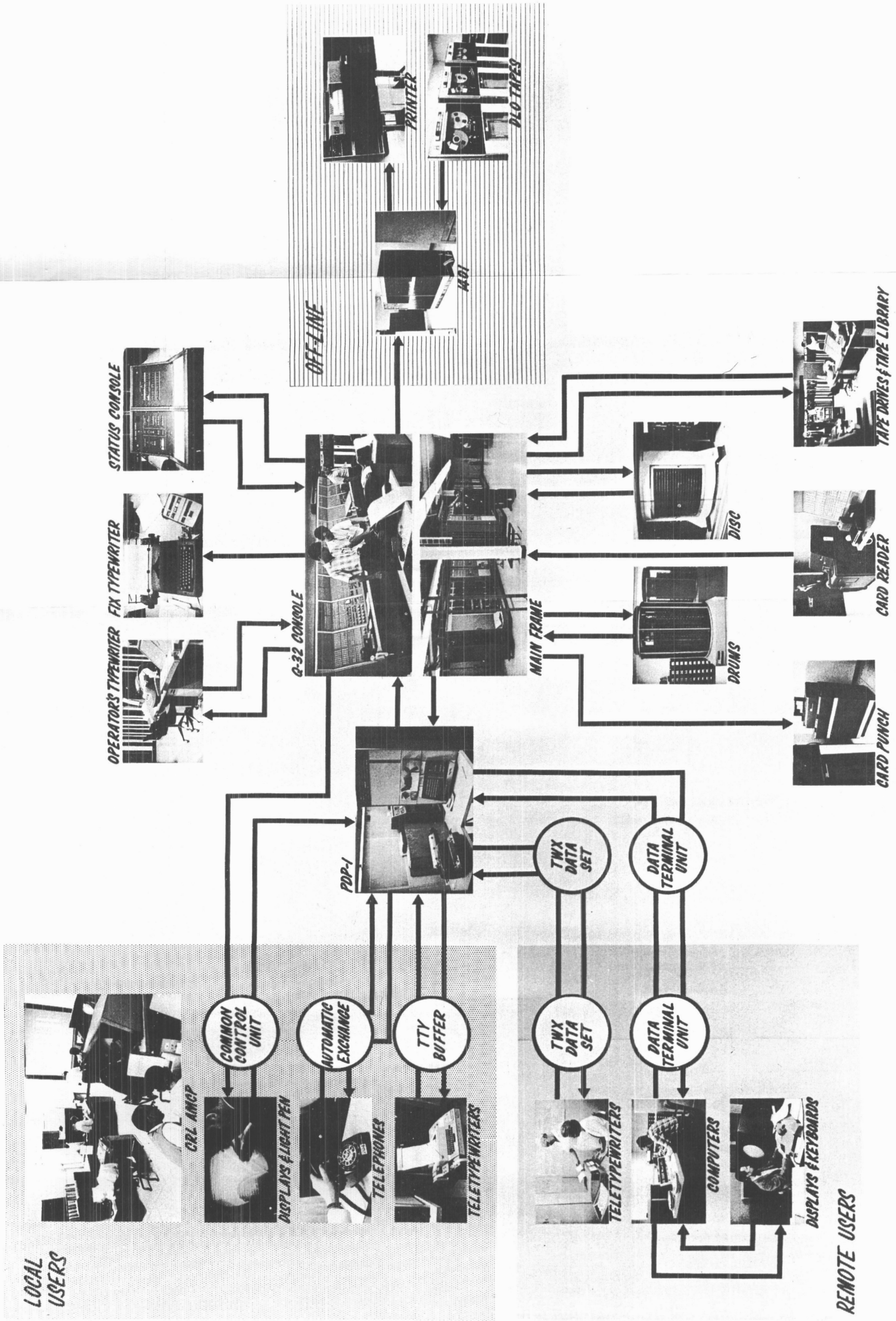


Figure 6. AN/FSQ-32 Equipment Configuration

COMPONENTS	NUMBER	CAPACITY/SPEED	TOTAL
<u>AN/FSQ-32 COMPUTER</u>			
<u>Main Core Memory</u>	4	16,384 words	65,536 words
• Cycle time 2.5 μ sec.			
• 48-bit word			
<u>Input Core Memory (Buffer)</u>	1	16,384 words	16,384 words
• Cycle time 2.5 μ sec.			
<u>Drum</u>	5	139,264 words	417,792 words
• Access time 11.5 ms.			
• Word transfer rate 2.75 μ sec.			
<u>Disc File</u>	16 discs	262,144 words	4,194,304 words
• Access time 225 ms.			
• Word transfer rate 12 μ sec.			
<u>Tape Drives (729-IV)</u>	12	112 1/2 ips	
<u>Card Reader (714)</u>	1	250 cpm	
<u>Card Punch</u>	1	100 cpm	
<u>Typewriter</u>	2	100 wpm	

ASSOCIATED COMPUTERS (ON/OFF-LINE)

<u>PDP-1</u>			32K words
• Shares input core memory of Q-32			32K words
• Cycle time 5 μ sec.			
• 18-bit word main core memory	1	4K words	4K words
<u>1401-D</u>			
• Core memory	1	4K words	4K words
• Printer	1	600 lpm	
• Tape drives (729-IV)	1	112 1/2 ips	
• Card reader (Uptime)	1	850 cpm	

I/O DEVICES

<u>Teletypes and Typewriters</u>			
• Model 33 Teletypes	34	100 wpm	
• TWX data sets (remote users)	6	100 wpm	
• Soroban typewriters	3	90 wpm	
• Telex data sets	1	100 wpm	
• Data-Phone sets	6	100 wpm	
• IBM 1051-1052	1	150 wpm	
• Communications testboard	1		
<u>Display Consoles</u>	6	2K char. max. (per console)	
• Light pens			
• Vector-generator capability			
• Graphic tablet	1	5K points/sec.	
• Keyboard	2	100 wpm	
<u>Telephones</u>			
• Links for simultaneous conversions	12	100 wpm	
• Phones	35		
• Recording by PDP-1 (who called, when, how long)			

- 32-1 - Table I. Characteristics of AN/FSQ-32 and Related Equipment

32-1

32-2

- Executive or system monitor and control
- Input processor
- Retrieval capability
- Statistical processing capability
- Simulation capability
- Output processor
- Current awareness function
- Thesaurus capability function
- System fault diagnostic capability

A higher-order, problem-oriented language was identified by Potter et al (1966) as a system requirement. Additionally, the following languages were specified as probable system candidates: FORTRAN, COBOL, and JOVIAL. The art of compiler development has resulted in several compatible computer hardware configurations being supplied with FORTRAN, or COBOL, or both compilers. JOVIAL is a more flexible language combining the better capabilities of FORTRAN, COBOL, and ALGOL, yet retaining only a minimum of their limitations.

LUCID (Language Used to Communicate Information System Design) and TINT (Teletype JOVIAL Interpreter) are presently operating time shared, on-line computer program systems, written in JOVIAL and available on the AN/FSQ-32. Thus, to facilitate the development of data and vocabulary analysis techniques, the initial research will be organized in a LUCID format.

Other Software

The full requirements for an operational system's software are: users' and controllers' operations and maintenance manuals, a training course, awareness or cognizance of Project LEX--a Department of Defense directed effort to prepare a DoD-Wide Technical Thesaurus--requirements, and other documentation. These software requirements will initially be met through a user's and a controller's guide.

Interface - Other Systems

Long before the advent of the AFSCM/AFLCM 310-1 data requirements, contractors, military customers, and other government agencies were becoming concerned with data generation and data management. The problem had been proliferated into a seemingly endless array of manual or automated data generation systems. The introduction of the 310-1 series reflects the Air Force's concern

over the data problem. Some of these data systems have a direct impact on the research by providing input data; other systems may be affected through the use of pilot study generated data. Therefore, consideration must be given to the data generation and data requirement aspects of other data systems.

Pilot study specification:

- Maintain cognizance of the data requirements and data output to determine impact on, or impact from, the research. Concrete research results may not be possible, but specific instances of interface recommendations will be provided.

Pilot study experimental system specification:

- Identical to the pilot study specification

Interface - Data Management

Included in all technical data systems is a requirement, sometimes unspecified, to comply with a data management system. Technical data systems are intended to manipulate individual items of data--documentary or single fact information--and to present the results in formats specified by the users. Data management systems are intended to determine the most advantageous way that the reports from a technical data system can be used, as decision aids, by various levels of management.

Data + Analysis = Information--Technical Data

Information + Judgment = Decision--Data Management

The pilot study and any iterations that may follow, prototype or operational system, is directed to technical data generated in support of Air Force and NASA aerospace system development programs. Therefore, the pilot study must be responsive to and compatible with Air Force and NASA data management system requirements. The design of the pilot study is not restricted to current data management system requirements. A technical data system developed to respond only to current data management requirements will not be able to adjust to changes in data management procedures except with considerable costly reprogramming. For example, a number of excellent data analysis and simulation programs have been written in FORTRAN and SIMSCRIPT; IBM is perfecting PL/I (Programming Language I) and GPSS (General Purpose Simulation System) as more powerful replacements for FORTRAN and SIMSCRIPT; the users of the current programs will be faced with an extensive and expensive reprogramming effort if they wish to take advantage of these improvements. The same analogy can be applied to the current Air Force and NASA data management data systems.

Pilot study specifications:

- Determine data requirements necessary for compatibility with Air Force and NASA data management programs
- Comply with these requirements through narrative descriptions of intent or by actual imbedding in the pilot study computer software

Pilot study experimental system specifications:

- Identical to the pilot study specifications

Research Testing and Evaluation

The purpose of a research testing and evaluation program is to provide measures of compliance with the original development requirements (see page 2) and to receive judgments concerning pilot study capabilities, limitations, and concept implementation.

Research testing and evaluation is an activity that compares research results with the original conceptual requirements. The conceptual requirements were generated as a result of a series of interviews, personal contacts, and questionnaire responses (Hannah et al, 1965; Whiteman, 1965; and Hannah and Reed, 1965). The concept and the research, at that time, were sensitive to the potential user's requirements and needs.

Development testing and evaluation is the process of assessing the research results against the original user needs expressed as conceptual requirements. Deviations from the requirements must be adequately substantiated and justified.

Pilot study specifications:

- Conduct a research test comparison between current research results, research directions, and the conceptual requirements
- Identify new research requirements as a result of the research test

Pilot study experimental system specifications:

- Identical to the pilot study specifications

Field testing and evaluation is a two step process designed to insure that potential user's requirements and needs can be met with the research products and that the potential user can operate the products with the tools provided. The evaluation also provides the user with a medium designed to acquaint the researcher with new requirements or needs. This process is to be conducted with potential users--in the field--or at the pilot study development facility. The first part of this process is as follows:

- (1) Design a questionnaire or interview procedure that begins by describing the data included in the experimental data pool; describes the logical operations that are possible on these data; and requests that the recipient respond with a number of typical questions that could be asked, using the logical operators, and the data contained in the experimental data pool
- (2) Mail questionnaires to a sample of potential users of a data system
- (3) Tabulate and analyze the questionnaire responses to sort similar types of queries
- (4) Phrase user queries, on-line, through pilot study experimental system and compare the results obtained from the computer with the anticipated results

The second part of this process is to be accomplished as follows:

- (1) Select a sample of test subjects from a population of Air Force, NASA, and industry personnel. The ideal test subject is one familiar with the scope and use of human factors task data, and unfamiliar with the pilot study experimental system.
- (2) Provide a complete user's guide and a brief orientation to the operation of the pilot study experimental system
- (3) Encourage the test subject to use the pilot study experimental system in solving typical human factors task data problems
- (4) Request the subject to comment--ease of handling apparent difficulties, recommendations for improvement--on a tape recorder or in writing as the pilot study experimental system is exercised
- (5) As a result of the combined experiences and test subject comments, recommendations for the development of future system iterations will be based on factual rather than conjectural evidence.

Pilot study specifications:

- Prepare a detailed test and evaluation plan
- Select a sample of test subjects from human factors and SPO personnel for the two test phases
- Prepare a detailed questionnaire to provide data for evaluation testing
- Prepare a test plan (scenario) for the field evaluation testing
- Select methods for analyzing the task data

Pilot study experimental system specifications:

- Identical to the pilot study specifications

Guides

Guides will be provided to instruct and assist the controller--SPO control group computer operator--and the pilot study experimental system user in their operation of the pilot study experimental system. These guides must contain instructions necessary for the operation of the remote consoles and of all computer programs--whether off-the-shelf or specially developed--that comprise the pilot study experimental system software package.

It will be necessary to provide separate guides for the controller and for the user of the system. A control must be exercised over the loading and updating of current files to safeguard the integrity of the data. Access to these programs, then, should be provided only to the controller, not the users, of the system. Instructions for both guides will encompass the following areas:

- Establishment of communication
- Formulation of input
- Interpretation of output
- Recovery procedures

User's Guide

The design requirements for a user's handbook will be to provide instruction and logical examples for the operation of the following software components:

- Querying the data
- Loading and updating user profiles
- Current awareness capabilities
- Analytical techniques

Pilot study specification:

- Use pilot study experimental system user's guide to determine pilot study specifications

Pilot study experimental system specifications:

- Prepare pilot study experimental system user's guide

Controller's Guide

The controllers will receive instructions, also, in communication with the time sharing system itself, loading new data, and updating current data. The design requirements for a controller's guide are to provide instructions for the operation of the following software components:

- Time sharing system
- Loading new data and updating current data
- Querying the data
- Loading and updating user profiles
- Current awareness capabilities
- Analytical techniques

Pilot study specification:

- Use pilot study experimental system controller's guide to determine pilot study specifications

Pilot study experimental system specification:

- Prepare pilot study experimental system controller's guide

SECTION III

PILOT STUDY EXPERIMENTAL SYSTEM

INTRODUCTION

This section presents the design specifications for an experimental system that will satisfy those areas of the overall pilot study design requirements described in Section II that are to be carried into the development of a computerized evaluative system. The experimental system, herein referred to as the pilot study experimental system, provides the primary means of demonstrating and evaluating the concepts of the operational system discussed earlier in this report.

The primary activities of the experimental system include:

- Manual data preparation of task information for entry into computer storage
- Computerized file establishment that stores the prepared data into data base files
- Computerized file maintenance that provides the means to update the data base files in order to assure that their information content is kept current and correct
- Computerized data retrieval that retrieves and presents information from the data base files
- Computerized data notification that provides users of the experimental system with notifications whenever data in their particular areas of interest are entered into the data base files
- Manual profile preparation that provides the means of translating user-defined areas of interest for data notification into vocabulary controlled value lists (profiles) for entry into computer storage
- Computerized profile maintenance that establishes, maintains, and presents profile information
- Computerized analysis that performs standard and user-defined analytical processing of information contained in the data base files

Figure 7 illustrates the manner in which these individual activities are related to form the pilot study experimental system. The figure also identifies the major input and output of the system.

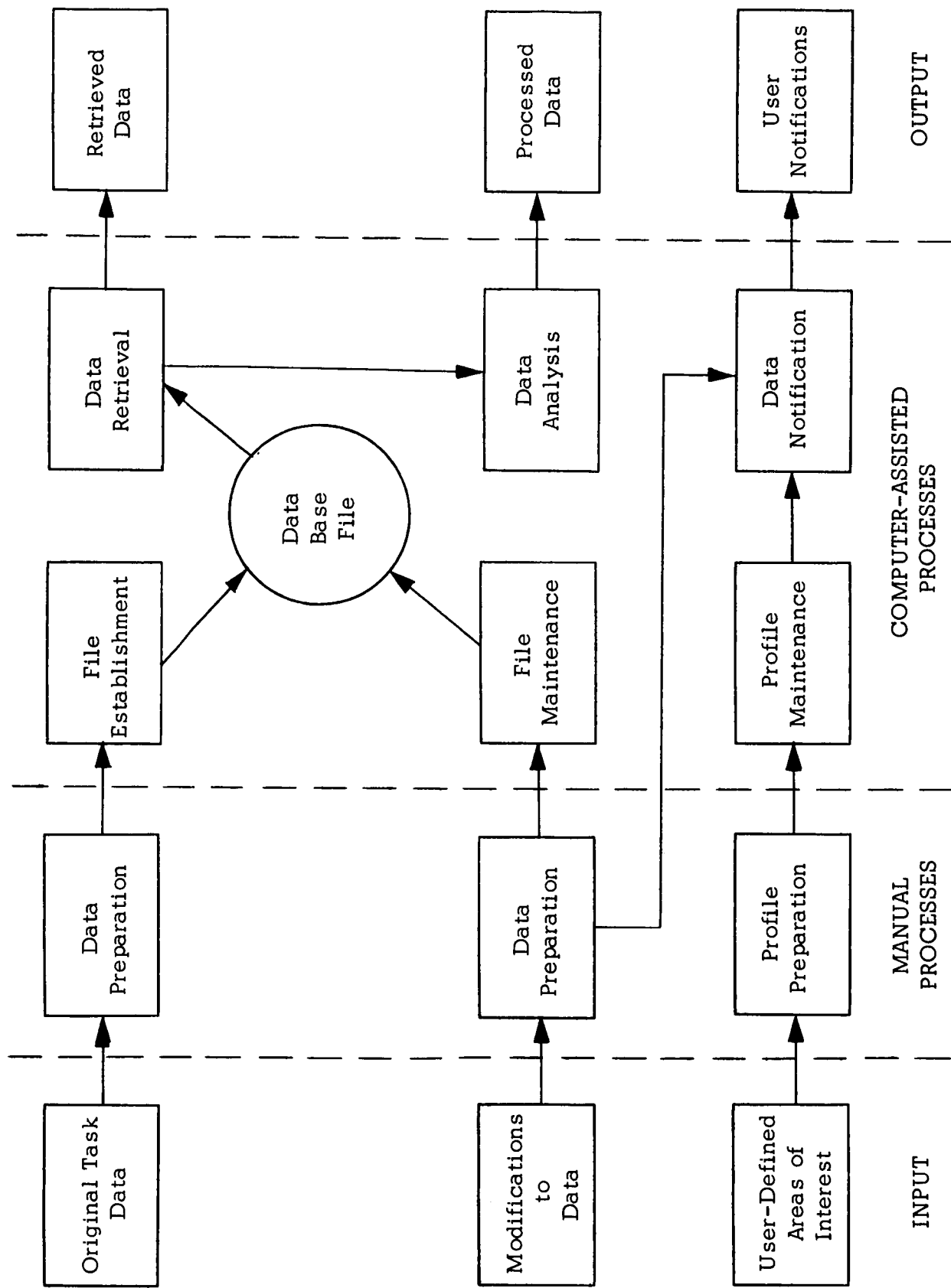


Figure 7. Pilot Study Experimental System

PILOT STUDY EXPERIMENTAL SYSTEM DESIGN SPECIFICATIONS

The design of the pilot study experimental system is affected by three major considerations:

- The capabilities attainable through the use of existing software that is readily available to the research effort
- The capabilities attainable through the use of software that is feasible to develop as part of the research
- The status of the concurrent pilot study research areas upon which the experimental system depends and with which it interacts

The degree of application of existing software to development of the experimental system depends upon the software capabilities available at the computer facility where the system will be developed and implemented. As indicated in Section II, the computer facility to be used is located in the research laboratory of System Development Corporation in Santa Monica, California. The facility is centered about a large-scale digital computer, the IBM AN/-FSQ-32 (figure 6, page 31). The computer operates in a time-shared mode and is accessed via on-line teletype console.

The time-sharing system in operation on the computer offers a number of software capabilities of direct value to the research. These include:

- A program development capability, by which programs can be developed, checked out and executed in a time-shared environment. This capability provides the use of a high-order programming language (JOVIAL), a complete debug package and standard editing routines. Development of small programs, such as statistical routines can be accomplished entirely on-line in an interactive mode through the use of an interpretive JOVIAL compiler, TINT.
- A general-purpose data management system--LUCID--that satisfies the primary information storage and retrieval requirements of the experimental system

Figure 6 and Table 1 (see Section II, pages 31 and 32) present the equipment configuration and general characteristics of the AN/FSQ-32 computer complex.

Design of the experimental system is considered from three broad design areas. These areas are: (1) information storage and retrieval, (2) current awareness, and (3) analysis. Figure 8 illustrates the grouping of the previously defined activities of the experimental system into the three design areas. The following discussions examine each of the areas in turn.

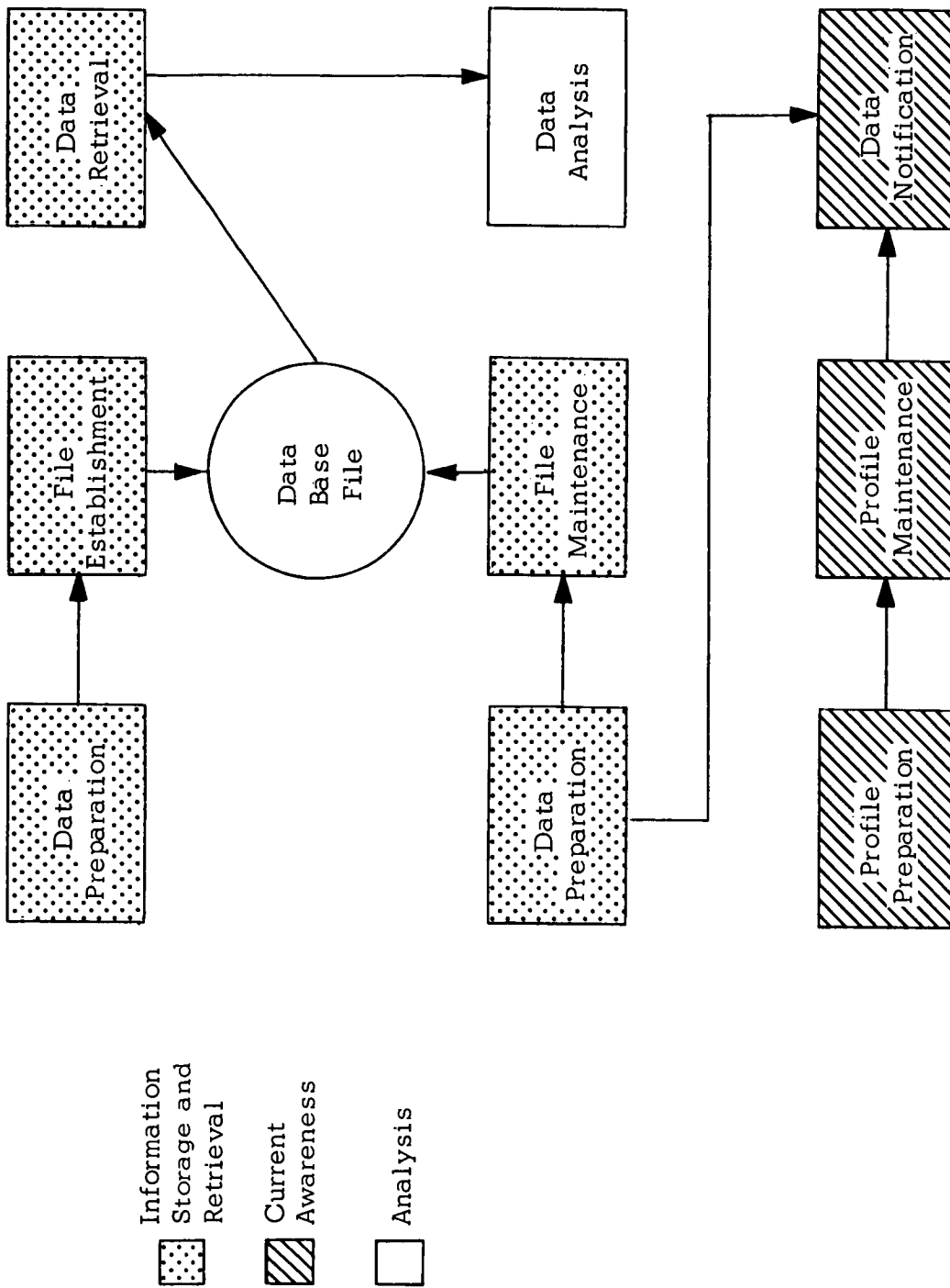


Figure 8. Pilot Study Experimental System Design Areas

Information Storage and Retrieval

The basic requirements for computerized information storage and retrieval of human factors task data are common to requirements for handling other types of similar information. These requirements center about the need to establish, maintain, retrieve, and present large volumes of data. An existing generalized information storage and retrieval system with proven large volume data handling ability--LUCID--will be used to satisfy the processing requirements of the experimental system in this area. A functional description of the LUCID system is presented in Section IV.

Data Preparation

Data preparation consists of the procedures necessary to convert raw task data to a format acceptable for computer processing. The data that will be processed by the experimental system consist of task information associated with the development of three separate aerospace systems. These systems, herein referred to as object systems, include the Airborne Launch Control Center (ALCC), the Heavy Transport Aircraft (C-5A), and the Saturn V Stage IC Space Booster Vehicle (SV-IC).

The task information for these systems exists in a variety of physical formats. These formats range from informal handwritten analysis worksheets to highly-formatted, typewritten forms. The first step in preparing the information for entry into computer storage is to arrange the information in a standard format for all object systems. This standardization compacts the information and facilitates its further preparation. Standardization involves extracting key information from the various input formats and arranging it on common collection forms. Specific guidelines for extraction depend upon the nature and format of the information as received, and are similar in level of detail to the guidelines established for the extraction of ALCC data (Potter et al, 1966). Example guidelines are presented in Appendix III; a completed collection form is presented in Appendix IV. The result of the extraction process is a completed collection form for each task analyzed. The information on each form is presented in the broad categories of information defined in Appendix I, i.e., mission information, equipment information, etc.

After the information is extracted, the collection forms are reviewed and edited in order to standardize, where possible, the representation of the actual data within each of the categories of information. For the experimental system, this process consists of applying the preliminary rules for verb usage defined by the concurrent pilot study vocabulary research (see Thesaurus, Section II, page 28), to the expression of human actions. For example, push-buttons are always depressed--never pressed, pushed, or hit. Subjective standardization is also applied to phrases that have the same meaning, but are presented differently in various instances. "Airborne operations," for example, is used rather than "in-flight operations."

Although the overall categories of information used to express task data for each of the object systems are fairly standard (see Appendix I), they are too broad to permit satisfactory computerized manipulation of the data by users of the experimental system. These categories must be subdivided into more definitive subcategories which represent the uniquely identifiable pieces of data that comprise the task data. The subcategories are not absolutely standard across all of the object systems, but vary in accordance with the specific information content of each of the systems and the level of task analysis performed. The following list represents those subcategories of information, hereafter simply referred to as categories, that have been identified as being representative of the data in each of the object systems being used. Categories identified for individual systems are variations of the list.

Source Identification
Revision
Revision Date
Remarks
Mission Phase
Mission Segment
Task
Performance Descriptors
Location
Hardware Information
Special Tools/Equipment
Task Frequency
Hazards
Safety Precautions
Performance Type
Total Task Time
Personnel Type
Personnel Number
Personnel Time
Difficulty
Criticality
Training Requirements

The following discussion presents the information content and value characteristics associated with each of the categories. Just as the categories are representative of the information contained in the three object systems, the value characteristics are also representative and do not necessarily reflect the characteristics of any particular object system.

- Source Identification contains the author, document, references, classification, and date associated with the analysis of a particular task. The information is alphanumeric in nature.

- Revision is the revision code associated with the task data as they are modified. It is usually a letter.
- Revision Date is the date associated with the latest revision of a task. If no revision has occurred, this category contains the date of original data. The data are considered to be numeric information.
- Remarks contains any remarks associated with the task data. The information is alphanumeric, usually one or more sentences in length.
- Mission Phase contains the broadest level of mission profile information. The information is alphanumeric and contains values such as "preflight operations," "airborne operations," and "postflight operations." Its purpose is to define the phase of a mission in which a task occurs.
- Mission Segment is a finer breakdown of mission phase. The information is also alphanumeric, and contains one or more values for each mission phase. The phase "airborne operations," for example, might contain segments that include "preattack operations," "attack operations," and "post-attack operations." Its purpose is to define the segment of a particular mission phase in which a task occurs.
- Task is the name of the task associated with the analysis. It is alphanumeric in nature.
- Performance Descriptors are the individual procedural steps required by individuals in order to perform a specific task. The information is alphanumeric and is generally quite lengthy because it is represented at a very fine level of detail.
- Location information is alphanumeric and describes the physical location associated with performing a specific task, relative to the system being analyzed. Values range from specific crew work stations for operational tasks to field maintenance facilities for maintenance tasks.
- Hardware Information is alphanumeric and contains the names or designations of the hardware associated with a specific task.
- Special Tools/Equipment is alphanumeric and contains those equipment items identified as being required to perform a specific task.
- Task Frequency describes the frequency with which a specific task is performed. Although essentially numeric, the information is represented as alphanumeric, and contains values such as "once per flight," "as required," and "twice per mission."

- Hazards contains alphanumeric information regarding hazards associated with the performance of a task.
- Safety Precautions are alphanumeric and contain statements regarding precautions to be taken during performance of a task.
- Performance Type contains alphanumeric words or phrases that describe the nature of a task. Typical values are "operational," "routine," and "maintenance."
- Total Task Time contains the total time required to perform a task, and is expressed in minutes.
- Personnel Type identifies the types of personnel required to perform a task either by position title (e.g., pilot, copilot, flight engineer) or by code (e.g., AFSC). The information is alphanumeric. More than one type of personnel may be identified for a task.
- Personnel Number contains one or more numeric values representing the number of each type of personnel required to perform a task.
- Personnel Time contains one or more times, each associated with a particular personnel type, and identifies the amount of time spent in performance of a task by each of the personnel types involved in the task.
- Difficulty contains a coded value that represents the degree of difficulty associated with the performance of a task.
- Criticality contains a coded value that represents the degree of criticality associated with the performance of a task.
- Training Requirements are those specific training requirements identified as being necessary for the personnel involved in performing a task. The information is alphanumeric.

After extracting, editing, and organizing the data; they are transcribed onto punched cards in the format required by the LUCID program responsible for establishing the initial data base files. This format and the program are described in Section IV. Because of the relative compactness and clarity of the information on the collection forms, no intermediate manuscripting of the information onto keypunch work sheets is required. Key punching is performed directly from the collection forms.

The collection forms and the keypunched cards are maintained as permanent physical files. The cards are also prestored on magnetic tape and sent to the AN/FSQ-32 computer facility for processing by the LUCID system.

File Establishment

File establishment consists of the procedures necessary to store the prepared data into computer files, referred to as a data base. Before the actual storing can be performed, it is necessary to indicate to the LUCID system, the organization and nature of the data for which files are being established. This permits LUCID to develop an efficient structure for each of the data bases and to recognize and interpret the data values that are to be stored in them. This preliminary processing is referred to as data definition.

Because information, as received for input to the experimental system, is task oriented, the organization of the data for computer storage will be centered about the task. All information associated with each task is logically grouped together in the data base files.

Although the basic organization of information is standard for all object systems, the categories of information necessary to describe the data are not. This requires that a separate data base be defined and developed for each object system. Each definition establishes the data categories, their relationships, and the characteristics of the data that exist in each category for a particular object system. Once the definitions are established, they are entered on-line, via teletype console to the LUCID program responsible for data definition processing. Formats for entering the definitions are discussed in Section IV. The result of the data definition process is a definition tape for each object system. Each tape contains definition tables that describe the characteristics of the data base being constructed for a particular system. These tapes are used by other programs of the LUCID system when establishing, maintaining, and retrieving data from the individual data base files.

After data definition is complete, the prepared task data are ready to be stored in data base files. The data for each object system and the related data definition tape are input to the LUCID system load program. The program stores the data into data base files according to the definitions. The result is a separate data base for each object system. At this stage, the term data base must be expanded in definition to include not only the actual data values stored, but also the data base indices that are generated by the storage program. These indices permit efficient access to the stored data for retrieval and maintenance functions. Figure 9 presents an illustration of the file establishment process.

File Maintenance

File maintenance consists of the procedures necessary to update the data base files once they have been established. There are two types of update information input to the experimental system. The first type consists of relatively large volumes of complete task information that are added to the data base. The second consists of relatively minor changes, primarily error corrections, that are made to data already in the data bases.

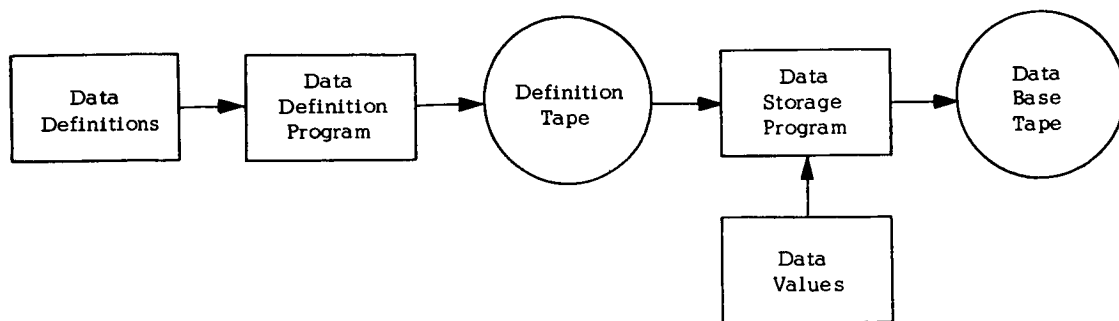


Figure 9. File Establishment Process

Batch Updating

The addition of large volumes of new data to the data base files in the form of complete task information is accomplished in the experimental system by the batch updating capability of the LUCID system. This requires constructing, for each object system being updated, a separate data base containing only the new information to be added to the files. The resultant data base is then merged with the related original data base. Merging is accomplished by another program component of LUCID. The merging process accepts the original and update data bases and constructs a new data base containing all of the information contained in the two input data bases. Figure 10 presents an illustration of the batch update process.

The process of preparing new data for entry into the experimental system and their subsequent entering into computer storage as a separate but related data base is similar to the processing and entering of the original data. The primary difference is that computerized data definition need not be repeated. The establishment of a data base containing only new data to be added to the existing data base of any particular object system can be accomplished by utilizing that system's original data definition tape. If it becomes necessary to completely redefine the categories and structure of a particular data base to satisfactorily accommodate

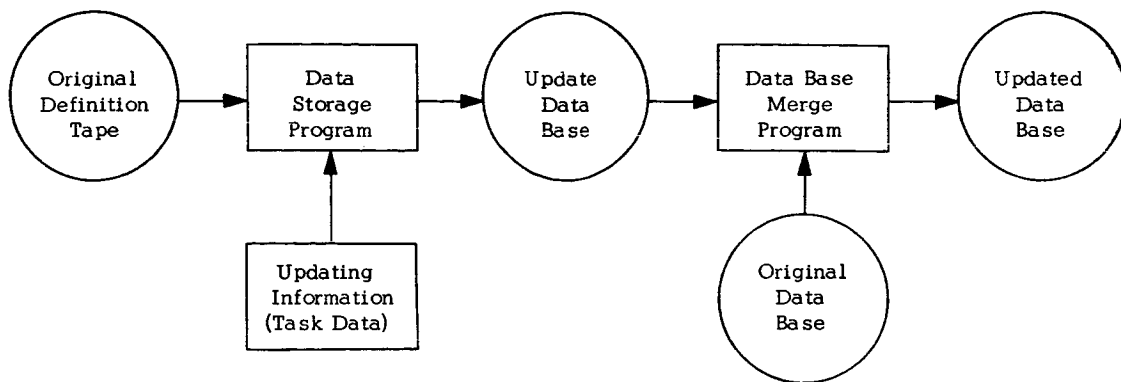


Figure 10. Batch Update Process

new data, the entire data base, including the updating information, must be reloaded in accordance with the new definition.

On-Line Updating

The modification of information once it has been stored in the data base, as opposed to the process of adding complete new tasks and accompanying data, is accomplished by the on-line update capability of the LUCID system. On-line updating permits updating of data bases to be performed directly on-line through the use of a teletype console. Types of updating available to the experimental system include:

- Addition of new task data to the data bases
- Addition of select data to task data already in the data bases
- Modification of select data already in the data bases
- Deletion of select data from task data already in the data bases
- Deletion of entire task data from the data bases

In all cases, on-line updating must be consistent with the categories and structure of the data bases as they have been defined during data definition.

On-line updating requires that a high degree of selectivity be used in order to adequately define those tasks for which updating is to take place. This selectivity is accomplished in the LUCID system by making all of the selective power of the system's retrieval program available for use when qualifying tasks for updating. The retrieval program, discussed in the following pages accepts the updating information as it is entered on-line and determines the task data in a data base that qualify for updating. When all updating information has been completely entered, the retrieval program passes lists of updating information and qualifying task data to the LUCID update program. This program then performs the actual updating of the data base and associated indices. The result is a new data base containing the requested changes. The original data base remains unaltered. Figure 11 presents an illustration of the retrieval and on-line update process.

Due to the relatively time consuming process of entering update information through the keyboard of an on-line teletype console, this type of updating is used primarily to correct errors that are not noticed when the data are originally entered into the data base files. Additions of data on complete new tasks can be handled through on-line updating but not as efficiently as through the batch update process described earlier.

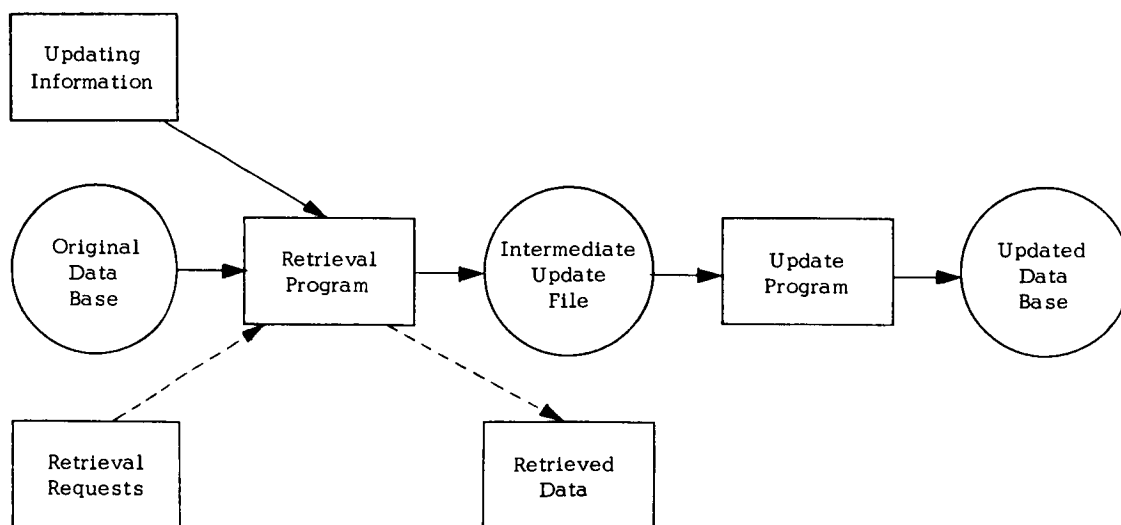


Figure 11. Retrieval and On-Line Updating Process

Data Retrieval

Data retrieval consists of the processing necessary to recover and present specific information from the data base files upon request. This capability is accomplished in the experimental system through the use of the LUCID retrieval program. By using this program, users of the experimental system are able to enter retrieval requests at an on-line teletype console and receive immediate on-line response from the retrieval program. The language for formulating requests is easy to use and provides a wide variety of retrieval options. Detailed discussion and examples of the program's capabilities are contained in Section IV.

Since the data base files for the experimental system are task oriented, retrieval requests are generally concerned with retrieving select information associated with specific task data in the data bases. In addition to task-oriented requests, the retrieval program also accepts requests for descriptions of the data bases, as established during data definition. Lists of unique data values associated with specific categories of data can be obtained in order to provide an insight into the data contained in the files. Simple arithmetic capabilities can also be performed.

Although the retrieval program only accepts queries related to one object system data base at a time, the program provides the capability of easily switching from one data base to another upon request. This capability permits users of the experimental system to retrieve and compare data across the three object systems. In order to facilitate this process, an additional "object system" data base will be developed and made available to the experimental system. The additional data base is referred to as the INDEX data base. Instead of task data, the INDEX data base contains data related to each of the object systems. This enables a user to query the INDEX to obtain reference to the specific object system data bases which will fully satisfy his request.

The categories of information to be contained in the INDEX data base have not been completely identified, but will include such categories as personnel and equipment type. For example, suppose a user is interested in obtaining information regarding tasks in which a particular type of equipment is used. These categories will enable him to first query the INDEX data base and determine the object systems that use the equipment. Then he can query the respective object system data bases for information regarding the tasks in which the equipment is used.

As discussed earlier, the retrieval program is used in conjunction with the update program to perform selective on-line updating.

Current Awareness

The current awareness feature of the pilot study experimental system provides the capability of notifying users of the system whenever information in their

particular areas of interest is entered into the data base files. This process involves the establishment of profiles that express the areas of interest of each user, the matching of the profiles against data as they are entered into the data base files, and the generation of notifications whenever favorable matches occur.

Profile Preparation

Profile preparation consists of manually selecting values from controlled value lists which are established as a result of the concurrent pilot study vocabulary research. Values are selected which best express the areas of interest for specific, as yet unidentified, users of the experimental system. The value lists are related to specific categories of information contained in the data bases. In order that a matching technique can be used between profiles and data base values, information for the data bases must be standardized according to these lists prior to entry into the data base files.

Information that is of interest to a user must be expressed by referencing values for categories that are predictable in content and from which meaningful value lists can be constructed. The initial data bases for the experimental system contain very few categories in which enough standardization of information presentation exists to permit satisfactory matching between profiles and data. Categories identified as potentially useful in expressing areas of interest include the object system, mission phases and segments within a system, equipment, types of personnel, and types of performance, e.g., maintenance, and operational. These categories permit areas of interest to be expressed, for example pertaining to:

- The maintenance tasks involving a particular piece of equipment used on a specific object system
- All tasks in which a particular AFSC participates during preflight operations on a specific object system
- All tasks, regardless of object system, in which a particular AFSC participates

Profile Maintenance

After areas of interest have been determined and expressed by controlled profiles, the profiles are made available to the computer program responsible for matching them with incoming data. This requires that the profiles be stored in computer files. For the experimental system, this is accomplished by the data editing capability provided by the time-sharing system in operation on the AN/FSQ-32 computer. The editing capability permits the establishment, maintenance and presentation of symbolic data files by use of an on-line teletype console. The editing programs provide the experimental system with the capability of:

- Establishing profiles by specific names
- Modifying profiles in response to changes in areas of interest
- Presenting profiles on-line for review
- Saving profiles on tape or disc for use as required

Data Notification

Data notification consists of the computer processing necessary to perform the matching between profiles and new data and to generate notifications when matches occur. The program responsible for data notification will be developed as a part of the research. It will accept, as input, the profiles established by the time-sharing system's editing program and the data values prepared for entry into the data bases through the batch updating process described earlier. The output consists of notifications reflecting matches between profiles and data.

Operation of the data notification program will be accomplished via on-line console. The program will operate in an interactive mode and perform the following functions.

- Request the name of a specific profile upon which matching is to be performed
- Locate the profile on disc storage
- Match the profile against the updating information--also on disc storage and identified by a predetermined name. Matching logic consists of scanning data on each task contained in the updating information and comparing it with the profile. Comparing includes the ability to compare like values within specific categories and the use of AND/OR logic.
- Provide an on-line count of the number of tasks that match
- Provide the option of having the notifications either generated on-line via teletype console or written on tape, printed off-line, and mailed to the user
- Generate a notification for each qualifying task

The content of a notification will include the following information:

- The object system associated with the matching task
- The associated mission phase and segment
- The task name

- A unique reference number specific to the task
- An indication as to whether the task is a new task or is being reentered into the data base as a result of a modification to the task. If a modification, an indication of the type of modification is given.

The unique reference number mentioned above is an identification number that is unique for every task within an object system. These numbers are assigned manually as the task data are prepared for entry into the data base files and are carried in a data base category defined for that purpose. The identification number provides, on each notification, a single identifier by which a user can conveniently qualify the related task for retrieval if more information is desired than is provided on the notification. Retrieval is accomplished as described under Data Retrieval (page 51).

The data notification program accepts the same input as the batch update process. A problem exists, however, for the preparation of notifications for task data that are modified (i.e., values in the task data are being changed) after having been previously entered into the data base files. Normally, this type of updating is performed by the on-line updating process. If the updating is done on-line, however, the changes are not available for use by the data notification program. To remedy the situation, any task data that exist in the data base and are to be considered for data notification as the result of a modification must be: (1) deleted from the data base by means of the on-line update process, and (2) reentered into the resultant data base as complete task data by means of the batch update process. This automatically qualifies it for processing by the data notification program. A category exists in each data base in order to indicate whether the task data are being entered into the data base files for the first time, or whether they are modified task data. If task data are reentered, the unique identification number associated with the task is not changed.

Figure 12 presents an illustration of the current awareness process.

Analysis

Little can be said with respect to the analysis capability provided by the pilot study experimental system. Concurrent pilot study research is engaged in determining whether or not the information planned for entry into the experimental system data bases is amenable to analytical processing, and, if so, to what types. The only specification that can be stated in this area is that the experimental system is to be capable of retrieving information from the data bases and making it available to standard or user-defined analysis routines for further processing, if it becomes desirable to do so.

Analysis routines, if applicable to the experimental system, will either be obtained from the program library that is available through the time-sharing

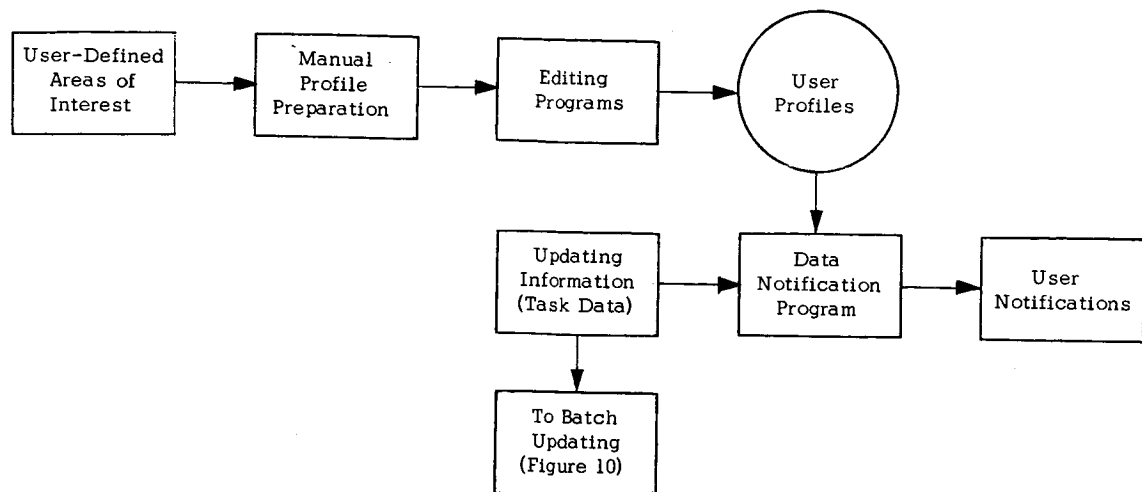


Figure 12. Current Awareness Process

system in operation on the AN/FSQ-32 computer, or developed through use of the JOVIAL compiler or the TINT program.

SECTION IV

FUNCTIONAL DESCRIPTION OF LUCID SYSTEM AS APPLIED TO HUMAN FACTORS TASK DATA

INTRODUCTION

The LUCID (Language Used to Communicate Information System Design) system is a member of the relatively new class of software systems commonly referred to as generalized "information" or "data management" systems. The objective of this class of systems is to provide a readily available computerized capability that can be applied to problems that, in themselves, are not necessarily related, but that share processing requirements common to all data management problems. These problems center around the need to establish, maintain, retrieve, and present large volumes of data. The advantage of these systems is that they allow users, not necessarily knowledgeable in programming or computer technology, to develop automated data handling capabilities that satisfy their own particular requirements in a minimum of time and training. This is accomplished without incurring the cost of designing, developing, and implementing special-purpose software. General purpose systems are not intended to be the answer to all data handling problems. In many instances, user requirements may be such that advantages obtained by the development of special purpose software outweigh advantages offered by an off-the-shelf generalized approach. However, many problems lend themselves to a satisfactory resolution by generalized systems.

LUCID represents one approach toward providing a generalized capability. It was developed as a research project by System Development Corporation (SDC) under sponsorship of the Advanced Research Projects Agency (ARPA), Department of Defense. The system operates on a time-shared IBM AN/FSQ-32 digital computer located at SDC in Santa Monica, California. The system operates within the framework of the SDC/ARPA Time Sharing System, and is referred to as TSS-LUCID (Time Sharing System-LUCID).

The demonstrated ability of TSS-LUCID to satisfy the general types of processing requirements envisioned for the information storage and retrieval aspects of the pilot study led to its selection for support of these requirements. In addition, the availability of the system via teletype link with the Santa Monica computer offered a convenient opportunity to begin experimenting with the pilot study data pool early in the study period.

The following discussion of the LUCID system is presented in two parts. The first part is a system overview and is intended to acquaint the reader with the LUCID approach to data management. The second part discusses the individual program components of TSS-LUCID.

SYSTEM OVERVIEW

TSS-LUCID is a system of programs designed to provide a computer-based means of storing, updating, and retrieving large volumes of user-defined data. Communication between user and system is accomplished through the use of local or remote interactive consoles connected on-line to the AN/FSQ-32 computer. The language used for communication is user-oriented. Few formatting rules are necessary to prepare information for entry into the system. The result is that a user of TSS-LUCID is able to develop an efficient "personalized" on-line data base allowing him to reference his own data in his own terminology. He can do this in a limited amount of time and with a limited degree of system familiarization. A user can accomplish these capabilities in the TSS-LUCID system by:

- Describing the format, characteristics, and relationships of his data to the system
- Presenting the actual data to the system, in accordance with its description
- Querying the resulting data base as required to obtain desired information
- Updating the data base as required to keep its content current and correct

Describing the Data Base

A TSS-LUCID data base is comprised of a collection of "entries." Each entry contains values for a group of related, uniquely identifiable, classes of information that adequately describe the object about which the data base is organized (e.g., personnel and equipment). Assume, for example, that a data base is to contain information associated with the human requirements necessary to fly a large military aircraft. The uniquely identifiable classes of information necessary to describe the operation might include:

- The tasks to be performed in order to accomplish the flight
- The time required to perform each task
- The criticality of each task
- The number of personnel required to accomplish each task
- The types of personnel required to accomplish each task
- The actual steps performed by each person in accomplishing each task

This list is far from complete. Depending on the intended use of the data, an actual description for this type of information might include such elements as

training requirements, levels of proficiency, safety precautions, hardware involved, hardware and human reliability, and costs. The list does illustrate the general types of information that can be entered into the system.

Once the individual classes of information necessary to describe the data base have been determined, the next step is to convey them to the description phase of the system. This is accomplished by providing the system with the following data regarding each class, or element, of information:

- A field number to be used as an identifier when loading actual data values into the system during the load phase
- A unique, meaningful name for the element consisting of one or more words
- A descriptor, indicating the format and characteristics of the data associated with the element as it is to be presented during the load phase
- An optional descriptor modifier, indicating element relationships with each other, or indicating that multiple values for an element can exist in a single data base entry

An example of a description of the previously described human requirements information in TSS-LUCID format is as follows:

```
1  TASK NAME
2  TIME
3  CRITICALITY CATEGORIES LOW AVERAGE HIGH
4  (PERSONNEL NUMBER) MAXIMUM 4
5  (PERSONNEL TYPE) STRING NAME
6  PERFORMANCE NAME
TERM
```

Element number 1, identified as TASK, is described as a NAME element, indicating that data values for this element exist in the form of alphanumeric data, i.e., strings of alphanumeric characters.

Element number 2, identified as TIME, contains no descriptor, indicating that data values for this element are decimal.

Element number 3, identified as CRITICALITY, is described as containing CATEGORIES of values that establish degrees of criticality for a task, and is limited to values of LOW, AVERAGE, and HIGH. Any other values encountered for this element during the load phase will be rejected by the load program.

Element number 4, identified as PERSONNEL NUMBER, is to contain integer information. MAXIMUM 4 indicates that no value greater than four will be

assigned to this element during the load phase. (Perhaps there are only four crew members assigned to the aircraft.) TSS-LUCID demands that all multiple word definitions and values, e.g., PERSONNEL NUMBER, be enclosed in parentheses.

Element number 5, identified as PERSONNEL TYPE, is defined as containing STRING information. This indicates to the system that more than one value may be assigned to the element in the same data base entry. For example, more than one type of person may be required to perform the associated task. In addition, NAME indicates that all values for the element will be in the form of alphanumeric data.

Element number 6, identified as PERFORMANCE, will contain a brief description of the actual human performance necessary to accomplish the associated task, and is, therefore, defined as NAME information.

TERM signifies the end of the description.

In addition to element descriptions, a name for the data base itself must be provided, as well as an entry terminator. The entry terminator will be used during the load phase to distinguish the values for one data base entry from the values to be assigned to the next data base entry.

The completed description is input to the TSS-LUCID translator, either directly on-line from interactive console or from prestored tape. The output of the translator, referred to as a translation, is a tape containing the definition and structure of the data base.

Loading the Data Base

After the data base has been described, the next step is to put data into it. This is accomplished by providing the load phase of TSS-LUCID with the translation tape obtained from the description phase and user-provided values that are to be loaded into the data base. Values are input in fields of data and are entered from tape, disc, or on-line interactive console. Each data field must be immediately preceded by a field number, representing the element with which the data is to be associated, as defined in the description phase. The end of the data fields for an entry is indicated by a word designated by the user as an entry terminator.

Continuing with the previous example, assume that two tasks are to be loaded into the data base. Each is to become a separate entry. The first task is that of obtaining Air Traffic Control (ATC) clearance and taxi instructions from the control tower before take-off of the aircraft. The task takes 2 minutes, has a high degree of criticality, and involves the pilot, copilot, and navigator. The second task is that of taxiing the aircraft into position for take-off. Taxiing takes approximately 10 minutes, has an average criticality, and is performed by the pilot.

The information for the two tasks, arranged in a format acceptable to the TSS-LUCID load program, is as follows:

```
1  (CONTACT CONTROL TOWER FOR ATC CLEARANCE AND TAXI INSTRUCTIONS)
2  2.0
3  HIGH
4  3
5  PILOT          5 COPILOT          5 NAVIGATOR
6  *(COPILOT CONTACTS TOWER: PILOT MONITORS CALL: NAVIGATOR
   COPIES CLEARANCE)
END
```

```
1  (TAKE POSITION ON RUNWAY AND HOLD)
2  10.0
3  AVERAGE
4  1
5  PILOT
6  *(PILOT RELEASES BRAKES, SETS POWER, STEERS TO TAKE-OFF POSITION,
   SETS BRAKES)
END
```

In the above example, the word END has been chosen by the user to indicate the end of data for each data base entry.

The input values are processed by the TSS-LUCID loader and are stored in data base entries according to the data base structure provided by the description phase. In addition, indices are generated to permit efficient access to information stored in the data base proper. The data base and its indices are output onto one or more data base tapes in preparation for use by the other system components.

Querying the Data Base

The inputs to the retrieval phase of TSS-LUCID consist of the data base and indices output from the load phase and user-provided queries requesting that specific information be obtained from the data base. Queries are entered via on-line console in a near-English query language. Program response is immediate, permitting a user to formulate queries on the basis of information obtained from the system as a result of previous queries.

TSS-LUCID queries consist of requests for presentation of one or more element values from those data base entries in which one or more qualifying conditions are satisfied. Qualifying conditions are expressed in logically combined

*

In actual format, text is all on one line.

element/value relational comparisons. If, in the previous data base example, a user is interested in obtaining the names of all tasks in which the pilot performs, the following query;

PRINT TASK WHERE (PERSONNEL TYPE) EQ PILOT

would immediately provide a list of all tasks that satisfy the request. If the user is interested in obtaining not only the tasks, but also the time that each task requires, and, in addition, wants to further qualify the query by limiting the output to those tasks requiring longer than 3 minutes to perform, the query would be;

PRINT TASK, TIME WHERE (PERSONNEL TYPE) EQ PILOT AND TIME GR 3.0

In addition to element/value comparisons, qualifying can be accomplished on the basis of the existence of information in the data base entries for specific elements. (All entries need not necessarily contain the information for all elements.)

Output can be sorted by numeric value if desired and simple arithmetic computations can be performed on elements containing numeric information.

Updating the Data Base

TSS-LUCID provides the capability of updating system-generated data bases to assure that their content is current and correct. Updating is accomplished by either of two methods--dynamic updating or batch updating. Dynamic updating is accomplished by entering the update information on-line in a manner similar to entering retrieval requests and in a language similar to the TSS-LUCID retrieval language. The user has the capability of removing entire entries from the data base, adding new entries, or modifying information in existing entries. In the example data base, a user might want to update the time required to perform a particular task. This could be accomplished by instructing the system to;

CHANGE TIME TO 1.5 WHERE TASK EQ (DISCONNECT GROUND POWER)

or, to remove all entries describing tasks of low criticality;

REMOVE WHERE CRITICALITY EQ LOW

Batch updating provides the capability of adding large volumes of new information to the data base as complete new entries. The new information must first be processed as a separate, but related, data base by the load phase of the system, using the same data base description that was used to load the original data base. Updating is then accomplished by merging the new data base and its indices with the old.

The result in either type of update is a completely new data base. The old data base remains unaltered.

TSS-LUCID PROGRAM COMPONENTS

The TSS-LUCID system consists of five individual computer programs. Each program has been written in the JOVIAL programming language and checked out through the use of software capabilities available within the time shared environment of the AN/FSQ-32 computer. All input/output requirements of the system are handled through the use of I/O routines common to the time sharing system. Together, the five programs consist of approximately 150,000 words of computer instructions and environment. The system programs consist of:

- A translator program, responsible for translating the data base description
- A load program, responsible for loading values into the data base in accordance with its description
- A query program, responsible for retrieving and presenting select information from the data base
- An update program, responsible for dynamically updating the data base
- A merge program, responsible for merging two or more data bases in order to perform batch updating

Translator Program

The system translator program, LUTRAN, is responsible for the translation of the user-provided symbolic data base description into definition tables that are referenced by other system components during their operation. The description itself can be entered by on-line console or from prestored tape. In either case, control of the program is accomplished by on-line communication between user and program, with the user supplying various control information when requested by the program.

As explained during the development of the description example in the system overview, a description consists of a number of element definitions. Each definition consists of specific fields of data (i.e., a field number, an element name, a descriptor indicating the type of data to be contained in the element, and an optional modifier for string information). Each field consists of one or more consecutive alphanumeric characters. Fields must be separated from each other by one or more blanks or spaces. Blanks or spaces can be included in a field by enclosing the entire field within parentheses.

Each element description is input to the translator in one or more teletype lines or one or more 72-column punched cards. Lines or cards of input terminating with a space followed by an asterisk indicate that the description for the element being described is continued on the next line or card of input.

The system overview explained the manner in which elements that are to contain alphanumeric, numeric, or categories of values are defined. String elements, that is, elements that are to contain more than one value in a data base entry, were also covered. When two or more string elements are to exist in an entry, it may be desirable to relate them in such a manner that the values for one element are to be directly associated with the corresponding values of the other elements. Such would be the case if each value for PERSONNEL TYPE in the example was to be related with a corresponding value in another string element such as PERSONNEL TIME. This element would indicate the amount of time worked by each particular type of personnel position during the task. By defining both elements as members of the same "string set," the system will automatically maintain a relationship between corresponding values of the two elements. This permits an association to be made between the types of personnel involved in a task and the amount of time that each type of personnel position worked during the performance of the task.

Elements that are to be defined as members of a set must be input to the translator in a group. Each element in the group is described in a normal manner, except that the descriptor STRING is not included. Each group of elements comprising a set must be preceded with a line or card containing the phrase SSET XX, where XX represents any two-character alphabetic identification for the set. Each set is terminated with a line or card containing the symbol S\$. For example:

```
SSET AA
5 (PERSONNEL TYPE) NAME
6 (PERSONNEL TIME)
S$
```

In addition to accepting descriptions for a new data base, LUTRAN provides the capability for a user to modify a description that has been previously translated by LUTRAN. In this case, the inputs consist of the translation tape containing the description to be modified, and the modifications. Modifications permit single elements or a complete string set of elements to be deleted; new elements to be entered, either individually or as members of new or existing string sets; and existing element descriptions to be replaced with new descriptions. For example:

```
DELETE 3 (deletes the description of element 3)
DELETE SSET XX (deletes the entire string set labeled XX)
INSERT SSET AA
4 (PERSONNEL TYPE) NAME (adds elements to string set labeled AA)
5 (TRAINING REQUIREMENTS) NAME
```

At the onset of operation, LUTRAN provides the user with the option of obtaining an on-line listing of the general rules governing the entering of information to the program. As processing continues, the user is asked to provide such control information as is required by the program. This information includes whether or

not the description is to be entered on-line, the name to be assigned to the data base, whether or not LUTRAN should repeat all of the description inputs so that they can be examined for accuracy, the reel number of the output tape, and so on.

As the data base description is processed, LUTRAN determines the basic structure of a data base entry by determining the amount of computer storage required to accommodate values for the various elements within each entry. The basic structure of an entry is essentially common to all entries for any one description, and is dependent upon the number and types of elements contained in the data base. Storage requirements for each element are determined by the element's definition. Elements that are to contain decimal values are allocated a full word of storage in each entry in order to be carried in floating-point notation. Elements that are to contain integer values are allocated only the number of bits of storage required to accommodate their largest legal value, as indicated by the MAXIMUM descriptor. In order to conserve space, alphanumeric information is not stored directly in the data base entries. Because of the unpredictable and highly variable length of this type of information, elements that are to contain alphanumeric values are only allocated enough storage to accommodate a number that references a table in which all unique alphanumeric values in the entire data base are stored. Similarly, elements that are to contain categories of values, as defined by the CATEGORIES descriptor, are allocated only enough storage to accommodate numeric representations of the various alphanumeric values that are eligible for the element; the numeric values are associated with their actual alphanumeric equivalents through a definition table.

An element need not necessarily contain a value in each data base entry, even though storage space in the entry has been allocated to it. In order for system programs to determine whether or not a value actually exists for an element, LUTRAN automatically defines, for each user-defined element, an associated internal element. The internal elements are each allocated one binary bit of storage in each data base entry, which are set and referenced by system programs in order to indicate and determine whether or not values actually have been loaded into the associated user-defined elements.

If errors are encountered during the translation, LUTRAN notifies the user of the condition that caused the error and permits the particular element definition containing the error to be corrected on-line.

After all element definitions have been successfully processed and the structure of the data base determined, LUTRAN writes the resultant definition tables onto tape in preparation for use by the load program. LUTRAN then notifies the user that the translation has been completed.

Load Program

The system load program, LODA, is responsible for the initial generation of a TSS-LUCID data base. Inputs to the program consist of the data base description

output by the translator program and user-provided values that are to be loaded as data base entries in accordance with the description. Values can be entered from tape, disc, or on-line console. As with the translator program, operation of the load program is controlled by on-line console.

As illustrated in the example in the system overview, values to be loaded must be arranged in fields of data. Each value for an element in the data base must be preceded by the field number assigned to the element when the data base description was processed by the translator program. The definition of a "field" is the same as defined for the translator program, that is, one or more consecutive alphanumeric characters. Fields must be separated from each other by one or more blanks or spaces. If a field is to contain imbedded blanks or spaces, as would be the case with multiple word alphanumeric values, the entire field must be enclosed within parentheses.

Values to be loaded into one data base entry must be grouped together, although not necessarily in field number sequence. The group of values must be followed by the entry terminator that was designated by the user during the translation phase. This is necessary to enable the load program to distinguish between values to be stored in one data base entry and the values to be stored in the next entry.

If input is entered via on-line console, one or more values, each preceded by the appropriate field number, may be entered on a single line of input. However, neither a value or a field number may be split between lines of input.

If input is entered in punched card format, either from tape or disc, columns 1-72 of each card may be used for information. Column 1 of each card is considered to be an extension of column 72 of the preceding card. Other than this, no specific card column requirements exist. Fields may begin in any column and extend for as many columns as necessary. More than one value may be entered per card, and, unlike on-line console input, values may be split between cards. The exception is that the user-designated entry terminator must not be split.

As LODA begins its operations, the user is requested to provide certain control information that is required by the program. This information includes: the reel number of the translator tape output by LUTRAN, the input device to be used for entering values that are to be loaded, and the reel number of the tape on which the completed data base and its indices are to be written.

Once the actual loading process begins, LODA compares input values with the description provided by the user during the description phase. If an error is encountered, the program informs the user that an error has been detected, the condition causing the error, where the error occurred, the contents of the field in error (or the entire entry if desired by the user), and the action taken by the program as a result of the error.

Accepted values are stored in the data base entries, either directly or indirectly through the use of reference numbers, in accordance with the data base structure output by the translator. As the data base is being constructed, a set of completely indexed files, referred to as a concordance, is constructed to facilitate access of information stored in the data base.

Figure 13 illustrates the structure of the concordance and its relationship to the data base. Note that each data base element has associated with it a complete list of all unique values occurring in the data base for that element. Each value is stored only once in any one list, regardless of the number of data base entries in which the associated element actually contains a particular value. For numeric data, the values themselves are stored in the lists. For alphanumeric data, reference numbers representing the alphanumeric data are stored. The alphanumeric values themselves are stored in another table which contains all of the unique alphanumeric values in the entire data base. The value list for each element is sorted low-to-high in order to facilitate searching a list for a specific value.

Associated with the value lists are retrieval lists that are entered via indices obtained from the value lists. A retrieval list is a list of occurrences that indicate, for each data value associated with an element, the data base entries in which the value occurs. The data base entries are referenced indirectly through indices into a directory table which contains the relative storage addresses of the data base entries themselves.

After LODA processes all of its input and generates the data base and its concordance, the program notifies the user that the load operation has been completed. The resultant data base is then ready for retrieval and updating.

Retrieval Program

The system retrieval program, QUUP, is primarily responsible for retrieving and presenting information from the data base in response to user-provided queries. The program also acts as an interface between the user and the dynamic update program. Updating functions of QUUP are described in the discussion of the update program. Inputs to QUUP consist of a data base output by the load or update programs and user requests for information concerning the make-up and content of the data base. Control of the program and all information requests are via on-line console.

As the program begins its operation, it requests the user to identify the data base to be used and to indicate its location by tape reel number or disc inventory name. If the data base is not on disc, QUUP will load it from tape and place it on the disc. From that point, a dialogue is maintained between user and program. The user formulates and enters a request for information, and the program provides immediate on-line response, either in the form of the information requested or an indication that the request was improperly entered and cannot be processed.

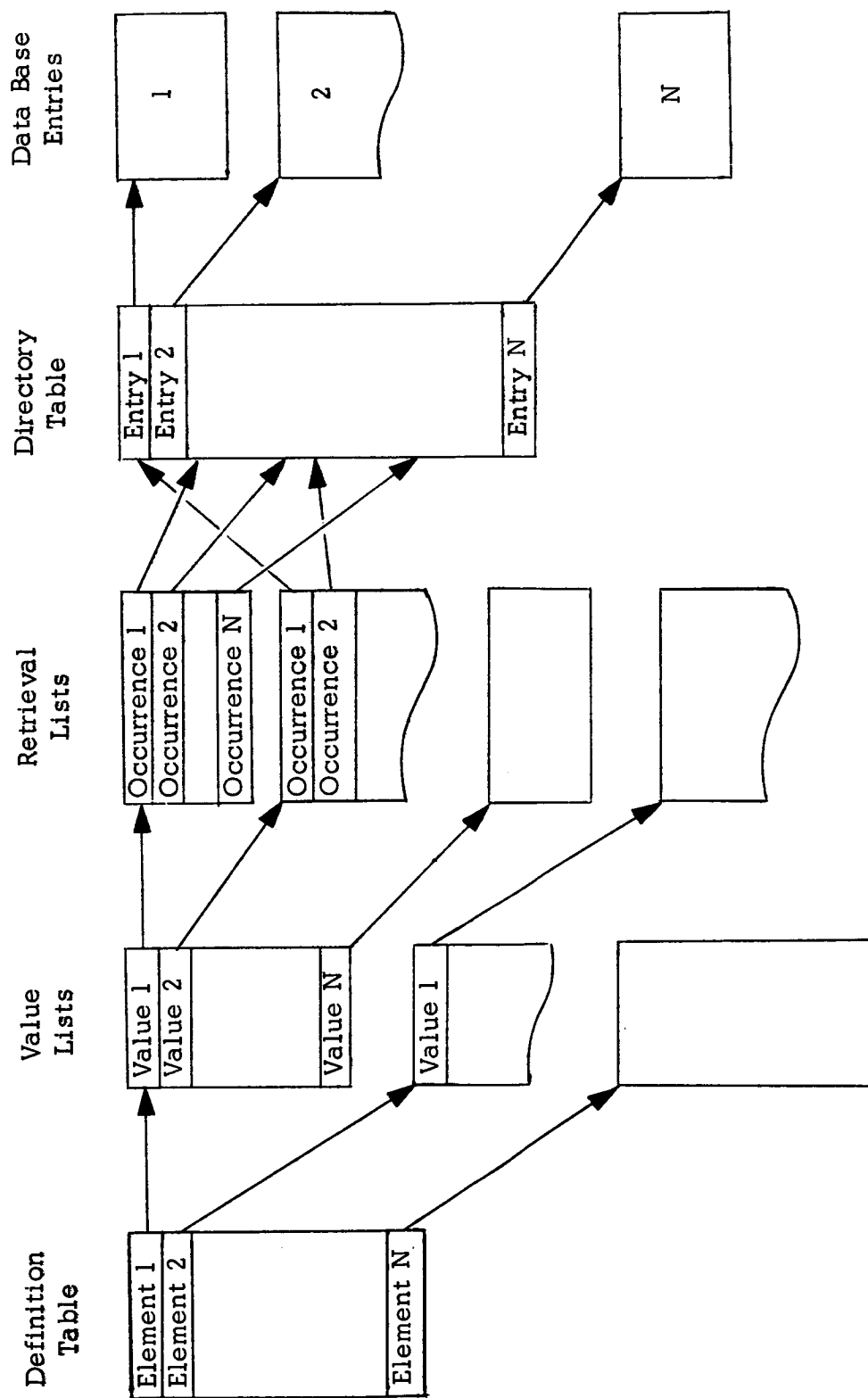


Figure 13. TSS-LUCID Concordance

Requests for information fall into three general categories:

- Requests for a description of the elements contained in the data base
- Requests for all unique values contained in the data base for specific elements
- Requests for specific information contained in the data base

If a user is unfamiliar with the element names and descriptions that were input to the translator program prior to the loading of the data base, he may request that QUUP describe the elements on-line. In response to the command "DESCRIBE," QUUP provides a list similar to the following, with respect to the example data base discussed in the system overview:

<u>SYNONYM</u>	<u>ELEMENT NAME</u>	<u>DESCRIPTION</u>
E1	TASK	NAME
E2	TIME	NUMBER
E3	CRITICALITY	CATEGORY
E4	(PERSONNEL NUMBER)	INTEGER
E5	(PERSONNEL TYPE)	NAME STRING
E6	PERFORMANCE	NAME

The synonyms, or "E" numbers, are element numbers assigned to the user-defined elements by the translator program. They are referred to as synonyms because they may be used in place of actual element names when the user enters requests for information. Similarly, the retrieval program uses "E" numbers when presenting retrieved information so that the user may readily identify output values.

If a user is interested in determining all unique values that occur in the data base for a specific element, he may, for example, enter "SHOW (PERSONNEL TYPE)." QUUP responds with a list similar to:

V1 COPILOT
V2 ENGINEER
V3 NAVIGATOR
V4 PILOT

The "V" numbers are value numbers for the particular element being shown, and are similar to "E" numbers in that they may be substituted for actual values during the entering of requests.

In order to request that specific information be retrieved from a data base, the user must formulate queries consisting of one or more Boolean expressions that indicate relationships between user-defined elements and data base values.

The expressions and relationships define a set of conditions that must exist in data base entries in order for the entries to qualify for retrieval. Expressions may be logically combined by the use of AND/OR criteria in order to permit qualifying on the basis of multiple conditions. A typical query, intended to provide a user with a list of the names and criticalities of each task in which the time required to perform the task is in the range of one to three minutes, would be;

PRINT TASK, CRITICALITY WHERE TIME GQ 1.0 AND TIME LQ 3.0

Relational conditions that can be expressed between elements and values for retrieval qualification include:

<u>Condition</u>	<u>Retrieval Language Notation</u>
equal to	EQ
not equal to	NQ
greater than	GR
greater than or equal to	GQ
less than	LS
less than or equal to	LQ

Since not all data base entries necessarily contain values for all elements, retrieval of an entry can also be qualified by determining whether or not values for specific elements actually exist in an entry. Assume that some of the tasks in the example data base had been loaded without values for the number or types of personnel required to perform the task. (This particular information may not have been known at the time the tasks were entered.) The qualifying expression;

WHERE (PERSONNEL NUMBER) EXISTS

would retrieve only those data base entries in which the personnel information has been entered, while;

WHERE (PERSONNEL NUMBER) FAILS

would retrieve only those in which the personnel information was missing.

Simple computations can be performed with numeric elements in the data base. These include counts, summations, averages, and determination of minimum and maximum values in select data base entries. For example, the average time required to perform all tasks in which the pilot participates can be determined by instructing QUUP to;

AVG TIME WHERE (PERSONNEL TYPE) EQ PILOT

In order to cause QUUP to sort its retrieval output prior to printing, the "BY" command must be used. For example;

```
PRINT TASK BY TIME WHERE . . .
```

causes QUUP to order the retrieved task names in a sequence corresponding to the time, low to high, that each task requires, prior to presenting the information to the user.

There are a number of language forms accepted by QUUP that facilitate the process of entering requests for information via the keyboard of an on-line console. Entering long alphanumerics, either as element names or values, is not only time consuming but increases the possibility of incorrectly entering the information.

The most frequently used form is that of substituting "E" and "V" numbers for element names and their associated value numbers, as obtained by use of the "DESCRIBE" and "SHOW" commands referenced earlier. For example, the query:

```
PRINT (PERSONNEL NUMBER) WHERE (PERSONNEL TYPE) EQ NAVIGATOR
```

can be shortened to;

```
PRINT E4 WHERE E5 EQ V3
```

Certain QUUP commands are repeatable commands. If for example, a query requesting the printing of a number of various elements was to be immediately followed by another query requesting that the same elements be printed, but retrieved on the basis of a different set of qualifiers, the second query could be shortened to;

```
REPEAT WHERE (new set of qualifiers)
```

Similarly, if the qualifiers remain the same, but the elements to be printed vary, the second query could be shortened to;

```
PRINT (new elements) WHERE SAME
```

The processing performed by the retrieval program in response to a query consists primarily of; interpreting the input request, passing the qualifying information through the data base indices in order to determine the entries that satisfy the request, retrieving the entries, and printing the information requested by the user. Consider the query;

```
PRINT TIME WHERE (PERSONNEL NUMBER) EQ 4
```

Through the definition table (see figure 13), the retrieval mechanism converts the name PERSONNEL NUMBER to the address of its value list. This list is

searched for a value of four. A pointer is then followed to the retrieval list which contains all data base entry numbers for entries describing tasks in which four personnel perform. The entry numbers are converted to storage addresses through the use of a directory table, and the qualifying entries are retrieved. The value for the element TIME is then extracted from each and printed.

If the qualifying were to be performed on an element containing an alphanumeric value (e.g., WHERE CRITICALITY EQ HIGH), the process is the same except the alphanumeric value is first located in a list that contains all of the unique alphanumeric values that exist in the entire data base. An associated reference number is thereby obtained and searched for in the value list for that element. Similarly, if information to be printed from qualifying data base entries is alphanumeric rather than numeric, reference numbers obtained from the qualifying entries must be converted to their actual alphanumeric values before printing. This is accomplished by entering the list of alphanumeric values with the reference numbers and locating the associated alphanumeric values.

For more complex queries, involving AND/OR logic, the retrieval process is the same except that the lists of qualifying entries obtained from the individual retrieval lists for the various qualifying expressions are logically combined in order to form a single retrieval list that completely satisfies the request. The resulting list contains only those entries that actually need to be retrieved from the data base. Figure 14 illustrates the manner in which retrieval requests are entered at an on-line console and the manner in which the retrieval program responds.

Update Program

The system update program, UPDATE, is responsible for updating the data base in response to user-provided modifications. Updating is dynamic in the sense that the user enters the modifications via on-line console and they are processed as received. The update program itself has no interaction with the user; the modifications are entered through the retrieval program, QUUP. The retrieval program internally stores the modifications until the user indicates that all modifications have been entered. QUUP then brings in the update program from disc and transfers control to it. UPDATE then processes the information passed to it by QUUP and performs the actual updating. Since modifications are entered on-line through the retrieval program, the language used to express the modifications resembles the language used for retrieval. The modifications may, in fact, be interspersed with retrieval requests. The responses to retrieval requests, however, do not reflect any of the modifications that have been requested until after the data base has actually been updated and made available to the retrieval program.

To add a new entry to the data base, a user must provide values for all elements that are to contain data in the new entry. For example, the request;

ADD TASK IS (TURN ON POWER), TIME IS 1.5, CRITICALITY IS AVERAGE

† SHOW (MISSION SEGMENT)

V1 (PREPARE FOR ENGINE STARTUP)
V2 (ACTIVATE ENGINES)
V3 TAXI
V4 (PREPARE FOR LINE-UP)
V5 (LINE-UP AND TAKE-OFF)

NEXT?

† PRINT TASK WHERE (MISSION PHASE) EQ (PREFLIGHT OPERATIONS) AND *
* (MISSION SEGMENT) EQ TAXI

E7 (RELEASE BRAKES)
E7 (STEER AIRCRAFT AND ADJUST POWER AS REQUIRED TO TAXI)
E7 (CHECK BRAKES DURING TAXI)
E7 (CHECK BRAKES WITH TOGGLE SWITCH IN NORMAL POSITION)
E7 (CHECK ANTI-SKID)
E7 (CHECK HEADING INSTRUMENTS)
E7 (REPORT TAXI CHECKLIST COMPLETE)
E7 (PARK AIRCRAFT)
E7 (ENGAGE BRAKES)

NEXT?

† SUM (TOTAL TASK TIME), AVG (TOTAL TASK TIME) WHERE SAME

SUM E16 16.07

AVG E16 1.78555556

NEXT?

† PRINT (TOTAL TASK TIME) WHERE TASK EQ (CHECK HEADING INSTRUMENTS)

E16 .33

NEXT?

† User Input

Figure 14. Sample Retrieval Request

would add a new entry to the example data base, in which the three named elements would contain the indicated values.

To remove existing data base entries, a qualifying expression, identical to that used for retrieval, must be provided by the user in order to indicate to the program the exact entry or entries that are to be removed. For example;

REMOVE WHERE TASK EQ (RAISE FLAPS)

would remove only the data base entry describing the indicated task. On the other hand;

REMOVE WHERE CRITICALITY EQ HIGH

would remove all data base entries describing tasks that have a high degree of criticality. Care must be taken to word the qualifying expressions to reflect only the entries in the data base that are actually to be removed.

To change element values in existing data base entries, or to assign values to elements that do not contain values, a request similar to the following must be used;

CHANGE TIME TO .73, CRITICALITY TO LOW WHERE . . .

Once again, care must be taken when expressing the qualifying that is to be used, so that only the desired entry or entries are modified.

The update program also provides the capability of inserting, changing, and deleting specific values of string elements. The language form is similar to those described above.

Although the retrieval program accepts the modifications as they are entered, the data base is not actually updated until the user enters the command;

UPDATE (new data base name) TAPE (reel number)

or, to get the data base updated onto disc rather than onto a specific tape;

UPDATE (new data base name) DISC (inventory name)

Upon receipt of the UPDATE command, QUUP loads the update program into computer memory, makes the modification information available to it, and transfers control to it. UPDATE then creates a new data base containing the requested modifications.

Merge Program

The merge program, MERG, is responsible for combining two TSS-LUCID data bases that share the same data base description. The program is used to perform batch updating, that is, additions of large volumes of data that could not efficiently be entered through the on-line QUUP/UPDATE modification process. In order to perform a batch update, a user must first operate the system load program, LODA, on the updating information and obtain an "update" data base. Program MERG can then be used to combine the update data base and the original, or primary, data base.

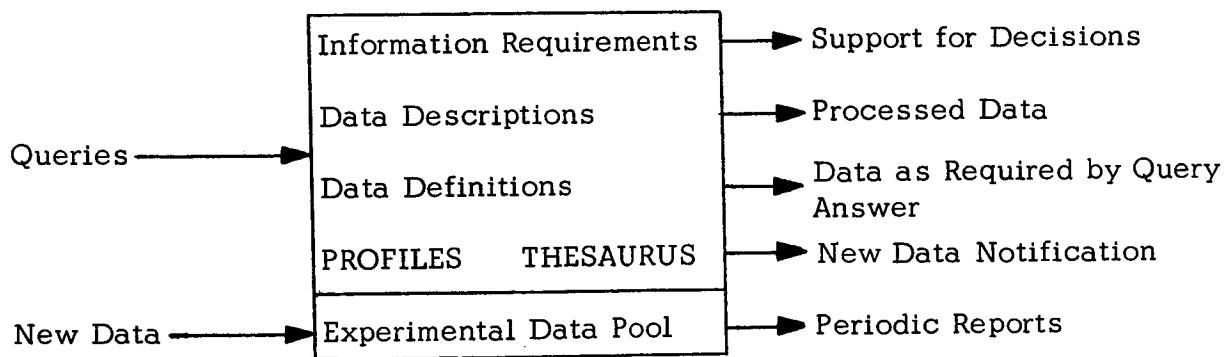
Control of MERG is accomplished via on-line console, and very little is required in the way of user interaction. The only information requested by the program is the location of the two data bases to be combined, and the location where the output data base is to be written. After this information has been provided, MERG combines the entries and associated indices from both input data bases into a single data base containing all of the input information. The two input data bases remain unaltered.

SECTION V

SUMMARY

A time sharing system is important to users and designers of both data management and management data systems. With the accelerated pace of technical and social change, these techniques of rapidly processing information will cease to be the property of a few specialists and become accessible to all. The partnership between man--technician or manager--and computer is still in the early formative stages. Development of computer hardware and software in the hands of computer specialists is a tool that a human factors specialist or manager--the user--requires. Many users will not have the time or talent for detailed training in one or a number of program languages or systems. It will be necessary for the computer specialist to produce a variety of easy to learn software packages; one for technicians, one for engineers, one for secretaries, and one for managers.

The pilot study experimental system is a preliminary attempt to design one of these packages. The heart of this package is the experimental data pool or data base. In order to accomplish anything useful, it is necessary to give the data pool an organization and structure.



The required procedures necessary for the organization of the data pool are as follow:

- Step 1 - Identification of users' requirements
- Step 2 - Selection, from potentially available data elements, of those actually required to satisfy users' needs. To accomplish this, a data definition is required, describing the software data elements in computer terminology--how large they are, what their meaning is, where they are stored, and how one can get at them.

- Step 3 - Identification of data relationships among the data elements. For example, an Air Force Specialty Code (AFSC) is a data element. This data element may be referenced in describing the discrete man-machine actions--procedural steps--required in determining training requirements and training equipment or total object system manning requirements. Thus, the same data element is used to satisfy the data needs of different users.
- Step 4 - Development of a data dictionary or thesaurus. The thesaurus relates various technical meanings of data names to a preferred term. Finally, a list of user interests--profiles--extracted from the thesaurus provides additional information requirements.

The use of the experimental data pool implies a system consisting of input, throughput, and output. Input requires that data be analyzed from three different, but coordinated, aspects. First, the data must be analyzed for data content as described by data elements and items. Secondly, the data must be analyzed to determine the analysis units embedded in the data and to identify their type--nominal, ordinal, interval, or ratio--and application. Thirdly, the data are analyzed to extract the computer data elements of the file structure. The data are finally ready for input into the experimental data pool.

Throughput contains three major processes that include:

- Using the query language to interrogate the data pool for either a specific fact, processed data--applications of specified analytic or simulation procedures to data, or both
- The executive or control concept as an integral part of throughput processing--The executive is the program that calls in the computer program modules necessary to process the user's query.
- The inclusion of data manipulation necessary for: experimental data pool file maintenance, consisting of update to current files, addition of new data, or purging of current files; and system program module maintenance, consisting of program error correction, new program or module addition, or removal of an old program or module

In the pilot study experimental system, the throughput processing is performed with LUCID.

Output provides a response to the user's query in any of the preselected formats available. Output also provides the vehicle by which new data notifications are processed.

The major areas, described in the preceding paragraphs, have been developed as part of the pilot study experimental system. Many other areas have been

researched. These include the necessity for adding data elements or items; more complex analytical processing, statistical processing, mathematical modeling and simulation; the use of cathode ray tubes (CRT), plotters or printers for output; and the control of classified or proprietary data.

LUCID is used as the computer software vehicle. There are known limitations to the use of LUCID as a tool in an operational data processing system. However, LUCID was developed as an experimental tool for applications such as this--the testing of human factors data handling techniques. A natural outgrowth of LUCID, a generalized data management system for operations, is being developed as Time-Shared Data Management System (TDMS). TDMS will accept LUCID type data file structures, eliminating the need for file conversion, and providing many of the data processing capabilities identified as requirements for the pilot study experimental system but unavailable in LUCID.

The experimental system will provide a vehicle for testing the human factors task data processing concepts presently defined. In addition, the system will provide an easier transition into the next design iteration through the use of TDMS.

APPENDIX I

DEFINITION OF DATA ELEMENTS

DATA ELEMENT	DEFINITION
1. Object System	The designator of a specific aerospace system
2. Mission Information	A specific operational maintenance profile or profile segment for the specified object system
3. Equipment Information	Specific data relating to the hardware and software required to accomplish the specified task
4. Performance Description	Specific data relating to the level of detail to be included in the related performance descriptions
5. Performance Characteristics	Specific data relating to the man/machine and man/man interfaces and duties required to accomplish the specified mission or segment
6. Hardware Characteristics	Specific data regarding the human engineering characteristics of the hardware required to accomplish the specified mission or segment
7. Personnel Description	The job title and/or Air Force specialty code of personnel required in the specified performance--any special skills or knowledge required of the performer are also noted
8. Time Information	Specific data regarding performance or mission related time values
9. Remarks	Miscellaneous comments and remarks necessary to explain any material contained in other data elements
10. Source Identifiers	Specific data regarding the origin and author, date of completion or revision, references used by the generators, and security or proprietary restrictions

APPENDIX II

DEFINITION OF DATA ITEMS

DATA ELEMENT	DATA ITEM	DATA ITEM DEFINITION
1.0 Object System	None	None
2.0 Mission Information	2.1 Mission	Descriptive identifier(s) of the object system's design objective(s)
	2.2 Phase	Qualifier of mission descriptor to detail major indentifiable operations
	2.3 Segment	Qualifier of mission phase descriptor to delineate mutually exclusive individual parts
3.0 Equipment Information	3.1 System	Descriptive identifier of major equipment, hardware and/or software, functional groups required to support the successful attainment of the object system's design objectives
	3.2 Subsystem	Qualifier of system descriptor to identify the individual functional equipment groups required to accomplish a man/machine (M/M) performance
	3.3 Components	Qualifiers of subsystem descriptors to identify the independent and dependent equipment combinations required to accomplish a M/M performance--Components will usually be identified by a Federal Stock Number (FSN) .

DATA ELEMENT	DATA ITEM	DATA ITEM DEFINITION
3.0 (cont.) Equipment Information	3.4 Part	Qualifiers of subsystem or component descriptor to identify the specific equipment item required to accomplish a performance--All parts will be identified by a FSN. However, during initial development a contractor's part number will be used pending the assignment of an FSN.
4.0 Performance Description	4.1 Level	Indicator of retrievable details of performance activity from experimental data pool--Reflexive to Mission (2.0) and System (3.0) Information Performance goal or objective
	4.2 Description	4.2.1 Verb - behavioral, descriptive, nonbehavioral Indicator of performance action required, of man/machine system, to accomplish Mission (2.0) or System (3.0) requirements
		4.2.2 Object - Usually simple or compound noun that describe the locus, man and/or machine, of the performance activity
		4.2.3 Modifiers - (Adjectives, adverbs, nouns, and pronouns) Qualifiers of object (4.2.2) used to specify the object of the performance

DATA ELEMENT	DATA ITEM	DATA ITEM DEFINITION
4.0 (cont.) Performance Description	4.3 Procedural Steps	Combinations of verb-object-modifier sets required to describe the individual sequential operations of the man/machine system required to accomplish the level of performance goal described in 4.2
		4.3.1 Verb (see 4.2.1)
		4.3.2 Object (see 4.2.2)
		4.3.3 Modifier (see 4.2.3) Additional specific values of performance--Requirements and/or tolerances permitted will be included to provide structure to the level of specificity required.
5.0 Performance Characteristics	5.1 Location	Specific information identifying the location of the performance being described--Similar performance activities may be required at different places within the aerospace system envelope. Locators may apply to the entire spectrum of system information (3.0).
	5.2 Frequency	A designator specifying the number of times a performance is required per Mission Information (2.0), designator, or per unit time
	5.3 Difficulty	Usually an object system specific (nonstandard) coded entry--The code is assigned, by the data (continued)

DATA ELEMENT	DATA ITEM	DATA ITEM DEFINITION
5.0 (cont.) Performance Characteristics	5.3 Difficulty	generator, on a subjective judgment basis after an assessment of the performance. Difficulties may be generated by: equipment design, personnel limitations, mission information constraints, etc.
	5.4 Criticality	Usually an object system specific (nonstandard) coded entry--The code is assigned, by the data generator, on the basis of a subjective judgment. Criticality factors may be generated by: mission success or failure, levels of equipment degradation, personnel/equipment hazards, performance constraints, etc.
	5.5 Hazards	Statements of an object system specific coded entry describing hazardous conditions--The conditions emanate from object system equipment or operating environmental conditions. Hazardous conditions result from either equipment design and operational requirements or as a result of equipment failure or human error.
	5.6 Training Requirements	Objective statements describing object-system operator or maintenance personnel training requirements
	5.7 Special Tools/ Equipment	Descriptive statements regarding requirements for special tools and/or equipment required for performance completion--The type, description, and part number, if available, will also be included.

DATA ELEMENT	DATA ITEM	DATA ITEM DEFINITION
6.0 Hardware Characteristics	6.1 Accessibility	Descriptive statements or object system specific codes regarding an equipment item's accessibility for operation and/or maintenance during a performance
	6.2 Visibility	Descriptive statements or object system specific codes regarding an equipment item's visibility or legibility for operation and/or maintenance during a performance
	6.3 Manipulability	Descriptive statements or an object system specific code regarding an equipment item's manipulability for operation and/or maintenance during a performance
	6.4 Equipment Status	Descriptive phrases regarding the development state of the equipment used in the performance--These conditions include: mock-up, breadboard, prototype, and production
7.0 Personnel Description	7.1 Type	Descriptive name (job title) of the type of operator/maintenance individual required for the performance; and, if available, the AFSC or other numerical designator--Additional job titles and AFSC's are included for helper/supervisory personnel required.
	7.2 Number	A value signifying the total number of personnel required for a specific performance (operator, maintenance, helper, supervisor)

DATA ELEMENT	DATA ITEM	DATA ITEM DEFINITION
7.0 (cont.) Personnel Description	7.3 Special Skills/ Knowledge	Descriptive statements regarding requirements for performance, specific special skills, and knowledge--Reflexive of statements contained in training requirements (5.6)
	7.4 New Skills/ Knowledge	Descriptive statements relative to an analytical requirement for new skills or knowledge developed from specific performance requirements--Related to statements contained in training requirements (5.6)
8.0 Time Information	8.1 Performance Time	Time values associated with individual performance requirements
		8.1.1 Fractional Performance Time Time values associated with fractional (hand, foot, arm, leg, etc.) performance measurements
	8.2 Total Time	Time values associated with the accumulation of performance time (8.1) from a start-to-stop reference
		8.2.1 Elapsed Time Time values associated with the accumulation of total individual performance activity
	8.3 Time Constraints	Time values associated with constraints that affect the performance accomplishment or initiation, e.g., the performance must be accomplished within X time units, or the performance must stay at Y time units from the start of the mission segment

DATA ELEMENT	DATA ITEM	DATA ITEM DEFINITION
9.0 Remarks		None, except as constraints to other analytical descriptions-- Related to the originating element or item by index number
10.0 Input Identifiers	10.1 Author	Identification of data generator
	10.2 Organization	Identification of data generator contractor/subcontractor
	10.3 Date	Identification of date of preparation of data generation
	10.4 Revision	Identification of the revision of a performance by a data generator
	10.5 Security/ Proprietary	A coded designator of the security/ proprietary classification of the performance and for the document
	10.6 Type of Performance	Descriptor that identifies the general classification of the performance, e.g., operational, maintenance, routine, and continuous
	10.7 References	Designators of the references used by the performance data generator-- Designators of required predecessor or successor performances

APPENDIX III

GUIDELINES FOR DATA EXTRACTION (ALCC)

General (Guidelines Apply to Appendix IV Format Only)

The fundamental task to be performed is to extract the data from the source materials and record it on the forms provided. The information should be recorded exactly as it appears in the source material. Few judgments will be necessary on your part to complete the sixteen items on the form.

Specific Instructions

Number forms sequentially beginning with 001. Place the number in space labeled Index Number.

Block 1 - Will always be "ALCC-AVE"

Block 2 - Will always be 1/25/66

Block 3 - Write Unclassified

Block 4 - Will always be "Boeing"

Block 5 - Document title will always be "ALCC Operational Task Analysis and Timeliness"

Block 6 - Data are found on line #30 of the Task Information Summary (TIS)

Block 7 - Data are found on line #11 of TIS

Block 8 - Data are found on line #1 of TIS

Block 9 - Data are found on lines #3 and 4 of TIS

Block 10 - Extract data from lines #18 and 19 of TIS

- a. Copy each different coded equipment designator from the time-line of the corresponding task time-line. (Examples of coded designators are circles in red on page 2 of Section 2 (Time-lines).)
- b. Each TIS has a corresponding TAWS (Task Analysis Work Sheet). This TAWS is found on the page following the TIS. Extract from paragraph 8 of the TAWS, any controls or equipment that appears with a check (✓) mark. Supplemental equipment information may appear in paragraph 2 of the TAWS; if so, include this information.

NOTE: Tools are not equipment. We are concerned here with equipment components and subassemblies: We are not concerned with conditions under which they are used. (See Block 16 remarks.) Do not duplicate equipments or controls.

Block 11 - Will always be blank

Block 12 - Although the section appears early in the form, it should be completed only after all other sections have been completed. (Proceed to

Block 13.)

- a. Miscellaneous comments necessary to explain any of the other items
- b. Specifically - if paragraph #6 (Special Handling) of TAWS is checked (✓) considerable - record from paragraph #2 of TAWS, the corresponding comment regarding what that special handling consists of.
- c. Record any comments from paragraph #5 of TAWS regarding Consequence of Deviation here.
- d. Any other pertinent information which you feel should be called to our attention regarding this "task."

Block 13 - Copy each step in sequence from Task Time-line of corresponding task.

NOTE: Do not include time information. It is recorded in another block.

- Block 14 -
- a. See line # 8 of TIS for location.
 - b. See line #10 of TIS for frequency.
 - c. See line #17 or TIS for criticality/difficulty.
 - d. See line #22 - 25 of TIS for Training Requirements.
 - e. See paragraph 2 of TAWS for special tools - if none, write none.
 - f. See paragraph 7 of TAWS for Safety Precautions.

NOTE: For this item, please label the information a - b - c - etc.

Block 15 - See line 12 of TIS.

Block 16 - See line 14-15-16 of TIS. Please give time, title, i.e., elapsed time, and numerical value.

At this point, complete Block 12, a - d.

APPENDIX IV

STANDARD INPUT DATA FORM

Aerospace Medical Research Laboratories Human Factors		Index No. 023
Task Data		
1) Object System: ALCC - AVE	2) Date/Revision: 1/25/66	3) Security/ Proprietary Classification: Unclassified
4) Originating Organization: BOEING	5) Author/Document: ALCC Operational Task Analysis and Timelines	
6) Reference: Drawings, flows, TAWS	7) Type of Performance: Operations	
8) Function/Task, Name/Number: (T) Send PLC - B Command 2.4.2.7	9) Mission Information: Perform Airborne Operations Perform 494L Post Attack Operations	
10) System Information: FIGA, 1999, 1997, 1996, 1988 ALCC System 1-LP, 2, 1-LP/2, RO/1/2, 1/3, 1/2 Push buttons, toggle switches, selector switches, thumbwheel selector		
11) Hardware Characteristics: None		
12) Remarks (Indicate Specific Referent Block or Subject): Consequence of Deviation Wrong target		

Form
AMRL-0 89
Aug 66

AMRL Human Factors Task Data		Index No. 023												
13) Performance Description: <table border="0"> <tr> <td> Rotate ADDRESS thumbwheels to insert address of WING/SQUAD/FLT/LF Check ALIGN toggle switch & change to MRT if required Rotate PLC-B TARGET thumbwheel to appropriate target Check DELAY toggle switch & throw to HOLD position if required Rotate COMMANDS selector to PLC-B position Confirm STA 1 selection agrees with message (check PROGRAM display) Throw RADIO PERMIT toggle switch to ON position Receive status: RADIO PERMIT display ON Depress & release message INITIATE button Receive status: IN PROCESS display ON Receive status: TRANSMITTING display ON </td> <td> Detect RF output & coordinate with STA 2 Receive status: MINIMUM COMPLETE display ON Coordinate "beyond range" with pilot navigator Depress & release STOP button Receive status: IN PROCESS, TRANSMITTING & MINIMUM COMPLETE displays OFF Coordinate with STA 2 Throw RADIO PERMIT toggle to OFF position Receive status: RADIO PERMIT display OFF Coordinate, return all switches to normal, & log information </td> </tr> </table>			Rotate ADDRESS thumbwheels to insert address of WING/SQUAD/FLT/LF Check ALIGN toggle switch & change to MRT if required Rotate PLC-B TARGET thumbwheel to appropriate target Check DELAY toggle switch & throw to HOLD position if required Rotate COMMANDS selector to PLC-B position Confirm STA 1 selection agrees with message (check PROGRAM display) Throw RADIO PERMIT toggle switch to ON position Receive status: RADIO PERMIT display ON Depress & release message INITIATE button Receive status: IN PROCESS display ON Receive status: TRANSMITTING display ON	Detect RF output & coordinate with STA 2 Receive status: MINIMUM COMPLETE display ON Coordinate "beyond range" with pilot navigator Depress & release STOP button Receive status: IN PROCESS, TRANSMITTING & MINIMUM COMPLETE displays OFF Coordinate with STA 2 Throw RADIO PERMIT toggle to OFF position Receive status: RADIO PERMIT display OFF Coordinate, return all switches to normal, & log information										
Rotate ADDRESS thumbwheels to insert address of WING/SQUAD/FLT/LF Check ALIGN toggle switch & change to MRT if required Rotate PLC-B TARGET thumbwheel to appropriate target Check DELAY toggle switch & throw to HOLD position if required Rotate COMMANDS selector to PLC-B position Confirm STA 1 selection agrees with message (check PROGRAM display) Throw RADIO PERMIT toggle switch to ON position Receive status: RADIO PERMIT display ON Depress & release message INITIATE button Receive status: IN PROCESS display ON Receive status: TRANSMITTING display ON	Detect RF output & coordinate with STA 2 Receive status: MINIMUM COMPLETE display ON Coordinate "beyond range" with pilot navigator Depress & release STOP button Receive status: IN PROCESS, TRANSMITTING & MINIMUM COMPLETE displays OFF Coordinate with STA 2 Throw RADIO PERMIT toggle to OFF position Receive status: RADIO PERMIT display OFF Coordinate, return all switches to normal, & log information													
14) Performance Characteristics: <table border="0"> <tr> <td>Location</td> <td>Aircraft</td> </tr> <tr> <td>Frequency</td> <td>3</td> </tr> <tr> <td>Crit./Diff.</td> <td>X-C</td> </tr> <tr> <td>Tng. Reqmts.</td> <td>Operator</td> </tr> <tr> <td>Spec. Tools</td> <td>None</td> </tr> <tr> <td>Safety Precautions</td> <td>None</td> </tr> </table>			Location	Aircraft	Frequency	3	Crit./Diff.	X-C	Tng. Reqmts.	Operator	Spec. Tools	None	Safety Precautions	None
Location	Aircraft													
Frequency	3													
Crit./Diff.	X-C													
Tng. Reqmts.	Operator													
Spec. Tools	None													
Safety Precautions	None													
15) Personnel Description: 1416, 3016, 29372C, 1045H	16) Time Information: Time (1) 1.2, (2) 0.8, (RO) 0.1, (3) 0.1 Elapsed Time 1.3 Total Time 2.2													

Form
AMRL-0 89
Aug 66

REFERENCES

- Hannah, L. Duncan, and Reed, Lawrence E., 1965. Basic Human Factors Task Data Relationships in Aerospace Systems Design and Development, AMRL-TR-65-231 (AD 630 638), Aerospace Medical Research Laboratories, Wright-Patterson Air Force Base, Ohio, pp iii + 55.
- Hannah, L. Duncan; Boldovici, John A.; Altman, James W.; and Manion, Raymond, 1965. The Role of Human Factors Task Data in Aerospace System Design and Development, AMRL-TR-65-131 (AD 621 379), Aerospace Medical Research Laboratories, Wright-Patterson Air Force Base, Ohio, pp vi + 87.
- Potter, K. W.; Tulley, A. T.; and Reed, Lawrence E., 1966. Development and Application of Computer Software Techniques to Human Factors Task Data Handling Problems, AMRL-TR-66-200 (AD 647 993), Aerospace Medical Research Laboratories, Wright-Patterson Air Force Base, Ohio, pp viii + 165.
- Reed, L. E., 1967. Advances in the Use of Computers for Handling Human Factors Task Data, SAE Paper 670291, Society of Automotive Engineers, New York, pp 12.
- Reed, L. E. and Wise, F. H., 1965. "Computerized Personnel Subsystem Information." Proceedings of Air Force/Industry Data Management Symposium September 28 - 30, 1965, Ballistic Systems Division, Norton Air Force Base, California, pp 271 + 277.
- Shaw, D. C. and Gopoian, S., 1966. A Foundation Course in Electronic Data Processing, TM-LO-2103/000/00, System Development Corporation, Santa Monica, California, pp 73.
- Whiteman, Irvin R., 1965. The Role of Computers in Handling Aerospace Systems Human Factors Task Data, AMRL-TR-65-206 (AD 631 182), Aerospace Medical Research Laboratories, Wright-Patterson Air Force Base, Ohio, pp ix + 173.
- MIL-STD-721B. Definitions of Effectiveness Terms for Reliability, Maintainability, Human Factors, and Safety, 1966.
- AFSCM/AFLCM 310-1. Management of Contractor Data and Reports. February, 1966.

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author) Systems Development Corporation 2500 Colorado Avenue Santa Monica, California 90406		2a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED	
		2b. GROUP N/A	
3. REPORT TITLE IMPLEMENTATION OF COMPUTER SOFTWARE TECHNIQUES TO HUMAN FACTORS TASK DATA HANDLING PROBLEMS			
4. DESCRIPTIVE NOTES (Type of report and inclusive dates) Final Report, 21 June 1966 - 30 June 1967			
5. AUTHOR(S) (First name, middle initial, last name) A. T. Tulley G. R. Meyer			
6. REPORT DATE September 1967		7a. TOTAL NO. OF PAGES 98	7b. NO. OF REFS 9
8a. CONTRACT OR GRANT NO. F33615-67-C-1036		9a. ORIGINATOR'S REPORT NUMBER(S)	
b. PROJECT NO. 1710			
c. Task No. 171006		9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)	
d.		AMRL-TR-67-127	
10. DISTRIBUTION STATEMENT Distribution of this document is unlimited. It may be released to the Clearinghouse, Department of Commerce, for sale to the general public.			
11. SUPPLEMENTARY NOTES		12. SPONSORING MILITARY ACTIVITY Aerospace Medical Research Laboratories Aerospace Medical Div., Air Force Systems Command, Wright-Patterson AFB, Ohio 45433	
13. ABSTRACT Research leading to the implementation of computer software techniques for handling human factors task data generated in support of aerospace system development programs is discussed. Techniques being explored in this research program are based on the assumption that a user-oriented computerized data system will help draw human factors specialists closer to needed data. The application of such a system will reduce the problem of data accessibility and allow more effective use of data in the system engineering process. Preliminary research leading to proposed data handling techniques is disuccessed. A computerized data handling system to store, retrieve, and process human factors task data is initially implemented through a pilot study. A discussion of the pilot study specification is followed by a presentation of the design specification for a computerized experimental system. The experimental system, referred to as the pilot study experimental system, provides the primary means for demonstrating and evaluating the research results against the original research goals. Computer software descriptions are presented for implementing the pilot study experimental system in a user-oriented environment in terms of information needs of human factors specialists.			

14.	KEY WORDS	LINK A		LINK B		LINK C	
		ROLE	WT	ROLE	WT	ROLE	WT
	Personnel subsystem data Task analysis data Information storage Information retrieval Data processing Automated systems Data management						