AD-A076 106 SYNECTICS CORP ROME N Y F/G 5/2
AUTOMATED AIR INFORMATION PRODUCTION SYSTEM. PHASE I. VOLUME II--ETC(U)
SEP 79 W NEHL , P MOULDER F30602-77-C-0065
UNCLASSIFIED RADC-TR-79-179-VOL-3 NL

1 OF 2
AD
A076106

(12) LEVEL III

RADC-TR-79-179, Vol III (of five)
Final Technical Report
September 1979

AUTOMATED AIR INFORMATION PRODUCTION SYSTEM, PHASE I
Air Facilities Subsystem

Synectics Corporation

W. Nehl
P. Moulder

ROME AIR DEVELOPMENT CENTER
Air Force Systems Command
Griffiss Air Force Base, New York 13441

AUTOMATED AIR INFORMATION PRODUCTION SYSTEM, PHASE 1 - Air Facilities Subsystem

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

RADC-TR-79-179, Vol III (of five) has been reviewed and is approved for publication.

APPROVED: John R. Baumann

JOHN R. BAUMANN
Project Engineer

APPROVED: Howard Davis

HOWARD DAVIS
Technical Director
Intelligence & Reconnaissance Division

FOR THE COMMANDER: John P. Huss

JOHN P. HUSS
Acting Chief, Plans Office

If your address has changed or if you wish to be removed from the RADC mailing list, or if the addressee is no longer employed by your organization, please notify RADC (IRRP), Griffiss AFB NY 13441. This will assist us in maintaining a current mailing list.

Do not return this copy. Retain or destroy.

(19) TR-79-179-VOL-3

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

(18) (6)

1. REPORT NUMBER: RADC-TR-79-179, Vol III (of five)
2. GOVT ACCESSION NO.
3. RECIPIENT'S CATALOG NUMBER

4. TITLE *(and Subtitle)*: AUTOMATED AIR INFORMATION PRODUCTION SYSTEM, PHASE I. ~~Volume III.~~ Air Facilities Subsystem. AD A076105
5. TYPE OF REPORT & PERIOD COVERED: Final Technical Report. 26 Apr 77 — 26 Jul 78
6. PERFORMING ORG. REPORT NUMBER: N/A

(10) (15)

7. AUTHOR(s): W. Nehl, P. Moulder
8. CONTRACT OR GRANT NUMBER(s): F30602-77-C-0065

9. PERFORMING ORGANIZATION NAME AND ADDRESS: Synectics Corporation, 310 E. Chestnut Street, Rome NY 13440
10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: 91370307

(11)

11. CONTROLLING OFFICE NAME AND ADDRESS: Rome Air Development Center (IRRP), Griffiss AFB NY 13441
12. REPORT DATE: September 1979
13. NUMBER OF PAGES: 135

(12) 135

14. MONITORING AGENCY NAME & ADDRESS *(if different from Controlling Office)*: Same
15. SECURITY CLASS. *(of this report)*: UNCLASSIFIED
15a. DECLASSIFICATION/DOWNGRADING SCHEDULE: N/A

16. DISTRIBUTION STATEMENT *(of this Report)*

Approved for public release; distribution unlimited.

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

Same

18. SUPPLEMENTARY NOTES

RADC Project Engineer: John R. Baumann (IRRP)

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

Automated Cartography
Automated Flight Information Product Production
Text Composition and Editing
Electron Beam Recorder Film Output

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

This report details developmental efforts in providing the initial phase of a fully automated Air Information Production System (AAIPS) for the Defense Mapping Agency Aerospace Center. The system is used to produce DOD Flight Information Publications (FLIPS); Navigation/Planning and Special Purpose Charts; Special Products; and the Automated Air Facility Information File. The requirements, functional design and operational considerations of the AAIPS Charting, Air Facilities, and Publishing Subsystems are presented. The principal purpose of the three subsystems is the reduction of the (Cont'd)

DD FORM 1 JAN 73 1473

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*

407 633 LB

Item 20 (Cont'd)

labor (manual) required for the revision and publication of information critical to flight operations and logistical planning. Improvement of response time between receipt of changes to air navigation/air facilities data and the dissemination of new data to all users, is also provided. The Publishing Subsystem permits publications to be produced on electronic equipment and extends the power and flexibility of digital manipulation to the updating and reformatting of publications. The Air Facilities Subsystem provides maintenance of the AAFIF data bases, selective data base retrieval, special report generation and generation of formatted tape files for film negative output. The Charting Subsystem provides capture, revision and output of graphic data appearing throughout the DMAAC Flight Information Publications, through preservation of data in digital form and providing techniques to simplify alteration of the data.

TABLE OF CONTENTS

SECTION I INTRODUCTION

1.0	Background	1
1.0.1	Major Objectives and Management Approach	1
1.0.2	AD Production Overview	2
1.0.2.1	Flight Information Publications	2
1.0.2.2	Enroute Charts and Supplements	3
1.0.2.3	Aeronautical Information Special Products	3
1.0.2.4	Automated Air Facilities Information File (AAFIF)	3
1.0.2.5	Organizational Responsibility	5
1.0.2.6	AAFIF Processing	5
1.1	Purpose	6

SECTION II FUNCTIONAL REQUIREMENTS 8

2.0	Subsystem Requirements and Capabilities	8
2.0.1	Hardware Selection/Operational System	8
2.0.2	AAFIF System Programs	8
2.0.3	On-Line Capability	10
2.0.3.1	Log-on/Log-off	10
2.0.3.2	Display Processing	10
2.0.3.3	Input Processing	10
2.0.3.4	RETRIEVE	10
2.0.3.5	UPDATE	11
2.0.3.6	ADD	11
2.0.3.7	DELETE	11
2.0.3.8	Transaction Log	12
2.0.3.9	Save and Restart Capability	12
2.0.4	Applications Programs	12
2.1	Workload	12
2.2	Data Base	13

SECTION III SUBSYSTEM DESIGN 14

3.0	Design Criteria	14
3.0.1	Table Implementation of Essential Functions	14
3.0.2	Menue Oriented On-Line Control	15
3.0.3	System Security	15
3.1	Structure Overview	16
3.1.1	Major User Software	16
3.1.2	Major System Software	16
3.1.3	On-Line/INFOS Interface	16

TABLE OF CONTENTS (Continued)

3.2	Subsystem Data Base	20
3.3	Subsystem Personnel Functions	21
3.3.1	ADA System Personnel	21
3.3.2	Operator Personnel	21
3.3.3	Definition of Personnel Functions	21
3.3.4	Function Scheduling	22
3.4	Subsystem Software Functions	23
3.4.1	The Air Facilities On-Line System	23
3.4.1.1	Menue-Oriented Displays	24
3.4.1.2	Retrieval Displays	27
3.4.2	System Tables	27
3.4.3	Label Book Tables	29
3.4.3.1	Processing Functions	29
3.4.3.2	Samples of Label Tables	29
3.4.3.3	Entry and Load Programs	34
3.4.3.4	Table Column Definitions	34
3.4.3.5	Page Header Line	39
3.4.3.6	Table Loading	40
3.4.3.7	Computed Header Line Columns	40
3.4.3.8	Load File Sample	41
3.4.3.9	Edit Table Interface	43
3.4.3.10	Check List for System Table Loading	43
3.4.4	Edit Tables	45
3.4.4.1	Implementation Alternatives	45
3.4.4.2	Format Table Rules	46
3.4.4.3	Samples of Format Tables	48
3.4.4.4	On-Line Table Verifications	52
3.4.4.5	Sample of On-Line Test Session	53
3.4.4.6	Format Checks for Geographic Latitude and Country Codes	55
3.4.4.7	Logical Check Tables	62
3.4.4.8	Sample of Logical Checks	63
3.4.4.9	Sample of Computed Inputs	68
3.4.4.10	Boolean Retrieval	73
3.4.4.11	Edit Tables Loaded to the System	75
3.4.4.12	Edit Table Processing	78
3.4.4.12.1	ET - Edit Table Test Package	78
3.4.4.12.2	EPRT - Edit Table Systems Loading	83
3.4.5	Tables for Interactive Communications	84
3.4.5.1	Display Generator	84
3.4.5.2	Direct Access Test File	86
3.4.5.3	Error Communications Display Generator	88
3.4.5.4	Error Text File	91
3.4.6	On-Line Programs	94
3.4.6.1	AFF - Air Facilities Executive	94

TABLE OF CONTENTS (Continued)

3.4.6.2	DISPLAY	97
3.4.6.3	UPDATE	99
3.4.6.4	ADD	101
3.4.6.5	DELETE	103
3.4.6.6	RETRIEVE	105
3.4.6.7	Interactive Modules	107
3.4.6.7.1	Display Generator	107
3.4.6.7.2	Read/Echo Module	108
3.4.6.7.3	Error Check Routines	109
3.4.6.7.4	Application Principles of Interactive Routines	111
3.4.7	Error Messages	112
3.4.7.1	System, Table, Software Error	112
3.4.7.2	On-Line Errors	112
3.4.7.3	List of Implemented On-Line Messages	112
3.4.8	Application Programs	117
3.4.8.1	PØ1ASD (UNIT INDEX)	117
3.4.8.2	PØ9ASD (SUBSAR)	119
3.4.8.3	NAME CHANGE	121
3.4.8.4	HISTDUP	121
3.4.8.5	AIFUPD	121
3.4.8.6	ASSOTW	121
3.4.9	Operating Instructions	121
3.4.9.1	Operational Programs	122
3.4.9.2	Directory Selection	122
3.4.9.3	Table Generation and Loading Programs	122
3.4.9.4	Function Analysis of Logical Tables	124

SECTION IV	CONCLUSIONS AND RECOMMENDATIONS	125
4.0	General	125
4.1	Conclusions	125
4.2	Recommendations	125

LIST OF FIGURES

FIGURE NO.		
2-1	Air Facilities Subsystem Functions	9
3-1	Major User Software	17
3-2	Major System Software	18
3-3	On-Line Software/INFOS Interface	19
3-4	Menue Oriented Display	25
3-5	Retrieval Display	26

SECTION I

INTRODUCTION

A major transition from a batch oriented card and tape processing system to an interactive on-line retrieval and update system was planned and implemented by Synectics Corporation for the Automated Air Information Production System (AAIPS).

The total Synectics effort encompassed three subsystems: Air Facilities, Publishing, and Charting. This technical report describes the Air Facilities subsystem, its background, objectives, implementation concepts, and operational characteristics.

1.0 Background

The Automated Air Information Production System (AAIPS) has been implemented by the Synectics Corporation under contract to Rome Air Development Center for the Defense Mapping Agency's Aerospace Center. The scope of this effort covered the analysis, as well as the design and specification of all hardware and software components comprising each subsystem of AAIPS. The effort included the acquisition and integration of all hardware components and custom software to create a Pilot system during this Phase I effort. A demonstration of all Pilot subsystems operating collectively to manage and process 1/15th of the normal AAFIF work flow constituted the major evaluation criteria for determination of whether to pursue Phase II of the AAIPS program.

1.0.1 Major Objectives and Management Approach

The employed strategy for the development of the Air Facilities subsystem has been that of management by objectives. The establishment of these objectives provided the guidelines by which the system goals have been reached. These objectives were outlined as follows:

- Establish milestones by which progress can be measured
- Provide support to process 1/15 of the AAIPS workload (Phase I)

- o Provide an AAFIF data base, eliminating redundancy and wasted space
- o Provide an on-line interactive capability to update and retrieve data from that data base
- o Retain maximum flexibility for the user to change or redefine essential system functions without future software changes or vendor support
- o Upgrade the built-in machine intelligence to achieve greater information integrity and a better, automatic supervision of on-line transactions
- o Simplify and speed up the bulk of regular operating procedures for major savings of operating costs and faster, up-to-date service

1.0.2 AD Production Overview

The Aeronautical Information Department (AD) of the Defense Mapping Agency Aerospace Center (DMAAC) is responsible for the acquisition, maintenance evaluation and exploitation of aeronautical information to support Defense Mapping Agency (DMA) Aerospace Charts and Flight Information Publications (FLIPS) distributed worldwide. This information is provided to the Department of Defense (DoD) and other agencies and authorized users for flight operations and logistical planning purposes.

The major AD production programs include:

- o DoD Flight Information Publications (FLIPS)
- o Navigation/Planning and Special Purpose Charts
- o Special Products
- o Automated Air Facility Information File (AAFIF)

1.0.2.1 Flight Information Publications

FLIPS products are associated with the following geographical areas:

- o Alaska
- o Pacific, Australia, Asia, Antarctic
- o Canada North Atlantic
- o United States
- o Caribbean South America

- o Europe North Africa Middle East
- o Africa

For each geographical area AFLIPS of the following general types are produced: Planning Documents, Enroute Charts and Supplements and Terminal Procedures.

1.0.2.2 Enroute Charts and Supplements

These publications consist of 83 flights charts and six textual supplements. The kinds of information contained within these documents are:

- o airway system/special use airspace
- o aerodrome data
- o navigational facilities

Collectively, the Enroute Supplements contain 140,000 lines of text and almost 11 million characters of information. Nearly 45,000 update transactions are performed on these documents annually. These transactions represent over 62,000 lines of text changes. The Enroute Supplements for foreign countries also contain sketches of selected aerodromes and heliports.

The Enroute Charts are produced in the large graphic format (20" x 45") and typically require over 800 changes per year.

1.0.2.3 Aeronautical Information Special Products

The special products are of three varieties:

- o Aeronautical Video Mapping
- o Tactical Situation Displays
- o Air Field Diagrams

1.0.2.4 Automated Air Facilities Information File (AAFIF)

The AAFIF system satisfies requests for free world air facility information and to support contingency planning functions of DoD approved user organizations. It is an automated file of evaluated information pertaining to all foreign free world airways. Approximately 46,000 airfield records are currently maintained. Around 2,000 information updates to AAFIF are received

by AD daily. The actual physical size of AAFIF is in excess of 400 million characters. With the old system this information resides on 21 reels of magnetic tape. The informational content of AAFIF is categorized as follows:

- o General Identification and Description
- o Operational Users
- o Navigation Aids and Communications
- o Airfield Description
- o Maintenance and Servicing
- o Special Purpose Equipment Base Services
- o Transportation Weather

Outputs derived from AAFIF source information are recorded on magnetic tape or printed. A total of about 90 different products constitute the scheduled production. Roughly 30 of these products are in the form of digital tapes and the remaining 60 products are hardcopy printed reports. Scheduled product users are:

- o Defense Intelligence Agency (DIA)
- o World Wide Military Command and Control System (WWMCCS)
- o Other U.S. Government Agencies

Currently the yearly average magnetic tape production is about 800 tapes, with an average weekly production of 15 tapes. Peak load activity occurs with the coincidence of varying production cycles and may mount to as high as 100 tapes per week. The volume of hardcopy printed material is in the neighborhood of 700,000 printed pages per year, which averages 13,000 pages weekly and realizes a peak load of about 80,000 pages per week.

AAFIF was initially implemented in 1960. In 1964 a file expansion study was performed which resulted in the first automated file design and the creation of the ASSOTW as output from this file. The implementation of the COBOL-bases system was completed in 1968 on an IBM 7094 computer system. Obsolescence of the IBM 7094 resulted in a conversion to the Univac 1108 which was completed in 1976. The existing AAFIF system includes both the data base and the ADP programs necessary to maintain and retrieve the information within the data base. Present AAFIF content was validated by a user survey in 1974.

1.0.2.5 Organizational Responsibility

The Aeronautical Information Department has several divisions to fulfill its functional role. Of those, ADA, the Automated Systems Division, is charged with the responsibility of processing and maintaining the AAFIF data base and the generation of scheduled data products derived from exploitation of the Air Facilities information. ADA also provides immediate response to ad hoc Special Air Information Requests (SAIRS).

1.0.2.6 AAFIF Processing

Operational control of AAFIF maintenance and applications processing is the responsibility of ADA personnel, who currently submit batch jobs to 1108 computer operators. Completed jobs are returned to ADA for distributing the outputs to appropriate end-user organizations.

The data base updating with the current, batch oriented Univac 1108 system is performed on a weekly basis. Scheduled query products are produced weekly, monthly, quarterly, semiannually and annually. Unscheduled products (SAIRS) are generated as needed. The latter occur at the nominal frequency of around 100 per year. The outputs of 95 percent of the SAIRS are printouts. Very small percentage of SAIRS generate digital tape products. Turn around time on the U1108 for the SAIR production programs is typically 24 hours.

The batch oriented AAFIF updating is initiated by Air Analysts in ADD who prepare special data entry forms for the AAFIF system. The data entry forms are then transferred to the keypunching contractor. Keypunched data cards are returned to ADA for entry into the AAFIF system. The cards are then transferred to magnetic tape by an off-line process through an IBM 1401 system. The transaction file (update card on tape) is the input to the AAFIF batch update process performed weekly. Each transaction is edit checked and verified automatically during the AAFIF updating. A feedback report of AAFIF updates is generated as a by-product of the AAFIF update process and is returned to the Air Analysts in ADD for confirmation. The present update/input system creates delays of several weeks in getting analyst changes into the AAFIF data base.

A major objective for the redevelopment of the Automated Air Information Production System (AAIPS) by Synectics Corporation was the transformation of the current data management system into a fast responding, interactive on-line capability.

1.1 Purpose

The main purpose of the Phase I effort has been the creation of an on-line retrieval and update system that permits interactive dialogs between users and the computer for a more efficient and timely maintenance of the AAFIF. Included in this objective is the requirement for a 'clean' data base in the years ahead.

The latter is of importance for the simple reason that the frequently updated data base information will be extracted for the generation of reports or magnetic tapes that are passed on to other departments of DMAAC, as well as to various other agencies. Since these reports and tapes serve as inputs for decisionmaking or further data processing at these locations, correct information is essential to avoid costly, time consuming, and embarrassing situations.

While 'timeliness' and 'cleanliness' of the data base operations are of supreme importance to the Air Facilities system, a greater operating efficiency is also desired for achieving major operating cost reductions in the future. As the volume tests of the Pilot system indicate, the possible cost savings promise to be of substantial proportions.

A high degree of built-in 'machine intelligence' is desirable for the purpose of controlling the on-line operations through expanded supervisory functions with automatic communications feedback in the case of input error or required user decision. By reducing the 'cost' of input errors and user guidance -- in terms of time delays or loss of data base integrity -- the manual operations become less critical. Format-incorrect or illogical entries just have less chances of getting accepted by the new system. This not only serves the purpose of speeding up the data entries, it also tends to relax the requirements for specific operator experience. The latter provides relief

and greater flexibility to the problems of work assignment and personnel management.

Any automated information system that wants to avoid premature obsolescence must make provisions towards future systems evolution, with the potential for ever increasing sophistication and automation. In the case of Air Facilities this means that many of the currently manual input operations must be expected to become replaced by update information in machine-readable form or by transmissions via direct computer-to-computer interfaces. For this reason a software approach is required that covers the full spectrum from manual inputs with relatively simple edit checks (i.e. human supervision) to most sophisticated editing procedures for automatic entries and computed inputs. By incorporating programs that allow the full range of data correlation and editing sophistication, the purpose of system stability can be achieved regardless in what mix of manual to automated inputs the system is actually used at any given time.

Another major objective of system design is a clear separation between functions for regular, daily operations, such as updating the AAFIF data base (ADA operators), and those of determining and redefining important editing and processing programs for the system itself (ADA systems personnel). The obvious purpose of this functional separation is the greater clarity and management economy with which those functions can be defined, organized, staffed, and supervised.

There is an important prerequisite for retaining system control in areas vital to ADA. This prerequisite demands that important system changes can readily be implemented through data entries in pertinent edit and system control tables, rather than through reprogramming efforts and software changes. The latter are time consuming and require critical steps such as recompilations, link-editing procedures, memory reallocations, and documentations. This usually leads, sooner or later, to a 'roadblock' situation where further system improvements and modifications are prevented. In contrast, it was the expressed purpose of the new design to provide ADA system personnel with effective means to change, mold, and redefine those vital system functions without encountering those obstacles that tend to become insurmountable in time.

SECTION II

FUNCTIONAL REQUIREMENTS

To develop a Pilot system as outlined in the preceding section the following functions had to be implemented.

2.0 Subsystem Requirements and Capabilities

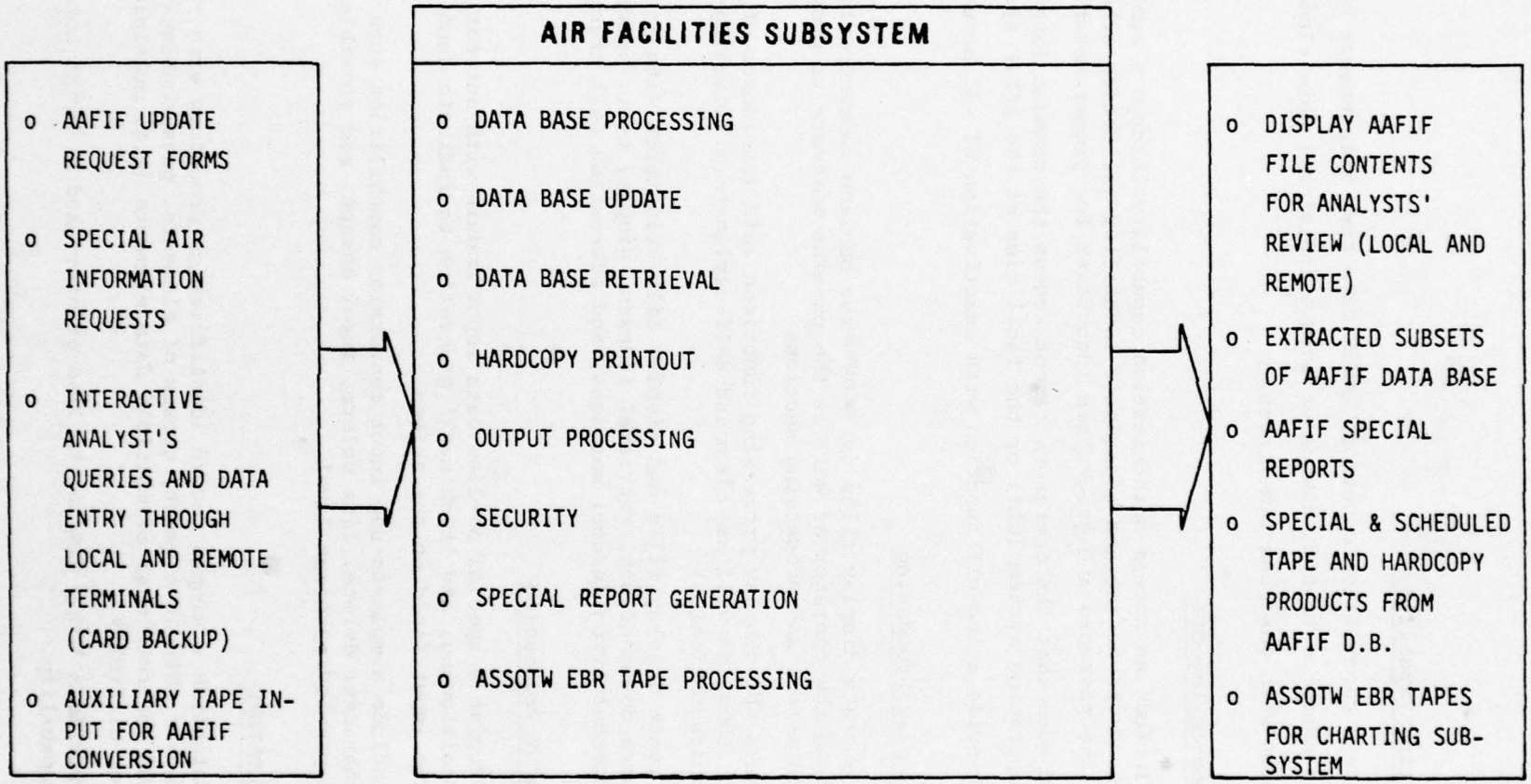
For a summary of the Air Facilities subsystem functions see the enclosed Figure 2-1. This figure describes in a simplified manner the inputs, processing functions, and outputs generated by the system. The following paragraphs provide a breakdown in greater detail.

2.0.1 Hardware Selection/Operational System

- Design a hardware configuration that will meet or surpass the requirements for the subsystem as stated in the Statement of Work
- Provide a matching operational system that meets all requirements of the computer system
- Generate a 'cookbook' for system usage and the maintenance of files

2.0.2 AAFIF System Programs

- Design System Tables for display control, data base element identification, and edit table selection
- Provide capability for establishing password tables and their easy modification
- Develop edit tables for format tests, logical correlations of data base elements, computed (automatic) inputs, Boolean retrievals, password, and Geo-Id correlations
- Provide input programs with table headers and automatic tab settings for the on-line creation of system and edit tables
- Create test programs for the on-line testing and changing of edit tables
- Implement a display session file with hardcopy output for edit table documentation
- Provide a card input program for the 'wholesale' loading of generated edit tables
- Devise programs for formal table checks, load module generation and systems loading



AIR FACILITIES SUBSYSTEM FUNCTIONS

Figure No. 2-1

2.0.3 On-Line Capability

Provide a capability to perform AAFIF data base maintenance by permitting the operator to add, change or delete individual data elements or whole records with ease and timeliness.

2.0.3.1 Log-on/Log-off

- o Include an access authorization capability through a combined password processing and geographic identifier validation scheme. Correlate password and geographic identifier for proper match
- o Ensure that the operators cannot access the command language of the operating system (CLI) or the facilities of the AAFIF system programs
- o Provide a log-off function with reactivation of all access controls

2.0.3.2 Display Processing

- o Generate display files or menues for on-line communications that lead the operator as well as the on-line software to proper display, decision, and processing sections
- o Make the display processing functions safe (avoidance of system crashes) as well as clear and self-explanatory (teaching-machine characteristics)
- o Create display files and display file structures with application area designations, retrieval indexes, display text, header, and communication (error) messages, and subroutine call arguments

2.0.3.3 Input Processing

- o Program a special on-line data input module with automatic cursor positioning, and 'tick mark' generation to indicate (and limit) the input field on the screen
- o Include simple-to-use input conversion capabilities such as character delete, line delete, input escape, and possible exit to a previous program level

2.0.3.4 RETRIEVE

- o Establish a simple record identification procedure with retrieval modes: Single element, groups of elements, page/continuation page, and the retrieval of multiple data elements (e.g. multiple runways per airfield)
- o Provide a 'copy display to line printer' and a 'print total airfield' capability

2.0.3.5 UPDATE

- o Allow for a single element, group of elements, page/continuation page, and 'multiple' on-line update capability. Hold all updates for final validation (logic checks)
- o Perform automatic format checks for each data element entered. Give immediate input error indication with options to repeat or skip the faulty data element input
- o Create a temporary file containing all update information. Ensure that file is saved in case of system failures (e.g. power outages), or can be requested to be saved (in case of update-rejections due to logical checks) for later continuation or update correction
- o Perform logical checks that correlate selected data base elements to make sure that entries fall within reasonable ranges and that right information categories have been chosen (i.e. make sense). Prevent updates that do not pass logical checks from being entered into the data base
- o Provide for an automatic update function (computed input) that is triggered by the update of other data base elements, correlated by logical tables. (Usually for a changed data category that represents a group of related elements)
- o Implement on-line text edit and update capabilities with string search, string replacement, and update verification schemes

2.0.3.6 ADD

- o Program a capability for the on-line addition of an entire airfield to the data base. Assure a guided, page-wise and item-by-item information entry that avoids skipped data fields and human oversights
- o Make the elaborate UPDATE functions available to the ADD process by generally treating the ADD process -- including format tests, logical checks, and computed inputs--as an UPDATE of a (previously 'empty') record

2.0.3.7 DELETE

- o Provide a special password/DELETE function correlation through a system access table in order to ensure that the somewhat dangerous deletion of an entire airfield from the data base can be accomplished only by special, designated personnel
- o Generate log and special output for deleted airfields

2.0.3.8 Transaction Log

- o Maintain a periodic access file containing terminal and airfield information, geographic identifier, date, and time-of-day
- o Keep log of all transactions (UPDATES, ADDS, DELETIONS) as they effectively alter the data base after passing format and logical checks

2.0.3.9 Save and Restart Capability

- o Provide capability for using the temporary file that contains all update information as a SAVE file to continue the update process after a system outage or operator interruption
- o Transfer each data base transaction to disk for safer storage than a file in main memory
- o Implement a deliberate save-file capability to be used by the operator to preserve an incompleted update session. (Transfer of the 'temporary file' to another that will be kept for later recall)

2.0.4 Applications Program

The applications programs deal with a larger section of the data base and are intended to produce reports and magnetic tape outputs to be used by other departments of DMAAC and other agencies.

- o Provide a Boolean search capability (on-line generated) for retrieving specific records/airfields from the data base with prescribed information contents
- o Allow limited sections of the (world-wide) data base to be subjected to the Boolean search capability
- o Provide sort capabilities that allow complex sort key specifications and processing. Implement multi-pass sorting
- o Provide a report generation capability for both hardcopy and mag tape output
- o Process security classifications by inserting pertinent notifications on the output media
- o Accomplish page breaking, header, and title page inserting
- o Include print control with column and line spacing commands

2.1 Workload

The AAFIF data base for the Pilot subsystem must contain approximately 2,200 airfield records in order to create the 1/15 data base volume for the required acceptance tests.

To exercise and test the new system, pertinent transactions are to be collected over a 2-week period for on-line input, function comparisons, and timing. The test material must cover all functions specified and correspond in quantity to the amount of transactions required to demonstrate the processing capabilities of the Pilot system.

2.0.3 Data Base

The data base structure of the Air Facility Subsystem must be designed to accommodate all data necessary for the Pilot maintenance and update functions. The chosen structure must provide ready access to single elements as well as groups of elements. Care has to be exercised to avoid restrictions which would likely cause substantial redesign and conversion problems in the future.

A major effort of the project has to be the reformatting of the AAFIF and the establishment of unique retrieval codes. At present an airfield record contains over 600 data elements that have to be organized in categories and subcategories.

The data base elements can contain regular alpha or numeric fields, narrative text of variable length, or multiple entries (e.g. to accommodate the information for several runways of an airfield).

SECTION III

SUBSYSTEM DESIGN

In Air Facilities a major transition has been made from a previously batch oriented card processing system to an on-line retrieval and update system that permits interactive dialogs between users and the computer for an efficient, up-to-date maintenance of the AAFIF data base. Along with this transition a considerable upgrading of the built-in machine intelligence has been implemented in order to supervise those on-line procedures for correctness in both form and information content. As a result, the AAFIF data base can be expected to become much 'cleaner' in the future, while the requirements for operator experience and qualifications can somewhat be relaxed.

3.0 Design Criteria

The software design has followed a Top-Down and Structured Programming design approach, as far as this was possible--or reasonable--within the given constraints of available main memory, required processing speed, and the inherent features of the chosen language (FORTRAN).

Practicality of program implementation, ease of understanding, and possible modification as requirements change, were of major design concern. While not being dogmatic with regard to a particular programming style, there had been little compromise when it came to the question of retaining control of essential system functions for the user.

3.0.1 Table Implementation of Essential Functions

To assume this control down to the element level of the data base, a series of tables had been created. The implementation of important software functions in table form enables the user to maintain, to expand, and if necessary to restructure the system through data entries alone. By avoiding software changes and reprogramming efforts in areas most vital to ADA, a high degree of vendor independence and system flexibility has been achieved.

This adaptability to changed situations will save cost and time and will prove to be of immeasurable value in the years to come.

3.0.2 Menue-Oriented On-Line Control

Another area of major concern was that of on-line operations. Instead of developing a special (and difficult to learn) command language, a menu-oriented display capability had been chosen to achieve smooth interfaces between the user and the computer. It should guide both operator and the system to any one of the required display, data input, processing, and error checking routines. The man-machine dialog should allow to perform this task in a systematic, safe and proper way such that confusion as well as system crashes could be avoided. It should also be self-explanatory so that it would be possible for an untrained person to become familiar with all operations, just by testing the various options presented to him. In its most elaborate form, the system should be able to fulfill the functions of a (very patient) teaching machine that allows the self-training of personnel.

3.0.3 System Security

Finally, system security, with an interlocking scheme of passwords and other information such as geographic identifiers, had to receive important design consideration. In addition, it was recognized that functional system security is of equal importance and goes much further than access control alone.

A clear separation of functions between ADA system personnel (for maintaining the system and redefining its behavior), and operators (for updating the data base) was indicated in order to better supervise, organize, and safeguard the day-to-day operations.

As it stands now, the operator environment provides all necessary on-line capability for maintaining the data base while preventing him from altering essential system functions (e.g. the specification of required edits).

The latter functions are reserved for ADA's system personnel, and special software packages have been provided for this purpose.

3.1 Structure Overview

The structure of the developed software can be presented on two levels:

- o Grouped by major processing functions
- o As implemented in various programming media such as high level programming language and data base system software

Both types of breakdowns are presented in Figures 3-1 through 3-3.

3.1.1 Major User Software

Three groups of software routines have been implemented as shown in Figure 3-1:

GENERAL: The main on-line procedures with the indicated functions as performed by operators and system analysts to update and maintain the data base. (The crossing - sign in the line to the 'Delete' box indicates that special password restrictions might be desirable for the function in question.)

SPECIAL: Special functions and applications for generating output tapes and reports (preferably during the off-hours since longer processing times are involved).

ON-LINE CONTROL: Supportive software routines for program control of the on-line features. These 'utilities' will be used extensively during all on-line processes. They guarantee on orderly man-machine dialog and minimize the potential for mistakes and formal input errors.

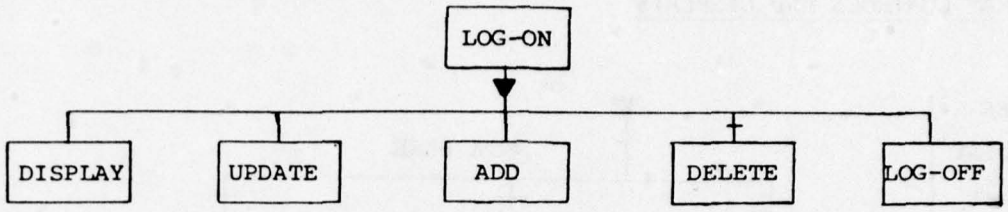
3.1.2 Major System Software

Figure 3-2 indicates the developed software for implementing system tables and defining essential system functions by ADA system personnel. Each group requires programs for entering, displaying, updating (changing), and systems loading of pertinent system control tables.

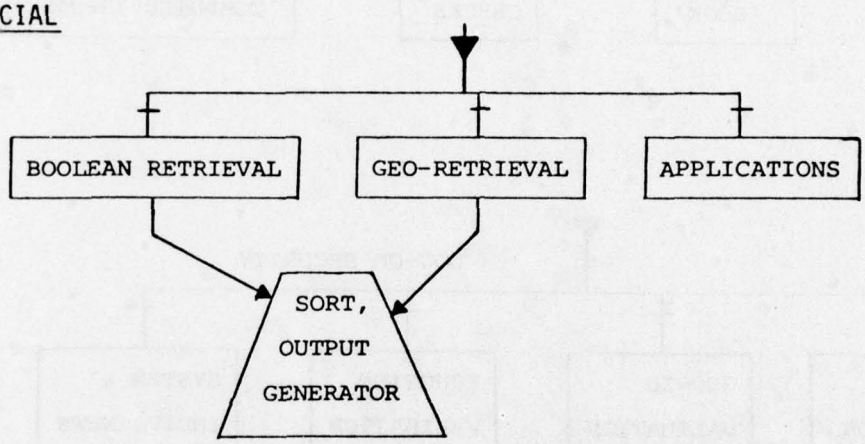
3.1.3 On-Line/INFOS Interface

Figure 3-3 shows the software structure of on-line routines, written in the high level FORTRAN language and their interactions with data base elements and system tables as currently stored in Data Generals INFOS data base management system.

GENERAL



SPECIAL



ON-LINE CONTROL

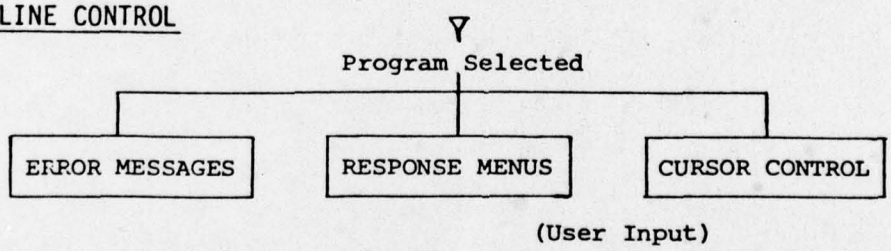


Figure 3-1 Major User Software

SYSTEM FILES FOR TABLE-DRIVEN
PROGRAM CONTROLS AND DISPLAYS

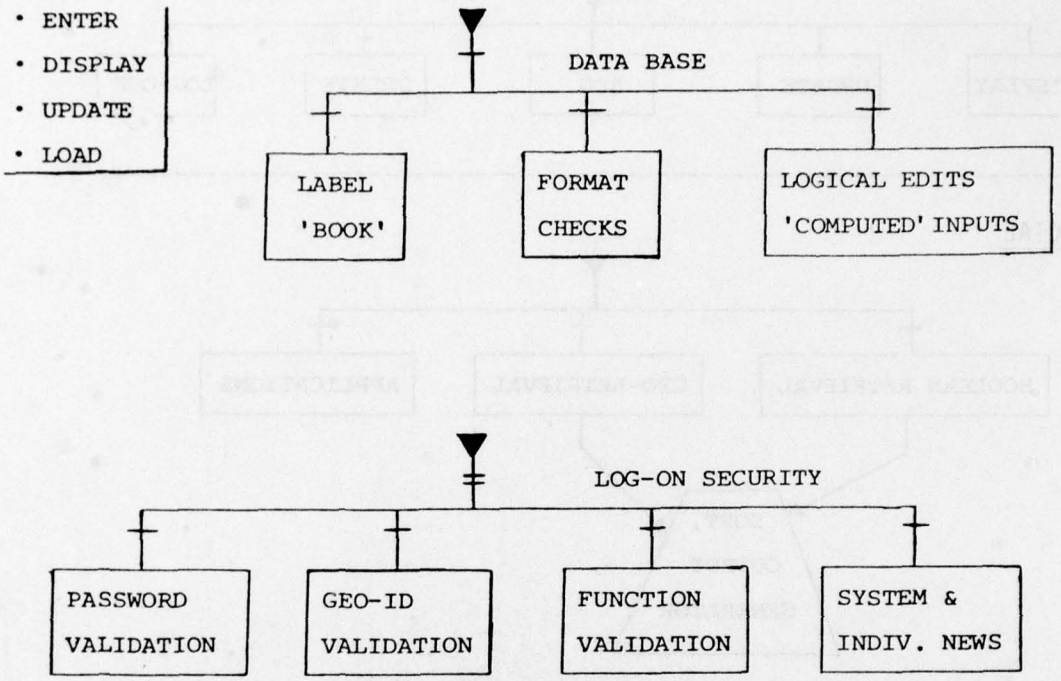


Figure 3-2 Major System Software

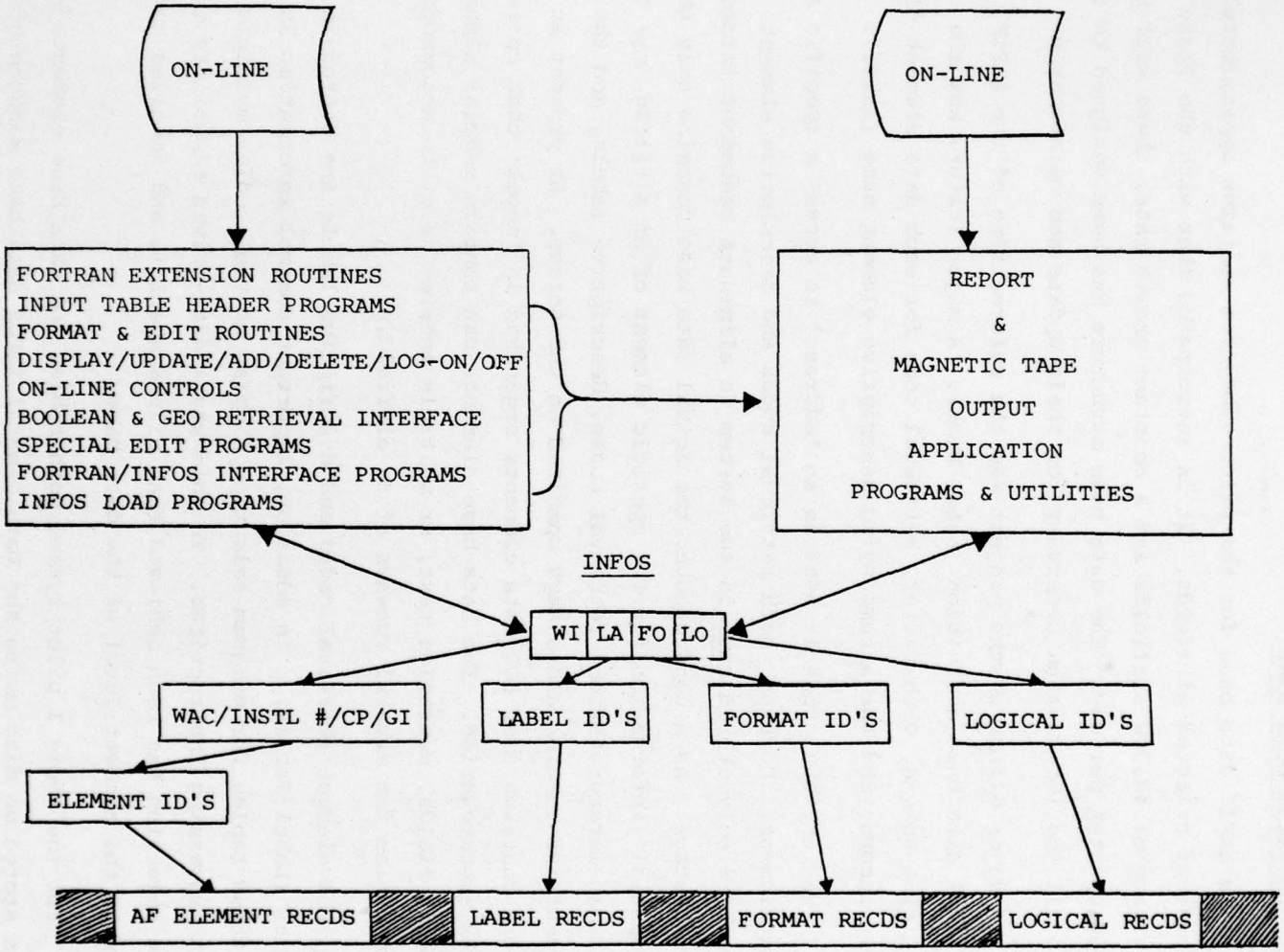


Figure 3-3 On-Line Software/INFOS Interface

3.2 Subsystem Data Base

The AAFIF data base for the pilot subsystem contains approximately 2,200 records of selected airfields. It is anticipated that with the Phase II data base of 48,000 airfields and a constant growth rate, there will be about 2,000 updates per day. The data base structure has been designed to accommodate all the information necessary for their update and maintenance.

A major effort of the project was the reformatting of the AAFIF, resulting in a new data base definition (Label Book). A major feature was the establishment of a unique, 4-character, retrieval code for each data element of an airfield record, and the accompanying descriptive element name (label).

The retrieval code is used as an 'address' to access a specific airfield data element. However, both retrieval codes and descriptive element names are only stored once in the system to eliminate redundant information from storage. As a consequence, the actual data base contains only the 'essential' information for the specific element of an airfield. For the ease of human interpretation, retrieval codes, descriptive labels, and the actual element information are always combined on the screen. At present an airfield record contains over 600 data elements organized in 'pages' that correspond to 82 subcategories. The data base elements can contain regular alpha or numeric fields, narrative text, or multiple entries (e.g. to accommodate the information for several runways of an airfield.)

The element retrieval codes and descriptive labels are stored in special tables (Label Tables). In addition, important control information is included in those tables for program selection, display control, edit, multiple and text processing instructions. In this way a data-driven capability of function selection has been achieved that allows specific and detailed control down to the element level of the data base.

For the Phase I pilot system both tables and data base elements have been stored on disk using the Data General INFOS data base management package as outlined in the previously presented Figure 3-3.

3.3 Subsystem Personnel Functions

For reasons of practicality, manpower allocation, training, supervision, and functional security, the potential for a clear separation of personnel functions has been provided. This has been accomplished by creating special, system oriented programs for loading tables or changing essential system functions, as opposed to on-line capabilities that are data base oriented.

The strict separation of major software functions enables a corresponding separation of personnel functions: ADA system personnel, responsible for maintaining the system and changing its behavior if necessary, and operator personnel for updating and maintaining the AAFIF data base.

3.3.1 ADA System Personnel

The system personnel is responsible for defining essential system functions, for table design, testing and loading, as well as for security implementation, password tables, and other system supervisory functions.

3.3.2 Operator Personnel

The operator personnel and analysts have available at their command all necessary on-line capabilities for maintaining and updating the AAFIF data base. However, they are prevented by the design from altering essential system functions, such as specifications for required edits. Their program framework, as intended, does not allow them to even access and use the common instructions of the Command Language (CLI) provided by the computer manufacturer. This prevents them from bringing the system down, or conducting other undesired manipulations (e.g. requesting outputs of the password tables). Since the latter capabilities, clearly, are in the domain of ADA's system personnel, they have been separated and safeguarded as such.

3.3.3 Definition of Personnel Functions

It is obvious that a software design which follows clear functional objectives, facilitates a corresponding division of personnel functions and responsibilities as well. It makes it easier for management to define

specific rules, to train for them, and to maintain supervisory control over both, personnel and the system.

In cases where a finer subdivision of functions is desired, the inherent capability of Logical Table processing readily allows that 'fine tuning' of functions. For example, suppose that the (somewhat dangerous) DELETE function for deleting a whole airfield record should be safeguarded and performed only by one or a few specially trained persons. The system design allows this special case to be readily implemented. All that is required is a simple expansion of the present password capability of correlating 'password/geo-id' to one of correlating 'password/geo-id/transaction'--in our example the 'transaction' being the DELETE command.

3.3.4 Function Scheduling

In a similar way priority and scheduling problems can effectively be implemented. Since date and time information is always available to the system, a 'password/geo-id/transaction/time-range' correlation could be postulated as well, to restrict the access for certain transactions and persons to certain hours of the day, thus putting a time lock on the system as whether it were guarded like a bank vault.

This time-lock capability, by the way, need not be limited to the purpose of assuring greater internal security (no 'unwitnessed' critical transactions during unusual hours). It could also be used in the reverse for the purpose of excluding certain transactions from being run during regular hours. This could apply to lengthy load operations or certain application programs in order to safeguard fast turn arounds of on-line operations during the main shift hours.

The chosen software structure and the implemented capabilities thus provide a wide range of effective management controls for both system and personnel functions. Their number of combinations is only limited by ideas. They can readily adjust to any level of desired complexity. The negligent implementation costs encourage experimentation with the possible result that in successive adjustments a close-to-optimum division of labor and system functions can be found for maximum payoff and user satisfaction.

3.4 Subsystem Software Functions

The software developments had to achieve a major transition from a previously batch oriented card processing system to an on-line retrieval and update system that permits interactive dialogs between users and the computer for an efficient, up-to-date maintenance of the AAFIF data base. To make a major progress in the implementation of automatic system functions, a considerable upgrading of the built-in machine intelligence had to be accomplished in order to supervise the on-line procedures for correctness in both form and information content. As a result, the AAFIF data base can be expected to become much 'cleaner' in the future. The immediacy of computer responses towards wrong or format incorrect inputs tends to make the manual operations less critical, thereby speeding up and simplifying the update process.

In addition to the on-line capabilities, a number of batch-oriented processing functions had to be implemented for system loading and maintenance, and for the generation of special outputs such as reports and magnetic tapes (Application Programs).

3.4.1 The Air Facilities On-Line System

The on-line Retrieval and Update capability of the Air Facilities Subsystem employs several software modules and series of tables with software characteristics that perform complex functions in a very straight forward and simple to use manner. The implementation of important software functions and their associated data in table form enables the user to maintain, to expand, and if necessary, to restructure the system through data entries. By avoiding software changes and reprogramming efforts in areas most vital to the user a high degree of vendor independence and system flexibility has been achieved. This adaptability to changed situations will make the system more responsive to the changing needs of ADA, and is bound to save considerable amounts of cost and time in the future.

Two basic types of displays are used to attain smooth interfaces between the user and the computer:

- Menu-oriented displays that permit the user to select from a variety of possible options
- Retrieval displays for updating data elements of any record in the system

3.4.1.1 Menu-Oriented Displays

The menu-oriented displays provide the interfaces for the various processing functions such as log-on, password processing, updates, Boolean searches, and so on. They allow a menu-oriented dialog that guides both user and system to any one of the required display data input, processing, and error checking routines. They perform this task in a systematic and safe manner that avoids operator confusion and system crashes.

The implemented display menus of the pilot (Phase I) system have information in great detail so that it is possible for an untrained person to become familiar with all operations just by testing the various options presented to him. In fact, the system can be regarded in this version as a very patient 'teaching machine' that is instructive, responsive -- as well as tolerant against 'dumb' user mistakes. (The latter will trigger feedbacks with requests or suggestions for proper input alternatives). If desired, the inherent flexibility of our table approach facilitates the establishment of a very 'terse' and to-the-point menu sequence as well. Such a condensed and time-saving version could readily be implemented during Phase II for those repetitive operations that have to be performed frequently by experienced personnel.

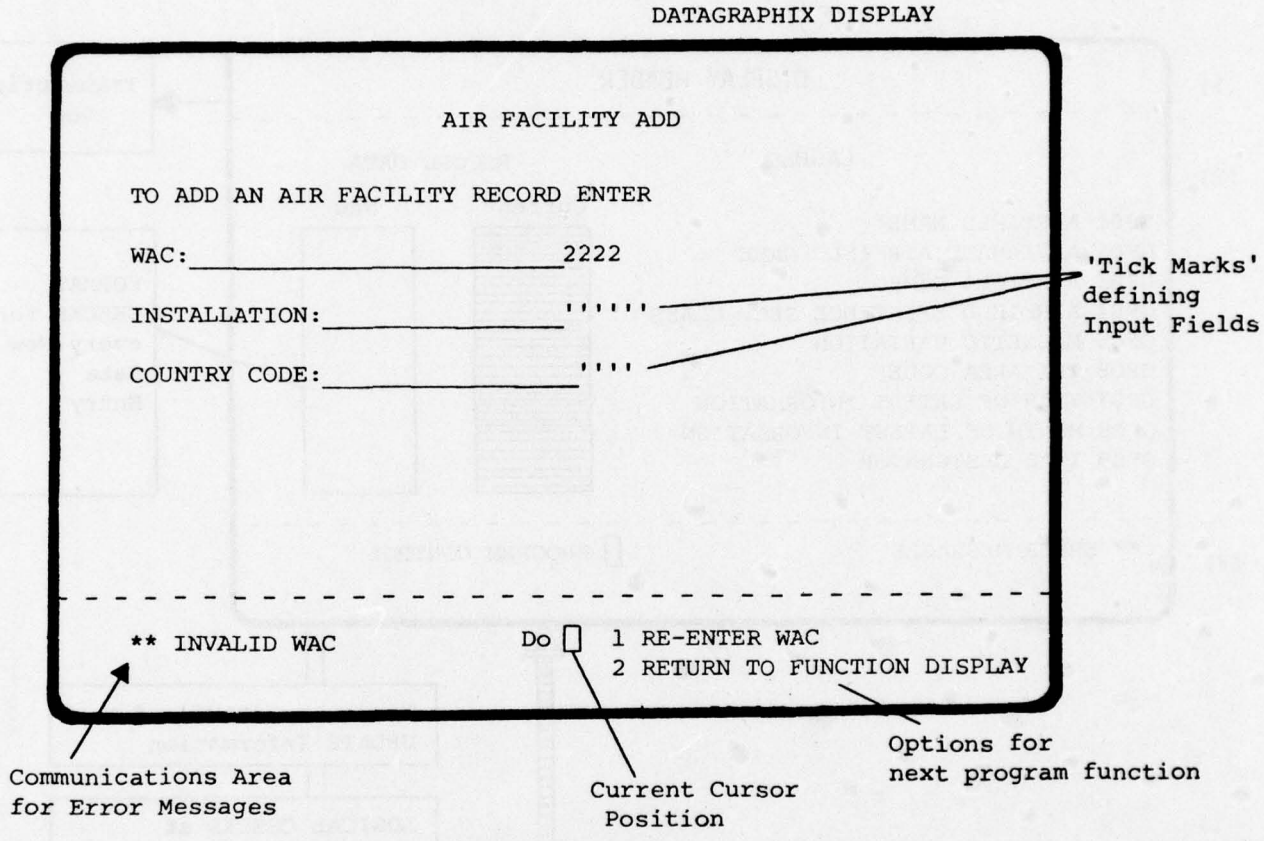


Figure 3-4 Menu-Oriented Display

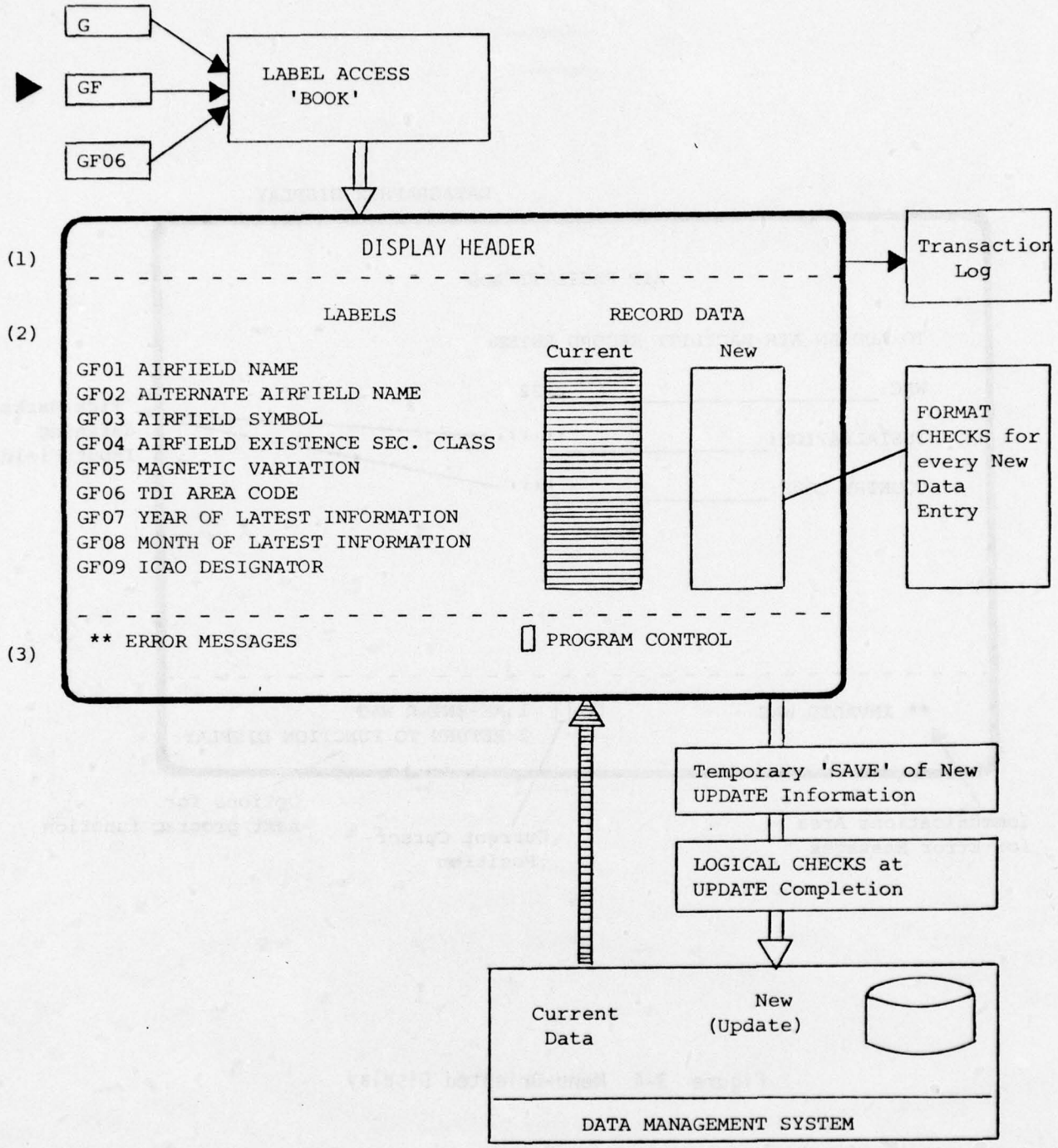


Figure 3-5 Retrieval Display

3.4.1.2 Retrieval Displays

The retrieval displays present small sections of the data base information for on-line review and update procedures. As the accompanying figure shows, the display is graphically divided into three areas:

(1) The Display Header with essential record information (WAC, Inst. No., Country Code/Province), geographical identifier, date and time. The Display Header stays the same during the update transactions of an air field (which might encompass several display 'pages').

(2) The main display portion depicting a section or 'page' of the current data base information with identifying labels, and a data entry field for new inputs or updates.

(3) A communications area for the display of error messages and the input of program control decisions. Whenever an input error is encountered while updating the main section (2), this communications area will be displayed with appropriately selected error messages and response options.

3.4.2 System Tables

Four types of tables are involved in the processing of the retrieval displays:

- Label Tables, containing data base element identification codes, element descriptions (labels), display processing instructions, references and program control information for required format and logical checks.

- Format Check Tables, that specify in detail which specific input data or type of input data, is acceptable as 'new' or 'update' information. The appropriate format check table will be evaluated immediately after each input has been completed. In case of a format failure, the cursor will jump into the communications area (3), with appropriate messages and response options displayed for remedial actions.

- Logical Check Tables. The format-correct update information for an air field will be held in a temporary SAVE file until all inputs for that

air field have been completed. They are then made subject to logical checks that test for valid interrelations between various data base elements.

There are many instances where a format-correct input still could be 'false'-- or make no sense -- if correlated with other information for that air field. This correlation is performed by the Logical Check Tables. Their extensive use can make the system quite 'intelligent' by constantly applying supervisory functions to the actions of a less knowledgeable -- or less alert -- group of users. In cases where, due to complexity, certain inputs should be made by the computer itself (rather than by the human operator), the SET command of the Logical Tables implements these inputs correctly and automatically.

- Communication Tables of the menu-oriented display type, discussed earlier. In case of an input error, or the completion of a page update, the appropriate communication display with messages and response processing options will be presented to the user for his decision on how to proceed.

Although the mentioned tables describe and determine the system behavior in its most important aspects, they remain strictly data driven Any changes, deletions, or expansions of processing functions are readily implemented. They do not require reprogramming and recompilation efforts with associated storage, linkage, overlay, and other inherent system problems that can make the task of change-implementation a time consuming, difficult, and -- for a system that must remain operational -- outright dangerous one. In contrast, a few 'systems people', knowledgeable in the workings of these tables, can now determine and 'massage' the system functions to almost any desired degree of detail and complexity with relative ease and without any outside assistance. Only in the few cases where desired changes of communication tables entail changes of on-line selected processing options as well, programming efforts may be required to accomodate the altered processing sequences.

3.4.3 Label Book Tables

An important effort of the project was the reformatting of the AAFIF, residing on the UNIVAC 1108, into an acceptable data base structure. This effort resulted in a document titled "System Design Plan and Specification for AAIPS; Appendix C, AAFIF Data Base Definition" ('Label Book'). Its major feature was the establishment of a unique, four-character, retrieval code for each data element of an airfield record, and the accompanying descriptive element name (label).

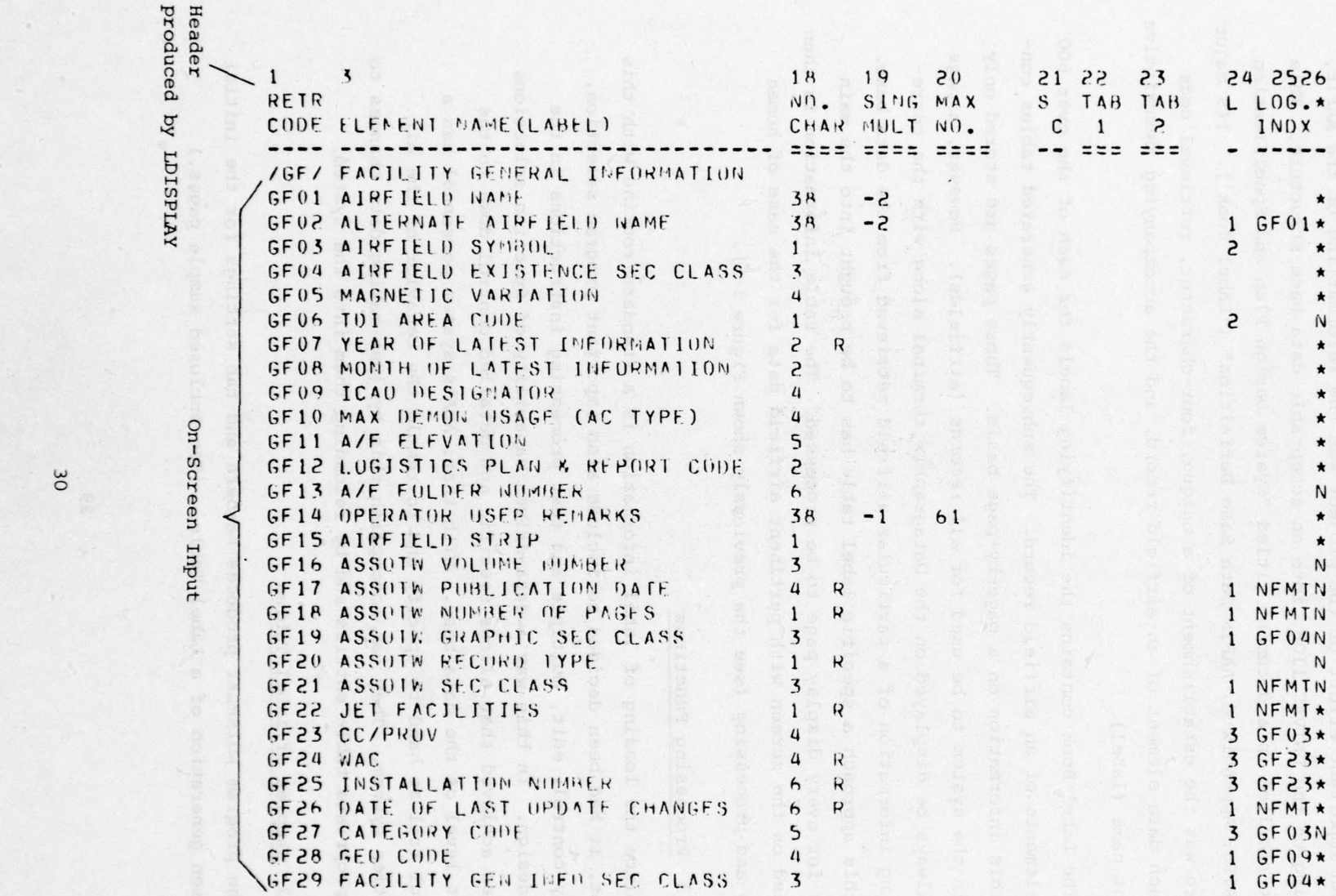
The Label Book contains the identifying labels for each of the over 600 data elements of an airfield record. The subsequently generated tables contain this information on a page-by-page basis. Those pages are stored only once in the system to be used for all records (airfields). However, a page will always be displayed on the Datagraphix terminal along with the corresponding information of a particular airfield retrieved from the data base. With this approach a specific label table has to be brought into the main memory for every display page to be processed. The table information is then combined on the screen with pertinent airfield data for the ease of human review and processing (see the previously shown Figure 3-5).

3.4.3.1 Processing Functions

Since the loading of label information is a standard routine with this concept, it had been decided to include also important program selection, display control, edit, multiple and text processing instructions in the table design. In this way a data-driven capability of function selections has been achieved that allows specific and detailed control down to the element level of the data base. With that, ADA's system personnel has a powerful tool at hand to specify and to remold the behavior of the Air Facilities system. They can accomplish this by just making data changes to the appropriate table entries and by reloading them into the system.

3.4.3.2 Samples of Label Tables

The program LDISPLAY produces headers and tab settings for the initial on-screen generation of a Label Table. (See enclosed sample pages.)



1 RETR CODE	3 ELEMENT NAME(LABEL)	18 NO. CHAR	19 SING MULT	20 MAX NO.	21 S C	22 TAB 1	23 TAB 2	24 L	2526 LOG.* INDX
/GF/	FACILITY GENERAL INFORMATION								
GF01	AIRFIELD NAME	38	-2						*
GF02	ALTERNATE AIRFIELD NAME	38	-2					1	GF01*
GF03	AIRFIELD SYMBOL	1						2	*
GF04	AIRFIELD EXISTENCE SEC CLASS	3							*
GF05	MAGNETIC VARIATION	4							*
GF06	TDI AREA CODE	1						2	N
GF07	YEAR OF LATEST INFORMATION	2	R						*
GF08	MONTH OF LATEST INFORMATION	2							*
GF09	ICAO DESIGNATOR	4							*
GF10	MAX DEMON USAGE (AC TYPE)	4							*
GF11	A/F ELEVATION	5							*
GF12	LOGISTICS PLAN & REPORT CODE	2							*
GF13	A/F FOLDER NUMBER	6							N
GF14	OPERATOR USER REMARKS	38	-1	61					*
GF15	AIRFIELD STRIP	1							*
GF16	ASSOTW VOLUME NUMBER	3							N
GF17	ASSOTW PUBLICATION DATE	4	R					1	NFMTN
GF18	ASSOTW NUMBER OF PAGES	1	R					1	NFMTN
GF19	ASSOTW GRAPHIC SEC CLASS	3						1	GF04N
GF20	ASSOTW RECORD TYPE	1	R						N
GF21	ASSOTW SEC CLASS	3						1	NFMTN
GF22	JET FACILITIES	1	R					1	NFMT*
GF23	CC/PROV	4						3	GF03*
GF24	WAC	4	R					3	GF23*
GF25	INSTALLATION NUMBER	6	R					3	GF23*
GF26	DATE OF LAST UPDATE CHANGES	6	R					1	NFMT*
GF27	CATEGORY CODE	5						3	GF03N
GF28	GEO CODE	4						1	GF09*
GF29	FACILITY GEN INFO SEC CLASS	3						1	GF04*

1 RETR CODE	3 **** LABL.02 **** ELEMENT NAME(LABEL)	18 NO. CHAR	19 SING MULT	20 MAX NO.	21 S C	22 TAB 1	23 TAB 2	24 2526 L LOG.* INDX
/GG/	GEOGRAPHIC COORDINATES							
GG01	GEOGRAPHIC LATITUDE	7						2 *
GG02	GEOGRAPHIC LONGITUDE	8						2 *
GG03	GEOGRAPHIC COORDINATE TYPE	1						2 *
GG04	GEOGRAPHIC LOCATION ACCURACY	1						3 GG03*
GG05	GRID COORDINATES	20						1 NFMT*
GG06	GRID SYSTEM	20						1 NFMT*
GG07	GEOGRAPHIC IDENTIFIER	2						3 GF23*
GG08	WORLD DIVISION CODE	1						*
GG09	BASE REFERENCE POINT TEXT	38	-1	372				N
GG10	GEOGRAPHIC COORDS SEC CLASS	3						1 GF04*

1 RETR CODE	3 LABL.03 **** ELEMENT NAME(LABEL)	18 NO. CHAR	19 SING MULT	20 MAX NO.	21 S C	22 TAB 1	23 TAB 2	24 2526 L LOG.* INDX
/GA/	ATM SOURCE OF COORDINATES							
GA01	ATM WAC	4						2 *
GA02	ATM INSTALLATION NUMBER	6	R					*
GA03	ATM SERIES	3	R					*
GA04	ATM PRODUCER	1						*
GA05	ATM SHEET NUMBER	4						*
GA06	ATM SUFFIX	1						*
GA07	ATM EDITION NUMBER	2						*
GA08	ATM EDITION YEAR	2	R					1 GF07*
GA09	ATM EDITION MONTH	2	R					1 GF08*
GA10	ATM CHART CLASS/HANDLING	3						1 GA10*
GA11	ATM CHART EXISTENCE CLASS	3						2 *

1	3	LABL.19	18	19	20	21	22	23	24 2526	
RETR CODE	ELEMENT NAME(LABEL)		NO. CHAR	SING MULT	MAX NO.	S C	TAB 1	TAB 2	L	LOG.* INDX
/NN/	NAVIGATION/COMMUNICATION AIDS									
NN01	N/C AIDS		1						1	NFMT*
NN02	PRIMARY APPROACH AID		3							*
NN03	PRIMARY OPERATING AGENCY		4							*
NN04	PRIMARY MAGNETIC BEARING		3							*
NN05	PRIMARY DISTANCE		3							*
NN06	PRIMARY CALL OR IDENTIFICATION		4							*
NN07	PRIMARY POWER OR CLASS		6						2	*
NN08	TWR	(NAVAID/COM)	1						3	NN01*
NN09	APP/C	(NAVAID/COM)	1						3	NN01*
NN10	VOR	(NAVAID/COM)	1						3	NN01*
NN11	TACAN	(NAVAID/COM)	1						3	NN01*
NN12	V/TAC	(NAVAID/COM)	1						3	NN01*
NN13	NDB	(NAVAID/COM)	1						3	NN01*
NN14	A/G	(NAVAID/COM)	1						3	NN01*
NN15	D/F	(NAVAID/COM)	1						3	NN01*
NN16	PAR	(NAVAID/COM)	1						3	NN01*
NN17	ILS	(NAVAID/COM)	1						3	NN01*
NN18	ASR	(NAVAID/COM)	1						3	NN01*
NN19	RNG	(NAVAID/COM)	1						3	NN01*
NN20	NAVIGATION/COMM AIDS TEXT		38	-1	620					N
NN21	NAVIVATION/COMM AIDS SEC CLASS		3						1	GF04*

RETR CODE	ELEMENT NAME(LABEL)	NO. CHAR	SING MULT	MAX NO.	S C	TAB 1	TAB 2	L	LOG.* INDX
/NP/	POINT-TO-POINT COMMUNICATIONS								
NP01	TELEPHONE	1						1	NN08*
NP02	TELEGRAPH	1						1	NN08*
NP03	TELETYPE	1						1	NN08*
NP04	CABLE	1						1	NN08*
NP05	CONTINUOUS WAVE	1						1	NN08*
NP06	RADIO TELEPHONE	1						1	NN08*
NP07	PT-TO-PT COMMUNICATIONS TEXT	38	-1	868					N
NP08	PT-TO-PT COMM SEC CLASS	3						1	GF04*

1 RETR CODE	3 ELEMENT NAME(LABEL)		18 NO. CHAR	19 SING MULT	20 MAX NO.	21 S C	22 TAB 1	23 TAB 2	24 2526 L LOG.* INDX*
/AR/	RUNWAYSRRUN SURFACE	(LOW END)	3	1	32				*
AR01	RWY SEQUENCE CODE		1	1	32				*
AR02	RWY LENGTH		5	1	32				3 GF27*
AR03	RWY WIDTH		5	1	32				1 AR02*
AR04	RWY SURFACE COMPOSITION		3	1	32				*
AR05	RWY SURFACE CHARTING CODE		1	1	32				3 GF27*
AR06	RWY CONDITION		1	1	32				*
AR07	RWY AIRCRAFT CAPACITY		4	1	32				*
AR08	RWY ESWL CAPACITY		6	1	32				N
AR09	RWY PSI CAPACITY		3	1	32				2 N
AR10	RWY LCN		3	1	32				2 *
AR11	RWY LCN VALIDITY		1	1	32				2 *
AR12	RWY EXTENSIBILITY		4	1	32				*
AR13	RWY LIGHTS		1	1	32				N
AR14	RWY HEADING	(LOW END)	3	1	32				*
AR15	RWY ARREST SYSTEM	(LOW END)	1	1	32				*
AR16	RWY ELEVATION	(LOW END)	5	1	32				*
AR17	RWY OVERRUN SURFACE	(LOW END)	3	1	32				*
AR18	RWY OVERRUN LENGTH	(LOW END)	4	1	32				2 *
AR19	RWY OVERRUN LCN	(LOW END)	3	1	32				2 *
AR20	RWY APP OBSTR CL	(LOW END)	2	1	32	9			N
AR21	RWY GRADIENT		5	1	32	1-			N
AR22	RWY APPROACH LIGHTS		1	1	32				N
AR23	RWY HEADING	(HI END)	3	1	32				*
AR24	RWY ARREST SYSTEM	(HI END)	1	1	32				1 AR15*
AR25	RWY ELEVATION	(HI END)	5	1	32				*
AR26	RWY OVERRUN SURFACE	(HI END)	3	1	32				1 AR17*
AR27	RWY OVERRUN LENGTH	(HI END)	4	1	32				1 AR18*
AR28	RWY OVERRUN LCN	(HI END)	3	1	32				1 AR19*
AR29	RWY APP OBSTR CL	(HI END)	2	1	32				1 AR20N
AR30	RWY GRADIENT		5	1	32				1 AR21N
AR31	RWY APPROACH LIGHTS	(HI END)	1	1	32				1 AR22N
AR32	RWY SEC CLASS		3	1	32	9			1 GF04*
AR33	RWYS REMARKS		38	-1	868				N
AR34	RWYS SEC CLASS		3						1 GF04N

3.4.3.3 Entry and Load Programs

LDISPLAY - Program that produces headers and tab settings for the on-screen generation of Label Book pages.

PLABL - Program for printing out the Label Book (currently 82 pages).

CLBL - Program to process all label pages by devising and making computed inputs to each table, by creating multiple displays from oversize pages, and by producing a file 'INLABEL' which is used for loading the processed tables to INFOS.

The contents of the Label Book Tables control and generate programmed functions as if those functions were implemented through specific software routines.

3.4.3.4 Table Column Definitions

The columns of the table (one 16 bit computer word per column) are used as follows:

Col. 1-2: Retrieval code for data element.

Col. 3-17: Description (identifying label of data element)

Col. 18: Number of characters to be displayed (left adjusted). If right-adjustment is desired, e.g. for numerical values, the letter'R' will be added in the fourth input position of column 18. The maximum display width for a text item is 38 characters. (For larger text fields see the following two items).

Col. 19: A positive number indicates the start position (or element-identification number) of a multiple of which the element in question is a member. Since multiples should always be presented together as a group (in order to avoid wrong data interpretations or faulty updates) the proper entry in this position will assure correct display processing of those multiples.

A negative number (-1) in this column means that the data base element in question is a text field. The information will not be displayed (due to space problems that otherwise would be encountered

with the regular retrieval display). The negative number merely causes a 'TEXT' indication to be shown. At the same time, however, text processing routines for these data fields are made available in separate displays if the user wishes to review or update such textual information.

Those data fields that carry a (-2) in this column are sufficiently short that they could be displayed on the regular retrieval displays -- using both 'CURRENT' and 'NEW' columns (desirable Phase II extension). The retrieval display would show the contents of this field. However, updating (i.e. the entering of 'NEW' information) would require a switch to the regular text processing display.

A zero or blank entry in Column 19 indicates a regular, single data element.

Col. 20: This column is evaluated only in conjunction with Column 19. If Column 19 is a positive number (indicating an element that belongs to a multiple group) then the number in Column 20 determines the maximum number of multiples allowed.

If Column 19 is a negative number (indicating a text field) then the number in Column 20 describes the maximum number of characters allowed in this text field.

Col. 21: This column is used for line display control purposes. The first character is a line space control number (S) that can be any number from '0' or 'space' to '9'. The second character of this column is the display character (C) which is to be used repetitively to form a graphic line. In case of a '0' or 'space', the next line will be used for displaying a subsequent entry. If S is between '1' and '8', the next entry line will be preceded by 1 through 8 line spacings. The last of these inserted line spacings will carry a character symbol (specified in C) repetitively. This forms a graph line consisting of that symbol.

In the case of S=9, the data entries are followed by a line spacing, again, using the symbol specified in C.

Example: In order to highlight and to separate a multiple group from single element items, a column entry of '1-' for the first item of the multiple group and a column entry of '9-' for the last entry of that multiple would achieve the desired display effect.

```
                                                    Col.
                                                    20
                                                    SC
    SINGLE DATA BASE ELEMENT            XXX
    ANOTHER SINGLE ELEMENT              XXX
    - - - - - - - - - - - - - - - - - - - - -
  2 FIRST ELEMENT OF A MULTIPLE         XXX         1-
  2 OTHER ELEMENT OF A MULTIPLE         XXX
  2 OTHER ELEMENT OF A MULTIPLE         XXX
  2 LAST ELEMENT OF A MULTIPLE          XXX         9-
    - - - - - - - - - - - - - - - - - - - - -
    SINGLE DATA BASE ELEMENT            XXX
```

The proper length of the inserted symbol line is computed and may vary from one display page to another.

Col. 22/23: Tab settings for the display start positions of label and data base information. These tab settings are computed to achieve a pleasant display arrangement, taking into consideration the maximum field width of data base elements in a page, left or right adjustment of the field, and a pertinent centering algorithm.

In special cases, manually inserted tab settings may be used. This has the effect of overriding the computed values.

Col. 24: This column is used for edit processing control purposes. The control numbers must be chosen with great care since they determine not only the extent of format and logical checks for a given data element, but also the required edit tables that must be accessible by the system. All specified edit tables must be internally correct and completely loaded without exception. Otherwise, serious processing and system malfunctions are to be expected.

Edit Program Control Number

▼

L	Format Checks	Logical Checks	
(Blank)	Own Table	No Logical Checks Required	
1	Different Table	No Logical Checks Required	*
2	Own Table	Own Table	
3	Own Table	Different Table	*
(4)	Not Used		
5	Different Table	Different Table	*

* Retrieval code of 'different' table must be listed in Columns 25, 26

As the above outline shows, only the numbers '0' or 'blank' and 1, 2, 3 and 5 may currently be used as edit controls. They indicate which edit tables are to be retrieved by the system to perform the required edit checks.

In the case of <u>even</u> control numbers only 'Own Tables' are involved, meaning that the name of the specific format or logical table is identical with the element-id listed in Columns 1 and 2.

In the case of <u>odd</u> control numbers 'Different Tables' are involved. Their name is listed in Columns 25, and 26. It means that an edit table, established for a different data base

element, can also be used for the element in question.

This 'indirect table addressing' feature cuts down on data loading and storage requirements. It is of particular value in logical checks that bring an extraordinary high level of machine intelligence to the problems of supervising user transactions, or even generating these transactions automatically (computed inputs). Since logical checks correlate the entries of several data elements, it is possible to refer all elements involved to one 'master' table for testing. In this way pertinent logical test procedures can be triggered by the update of any one of these elements rather than by the obvious 'main' event(s) alone. Although the table itself may show a highly structured processing hierarchy, the lowliest member of that hierarchy can set off the required test processing.

In a very simple and straight-forward manner a systematic and comprehensive multi-entry capability to complex processing procedures can thus be established.

Col. 25/26: Table name (index) of 'different' Format or *Logical Check table to be* used for the data element in question.

24 L	25 26 Index	Meaning:
1	GFØ4	Use FORMAT table GF04
3	LA15	Use LOGICAL table LA15
5	XXØ1	Use FORMAT table XXØ1 and LOGICAL table XXØ1

The entry NFMT (no format) will be used for those data elements that have no edit checks at all (e.g. fields with computed inputs).

Last Entry: * - Element used for U.S. limited data airfields.

N - Element not being utilized for 'U.S. Facilities' or 'U.S. Facilities, except GFØ2 and GF15'.

While table space has been provided for these two items, and pertinent entries have already been made, the processing consequences need to be defined in greater detail and program implementations have to be made in Phase II. During the loading process the last entry will be combined with the control information in Column 24 (+ 50 for '*', + 60 for 'N').

The columns 18, 19, 20, 22, and 23 will be converted into integers when loaded to the system. (These columns show a double underlining of the table header, produced by the program LDISPLAY.) Although the ASCII input column provides more than two character positions per column, the final (converted) column will only be of 16-bit length (one computer word) when loaded to the system.

3.4.3.5 Page Header Line

The 82 Label Tables, created as outlined in the previous paragraphs, also contain a table header (zero line) that describes the sub-category. For example:

```
Columns   1   3                                  17
          /GF/FACILITY GENERAL INFORMATION
```

While columns 18 through 26 need no manual entries, they are used nevertheless for inputs with special meanings, computed and inserted by the load program.

3.4.3.6 Table Loading

The load program for Label Tables CRTLBL performs the discussed integer conversion, computed inputs for program and display controls, and table size adjustments with the creation of continuation tables where necessary. It produces a load file 'INLABEL' for the loading of each processed table to INFOS. This enables the on-line processing software later on to retrieve from INFOS individual tables by placing it into a common storage area: COMMON /LTABS/ LABLX(624), LEDX(330). By equivalencing LABLX to LABL(26,0:23), the 26x24 dimensioned table (including the zero or header line) can be accessed in the desired form using column and row indices.

3.4.3.7 Computed Header Line Columns

Columns 18 through 26 of the zero or header line are used by the loading program that creates INLABEL. The computed and inserted data have the following meaning:

Col. 18: TAB setting to center the display.

Col. 19: Unused.

Col. 20: Total number of elements (rows) in the original Label Table, including header or zero row.

Col. 21: Unused.

Col. 22: Tab for display column header 'CURRENT', centered over columnn.

Col. 23: Tab for display column header 'NEW', centered over column.

Col. 24: Actual number of elements in display. (Tables that are too long will be broken up by the load program into two display tables. Extra lines for graphic separations of multiples are considered also. The actual number of elements describes the number of elements in each display -- rather than the total number of elements of the subcategory).

Col. 25/26: Identification of original Label Table,
e.g., LAØ1 (first table).

3.4.3.8 Load File Sample

On the following pages a small part of the Load File INLABEL is presented. It indicates the current table design as loaded to the system.

```
1  3                                   18     20   22 23  24  26
/GF/FACILITY GENERAL INFORMATION        30 0   30**  76 9520LA01
GF01AIRFIELD NAME                       38-2   0     8410050
GF02ALTERNATE AIRFIELD NAME             38-2   0     8410051GF01
GF03AIRFIELD SYMBOL                      1 0   0     8410052
GF04AIRFIELD EXISTENCE SEC CLASS         3 0   0     8410050
GF05MAGNETIC VARIATION                   4 0   0     8410050
GF06TDI AREA CODE                        1 0   0     8410062
GF07YEAR OF LATEST INFORMATION          -2 0   0     8410050
GF08MONTH OF LATEST INFORMATION          2 0   0     8410050
GF09ICAO DESIGNATOR                      4 0   0     8410050
GF10MAX DEMON USAGE (AC TYPE)            4 0   0     8410050
GF11A/F ELEVATION                        5 0   0     8410050
GF12LOGISTICS PLAN & REPORT CODE         2 0   0     8410050
GF13A/F FOLDER NUMBER                    6 0   0     8410060
GF14OPERATOR USER REMARKS               38-1   61    8410050
GF15AIRFIELD STRIP                       1 0   0     8410050
GF16ASSOTW VOLUME NUMBER                 3 0   0     8410060
GF17ASSOTW PUBLICATION DATE             -4 0   0     8410061NFMT
GF18ASSOTW NUMBER OF PAGES              -1 0   0     8410061NFMT
GF19ASSOTW GRAPHIC SEC CLASS             3 0   0     8410061GF04
GF20ASSOTW RECORD TYPE                  -1 0   0     8410060
/GF/FACILITY GENERAL INFORMATION        30 0   30**  76 95 9LA01
GF21ASSOTW SEC CLASS                     3 0   0     8410061NFMT
GF22JET FACILITIES                      -1 0   0     8410051NFMT
GF23CC/PROV                              4 0   0     8410053GF03
GF24WAC                                 -4 0   0     8410053GF23
GF25INSTALLATION NUMBER                 -6 0   0     8410053GF23
GF26DATE OF LAST UPDATE CHANGES         -6 0   0     8410051NFMT
GF27CATEGORY CODE                        5 0   0     8410063GF03
GF28GEO CODE                             4 0   0     8410051GF09
GF29FACILITY GEN INFO SEC CLASS          3 0   0     8410051GF04
```

```
/GG/GEOGRAPHIC COORDINATES              16  0   11** 6210910LA02
GG01GEOGRAPHIC LATITUDE                  7  0    0   8411452
GG02GEOGRAPHIC LONGITUDE                 8  0    0   8411452
GG03GEOGRAPHIC COORDINATE TYPE           1  0    0   8411452
GG04GEOGRAPHIC LOCATION ACCURACY         1  0    0   8411453GG03
GG05GRID COORDINATES                    20  0    0   8411451NFMT
GG06GRID SYSTEM                         20  0    0   8411451NFMT
GG07GEOGRAPHIC IDENTIFIER                2  0    0   8411453GF23
GG08WORLD DIVISION CODE                  1  0    0   8411450
GG09BASE REFERENCE POINT TEXT           38-1 372     8411460
GG10GEOGRAPHIC COORDS SEC CLASS          3  0    0   8411451GF04
/GA/ATM SOURCE OF COORDINATES           30  0   12** 76 9511LA03
GA01ATM WAC                              4  0    0   8410052
GA02ATM INSTALLATION NUMBER             -6  0    0   8410050
GA03ATM SERIES                          -3  0    0   8410050
GA04ATM PRODUCER                         1  0    0   8410050
GA05ATM SHEET NUMBER                     4  0    0   8410050
GA06ATM SUFFIX                           1  0    0   8410050
GA07ATM EDITION NUMBER                   2  0    0   8410050
GA08ATM EDITION YEAR                    -2  0    0   8410051GF07
GA09ATM EDITION MONTH                   -2  0    0   8410051GF08
GA10ATM CHART CLASS/HANDLING             3  0    0   8410051GA10
GA11ATM CHART EXISTENCE CLASS            3  0    0   8410052
/GN/NON-ATM SOURCE OF COORDINATES       24  0    9** 70101 8LA04
GN01NON-ATM PRODUCER                     2  0    0   8410652
GN02NON-ATM SCALE                       -4  0    0   8410650
GN03NON-ATM SERIES                       5  0    0   8410650
GN04NON-ATM SHEET NBR/OTHER PROD        12  0    0   8410650
GN05NON-ATM EDITION NUMBER               2  0    0   8410650
GN06NON-ATM EDITION YEAR                -2  0    0   8410650
GN07NON-ATM EDITION MONTH               -2  0    0   8410650
GN08NON-ATM CHART EXIST CLASS            3  0    0   8410651GA10
/GR/REFERENCES, GRAPHIC                 33  0    3** 79 92 2LA05
GR01GRAPHIC REFERENCE TEXT              38-1 372     84 9760
GR02REFERENCES, GRAPHIC SEC CLASS        3  0    0   84 9761GF04
/GL/LOCATION & LANDMARKS                33  0    3** 79 92 2LA06
GL01LOCATION & LANDMARKS                38-1 372     84 9760
GL02LOCATION & LANDMARKS SEC CLASS       3  0    0   84 9761GF04
/GT/TERRAIN & DRAINAGE                  33  0    3** 79 92 2LA07
GT01TERRAIN & DRAINAGE TEXT             38-1 620     84 9760
GT02TERRAIN & DRAINAGE SEC CLASS         3  0    0   84 9761GF04
/GC/CONTROLLING AGENCY/AGENCIES         33  0    3** 79 92 2LA08
GC01CONTROL AGENCY(IES) TEXT            38-1 372     84 9760
GC02CONTROL AGENCY(IES) SEC CLASS        3  0    0   84 9761GF04
/GS/SIGNIFICANCE                        33  0    3** 79 92 2LA09
GS01SIGNIFICANCE TEXT                   38-11116     84 9760
GS02SIGNIFICANCE TEXT SEC CLASS          3  0    0   84 9761GF04
```

3.4.5.9 Edit Table Interfaces

As outlined previously, Column 24 of the Label Tables contains an edit program control number that determines -- for each data element -- whether Format and Logical checks have to be performed by the on-line processing programs, and which tables have to be used (Columns 1, 2 for 'Own' tables; Columns 25, 26 for 'Other' or referred tables). In a similar way the control numbers could also be engaged for the proper retrieval of on-screen generated edit tables from RDOS, and their subsequent systems loading to INFOS. By using the load file INLABEL, a 'wholesale' loading capability for all edit tables could thus be achieved. The available information would also permit checks for completeness, and a procedure to indicate missing edit tables.

This capability has been implemented with the systems program EPRT. Besides loading the edit tables to INFOS, an output file ETLIST is created that spells out for each data element (required to have a format or logical table) whether and under what table name the loading has occurred, or which table is missing. This provides a most convenient check list for ADA's system personnel. It greatly facilitates maintenance of the system, even when substantive changes have occurred that effect all types of tables: Label, Format, and Logical.

3.4.5.10 Check List for System Table Loading

A small part of the edit table check list ETLIST is presented on the following page. Note that for any odd control number a table with different name than the Data Element I.D. is listed. The control number rules, again, can be stated as follows:

0 - Use 'Own' Format table
1 - Use 'Different' Format table*
2 - Use 'Own' Format and Logical tables
3 - Use 'Own' Format, 'Different' Logical table*
5 - Use 'Different' Format and Logical tables*

* - 'Different' table for 'odd' control numbers (entries in columns 25, 26 required).

Format Tables	L		Logical Tables	L	Data Element Requiring Edit Table I.D.	NO.
GF03	2		GF03	2	GF03	1
GF04	0				GF04	2
GF05	0				GF05	3
GF06	2		GF06	2	GF06	4
GF07	0				GF07	5
GF08	0				GF08	6
GF09	0				GF09	7
GF10	0				GF10	8
GF11	0				GF11	9
GF12	0				GF12	10
GF13	0				GF13	11
GF15	0				GF15	12
GF16	0				GF16	13
NFMT	1				GF17	14
NFMT	1				GF18	15
GF04	1				GF19	16
GF20	0				GF20	17
NFMT	1				GF21	18
NFMT	1				GF22	19
GF23	3		GF03	3	GF23	20
GF24	3		GF23	3	GF24	21
GF25	3		GF23	3	GF25	22
NFMT	1				GF26	23
GF27	3		GF03	3	GF27	24
GF09	1				GF28	25
GF04	1				GF29	26
GG01	2		GG01	2	GG01	27
GG02	2		GG02	2	GG02	28
GG03	2		GG03	2	GG03	29
GG04	3		GG03	3	GG04	30
NFMT	1				GG05	31
NFMT	1				GG06	32
GG07	3	MISSING	GF23	3	GG07	33
GG08	0				GG08	34
GF04	1				GG10	35
GA01	2		GA01	2	GA01	36
GA02	0				GA02	37
GA03	0				GA03	38
GA04	0				GA04	39
GA05	0				GA05	40
GA06	0				GA06	41
GA07	0				GA07	42
GF07	1				GA08	43
GF08	1				GA09	44
GA10	1				GA10	45

3.4.4 Edit Tables

The entered (and frequently updated) data base information of the Air Facilities Subsystem will be used by ADA and various other Government agencies for decisionmaking or further data processing. It is essential, therefore, to insure format correct entries that avoid costly, time consuming (and embarrassing) malfunctions at the respective computer sites. Format and logical checks of new or updated data are an important prerequisite for making all following programs and software packages work.

3.4.4.1 Implementation Alternatives

There are several ways for implementing format and logical checks in order to test input data for correctness. Most frequently, this task is accomplished by Special Programs that test and check the user-specified conditions for various data base elements. The software solution to the problem, however, has serious drawbacks. Special programming efforts are time consuming and require ever increasing resources of a system that is bound to grow in both size and complexity. An 'old' implementation that works tends to prevent later improvements from being incorporated. Changes are costly in terms of programming efforts and system outages. Unless everything is absolutely right from the start (and stays that way for years to come) the user may find himself 'boxed in,' or he tends to lose control over his system in spite of all that good 'modularity' and 'flexibility' of the employed software.

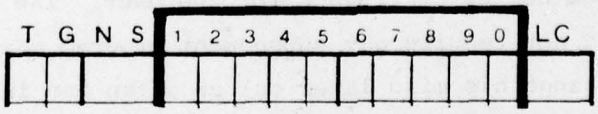
By using a Boolean Interpreter -- an interpretative program that would take a data string of meaningful logical statements and act upon accordingly -- one could retain a large degree of freedom for the user. The interpreter, once implemented, would not require any subsequent program changes. The user can specify (or change his mind later on) as often and in whatever detail he wishes by just forming another data string. However, it must be realized that the formulation of interdependencies in correct Boolean terms is a cumbersome process, mastered and properly understood only by a few who,

in turn, tend to become the human 'bottlenecks' of the system. A comprehensive, free-form Boolean Interpreter, furthermore, is not readily available. Its development would have required considerable effort, plus substantial system resources for software implementation.

To circumvent the outlined difficulties, Synectics has devised a Table Driven Solution to the problem. It is an interpretive program for the evaluation of logical tables. Like the Boolean Interpreter, the computer software need not be changed or extended ever, yet it is capable of handling edit tables of any complexity or size to accommodate today's and tomorrow's needs. The table structure takes advantage of positional information (as well as the data itself); there is no need for parsing algorithms, or for the use and the processing of brackets. The structured layout facilitates a well-defined, orderly approach that can readily be learned and mastered by noncomputer personnel. In fact, it has been shown that, after a short familiarization period, one can input and on-line test the implemented tables almost as fast as one could write them down in regular English phrases. The ease of controlling all important logical functions in this manner is the best guarantee for a responsive system that is truly mastered and 'owned' by the user.

3.4.4.2 Format Table Rules

The rules for setting up format tables are simple. There are three types of "Data Fields" only, linked together by simplified "Logical Connectors" (LC). The latter represent AND's and OR's. Brackets are implied only, there are no critical rules for matching them up correctly.



Data Field

The data specifiers T, G, N, S are used as follows:

T--Type of Data Field:

No Entry--Regular OR's
R --ASCII Range
N --Numeric Range
- The minus sign in Column T causes all elements in the Data Field to be interpreted as logical NOT's

G--Grouping of Characters:

Specifies the number of characters per group, taken at a time for evaluation

N--N Times:

Indicates the number of times a group of G characters of the input data is compared with the entries of the Data Field

S--Start Position:

Character start position of input data

The logical connectors, as the name implies, connect one data field to the next by simple AND or OR symbols. By using a series of different symbols, low, medium and high level OR's and AND's may be specified. This allows the construction of complex logical expressions without any need for complicated rules concerning the opening and closing of brackets.

O - OR	Low Level		
A - AND			
* - OR			
& - AND			
X - OR		--	LAST
E - AND	High Level	NE	LAST (Negated)

The last line of the table uses a 'connector' of '--' to specify the end of the table, or a 'NE.' The latter indicates a 'NEgated Output;' i.e. switching from 'FAIL' to 'PASS' (and vice versa). Although this feature is not frequently used in practice, it comes in handy in those applications where it is easier to specify in a positive manner (rather than in negative forms) what criteria an input should not have. By negating the output the correct interpretation is then achieved.

3.4.4.3 Samples of Format Tables

The mentioned rules are almost easier to be used than explained as the following samples indicate. The first three format checks show both stated requirements and table implementations, for the three data base elements GFØ3, GFØ4, and GFØ5.

The next page illustrates the establishment of specific character categories, based on the computer-internal representation of the ASCII characters.

The third page outlines the format requirements for ARØ7, "Runway Aircraft Capacity," and a possible table implementation of that requirement. The interesting feature of the presented solution is that the allowable codes are not just listed "as is' (with just interconnecting OR's), but broken up in AND's and OR's (low and medium) to achieve a shorter table and a faster processing sequence.

GFØ3
Airfield Symbol

1 Char. Alpha. Limited to A, B, C, D, E, G, H, X.

T	G	N	S	1	2	3	4	5	6	7	8	9	0	LC
1		1		A	B	C	D	E	G	H	X			--

GFØ4
Airfield Existence
Security Class

3 Char. Alpha. First Char = U, S or C
Second/Third Char = NF, EA, EN, EH, EB, EG, EE, EC, EJ, FP, WI or Blanks.

T	G	N	S	1	2	3	4	5	6	7	8	9	0	LC	
1		1		U	S	C								A	AND (
2		2				N	F	E	A	E	N	E	H	O	OR
2				E	B	E	G	E	E	E	C	E	J	O	OR
2				F	P	W	I							--	Exit

Boolean Picture for GFØ4: [1] A ([23] O [23] O [23]

GFØ5
Magnetic
Variation

4 Char. Alpha-Numeric. First 3 Char. must be numeric in the range of ØØØ-179. Char 4 must be E or W. If Char 1-3 = ØØØ Char 4 must be E.

T	G	N	S	1	2	3	4	5	6	7	8	9	0	LC		
N	3	1		Ø	Ø	1			1	7	9			A	Range	AND
1		4		E	W									*	Exit if True	] OR [
3		1		Ø	Ø	Ø								A		AND
1		4		E										--	Exit	

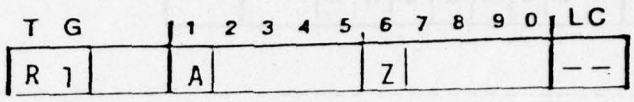
Boolean Picture for GFØ5: [123] A [4]] O [[123] A [4]

ASCII CHARACTER CODES

Decimal	7-bit Octal	Character
32	040	SP
33	041	!
34	042	"
35	043	/
36	044	$
37	045	%
38	046	&
39	047	'
40	050	(
41	051	)
42	052	*
43	053	+
44	054	,
45	055	-
46	056	.
47	057	/
48	060	0
49	061	1
50	062	2
51	063	3
52	064	4
53	065	5
54	066	6
55	067	7
56	070	8
57	071	9
58	072	:
59	073	;
60	074	<
61	075	=
62	076	>
63	077	?
64	100	@
65	101	A
66	102	B
67	103	C
68	104	D
69	105	E
70	106	F
71	107	G
72	110	H
73	111	I
74	112	J
75	113	K
76	114	L
77	115	M
78	116	N
79	117	O
80	120	P
81	121	Q
82	122	R
83	123	S
84	124	T
85	125	U
86	126	V
87	127	W
88	130	X
89	131	Y
90	132	Z

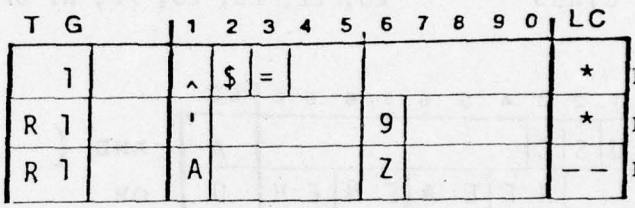
SPECIAL FORMAT CHECKS

ALPHA CHARACTERS ONLY



Range A - Z

DATA BASE CHARACTER SET

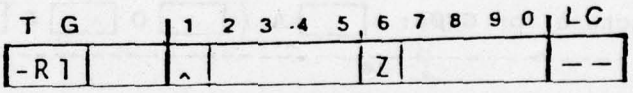


blank, $, = OR

Range ' - 9 OR

Range A - Z

NO FORMAT CHECK OF FIELD



NOT in Range A - Z

▲

Every manually inserted character is 'wrong' as input. (Effective prevention of manual inputs)

AR07	Runway Aircraft Capacity

4	Alpha-Numeric. Allowable codes are:

Aircraft

				Helicopters
A1	C124	F101	U6	AH1
A4	C130	F102	U7	AH56
A5	C131	F104	U8	CH21
A6	C135	F105	U10	CH34
A7	K135	F106	U17	CH37
A37	V137	F111	UV18	CH47
B26	C140	HU16	U21	CH54
B52	C141	O1		HH43
B57	707	O2		OH6
B66	727	OV1		OH13
C2	747	OV10		OH23
C5	DC8	P2		OH58
C7	DC10	P3		UH1
C9	1011	T28		UH19
C45	E2	T33		
C46	E3	T37		
C47	E4	T38		
C54	F4	T39		
C97	F5	T41		
C117	F14	T42		
C118	F15	T43		
C119	F84	U1		
C121	F86	U3		
C123	F100	U4		

Requirements

Example:

The first three lines of the table specify the allowable codes:

A1
A4
A5
A6
A7
A37

A		And
1 or 4 or 5		Or
6 or 7 or 37	*	

1	2 3	4 5 6 7 8	9
-TG	N S	1234567890	LC
---	----	----------	--
1	1	A	A
3	2	1 4 5	O
3		6 7 37	*
1	1	B	A
3	2	26 52 57	O
3		66	*
1	1	C	A
3	2	2 5 7	O
3		9 45 46	O
3		47 54 97	O
3		117118119	O
3		121123124	O
3		131135140	O
3		141	*
4	1	V137K135	*
4		707 727	*
4		747 DC8	*
4		DC101011	*
4		E2 E3	*
4		E4	*
1	1	F	A
3	2	4 5 14	O
3		15 84 86	O
3		100101104	O
3		105106111	*
4	1	HU16	*
1	1	O	A
3	2	1 2 V1	O
3		V10H6 H13	O
3		H23H58	*
1	1	T	A
3	2	28 33 37	O
3		38 39 41	O
3		42 43	*
1	1	U	A
3	2	1 3 4	O
3		6 7 8	O
3		10 17 H1	O
3		H19V1821	*
4	1	AH1 AH56	*
1	1	C	A
3	2	H21H34H37	O
3		H47H54	*
4	1	HH43	*
1	1	O	A
3	2	H6 H13H23	O
3		H58	-

3.4.4.4 On-Line Table Verifications

Synectics table-driven solution to the editing problem does not require program developments or program alterations to accommodate changing requirements. It permits the ready system loading, testing, and documentation of changed or new edit tables in an interactive environment with immediate computer feedback for testing, verifying, and fully concentrating on the desired editing function.

This capability has been achieved by implementing an on-line test and evaluation package that serves as a comprehensive tool for edit table design and their automatic loading to the system. This package allows the functional verification of all edit tables as they are entered into the system. It displays the actual processing and response to various classes of inputs by showing in detail whether, where, and why a table lets correct inputs pass while incorrect ones are rejected.

The built-in features serve two functions: (1) that of proper table design with an implicit training of those analysts who are responsible for establishing and maintaining these system controls and (2) that of verifying and documenting the editing functions.

The latter can be accomplished by the on-line session itself through simultaneous hardcopy output. The generated output thus serves as a basis for documentation and system verification.

The following samples should provide a good insight into the creation and on-line testing of such format tables. The original table input can be achieved on-line via the special program "FHEAD." This program produces header lines and automatically sets appropriate tabs for simplified input operations. (For logical tables the program "LHEAD" is used.) In addition, card input programs are available if off-line preparations through key punch personnel are preferred.

During the on-line test session the table input is subjected to a formal check to test the presence of key elements. The input is then converted and made ready for system loading. This is accomplished by eliminating the header, the automatic inclusion of default values, and by changing the LC code to (computer-internally used) numbers. The analyst can select from a number of options:

T--Test of up to 5 inputs simultaneously
L--Large test (one test input has up to 29 characters)
S--Select or Switch from FORMAT to LOGICAL tables
C--Change Table (for on-line changes, line insertion or deletions)
E--Exit the Test program or
NAME of next table to be tested--

3.4.4.5 Sample of On-Line Test Session

In the following example 'AR17' the option for 5 test inputs has been chosen. Detailed Data Field evaluations for each of these inputs are displayed plus the final outcome 'FAIL' or 'PASS.'

AR17	Runway Overrun Surface (Low End)	Alpha. Same codes as ARØ4 with the addition of A, E, N.
ARØ4	Runway Surface Composition	Alpha. Allowable codes are as follows: ASP, BIT, BRI, CLA, COM, CON, COP, COR, GRE, GRS, GVL, ICE, LAT, MAC, MEM, MIX, PEM, PER, PSP, SAN, SNO or U.

Original Table Input

1	2 3	4 5 6 7 8	9
-TG	N S	1234567890	LC
---	----	----------	--
3		ASPBITBRI	*
3		CLACOMCON	*
3		COPCORGRE	*
3		GRSGVLICE	*
3		LATMACMEM	*
3		MIXPEMPER	*
3		PSPSANSNO	*
3		U A E	*
3		N	-

(Entered as input file)

- 1-	3	1	1	ASPBITBRI 3
- 2-	3	1	1	CLACOMCON 3
- 3-	3	1	1	COPCORGRE 3
- 4-	3	1	1	GRSGVLICE 3
- 5-	3	1	1	LATMACMEM 3
- 6-	3	1	1	MIXPEMPER 3
- 7-	3	1	1	PSPSANSNO 3
- 8-	3	1	1	U A E 3
- 9-	3	1	1	N 8

Converted Table

(Loaded to the system)

NAME OF TABLE -- AR17

-TG	N S	DATA FIELD	LC	(E)	U	A	PSP	SSS
---	----	----------	--					
3		ASP BIT BRI	*	F	F	F	F	F
3		CLA COM CON	*	F	F	F	F	F
3		COP COR GRE	*	F	F	F	F	F
3		GRS GVL ICE	*	F	F	F	F	F
3		LAT MAC MEM	*	F	F	F	F	F
3		MIX PEM PER	*	F	F	F	F	F
3		PSP SAN SNO	*	F	F	F	T	F
3		U A (E)	*	(F)	T	T	.	F
3		N	-	F	.	.	.	F
				FAIL	PASS	PASS	PASS	FAIL

The above test gives an example for detecting a simple error in the table design. Since the (acceptable) input 'E' leads to a 'FAIL' situation, there must be something wrong with the edit table. A closer inspection reveals that the 'E' has been placed in the middle of the 3-character group instead of the beginning.

These displays are also listed in the session file for later printouts on demand. The printouts can be used as detailed working papers for (off-line) inspections, improvements, table changes, and the preparation of next on-line sessions at the system analyst's desk. Such use significantly alleviates a crowded terminal schedule, and improves both speed and quality of the 'next session' by allowing for a better preparation than would otherwise be possible. All tables tested during an on-line session will automatically be loaded to the system. The analyst has no choice in this matter in order to ensure that 'what he sees, is what the system gets.' A printout of the last test session (for a particular table) should consequently be kept, serving as the final load and test evaluation document. Only 'play' sessions are exempt from this rule.

3.4.4.6 Format Checks for Geographic Latitude and Country Codes

The following test printouts for GG01, "Geographic Latitude" and the suggested three solutions to the "Country Code" edit check problem provide good insights for the use, inner workings, and the potential improvements of such format tables.

ELT	ELEMENT NAME	REQUIRED EDITS
GGØ1	Geographic Latitude	Alpha-Numeric. First and Second characters must be numeric and less than 91, Third and Fourth must be numeric and less than 6Ø, Fifth and Sixth characters must be numeric and less than 6Ø. Last character must be 'N' or 'S'.

This Format Check Table, for Geographic Latitude entries, is an example for the use of Logical NOT's (-) and Numeric ranges (N). The first two rows of the table stipulate that the first 6 characters, taken one at a time (G=1), shall not be 'blank' or '.' . The following three lines give the numeric ranges for the two-digit groups (N2), starting at input positions 1, 3, and 5.

-TG	N	S	1234567890		LC
-	1	6	1		A
-	1	6	1	.	A
N2	1	1	00	90	A
N2	1	3	00	59	A
N2		5	00	59	A
1		7	NS		-

Since all Table rows are interconnected by logical AND's, the input must be true for each table row to PASS.

Test Input

-TG	N	S	DATA FIELD			LC	123456N
-	1	6	1			A	T
-	1	6	1	.		A	T
N2	1	1	00	==	90	A	T
N2	1	3	00	==	59	A	T
N2		5	00	==	59	A	T
1		7	N S			-	T
							PASS

-TG	N	S	DATA FIELD			LC	912345N
-	1	6	1			A	T
-	1	6	1	.		A	T
N2	1	1	00	==	(90)	A	(F)
N2	1	3	00	==	59	A	.
N2		5	00	==	59	A	.
1		7	N S			-	.
							FAIL

Numeric Range: From == To

continued

NAME OF TABLE -- GG01

TEST: -- 1262345

-TG	N	S	DATA FIELD			LC	1262345
- 1	6	1				A	T
- 1	6	1	.			A	T
N2	1	1	00	==	90	A	T
N2	1	3	00	==	(59)	A	(F)
N2		5	00	==	59	A	.
1		7	N S			-	.
							FAIL

TEST: -- 123456Y

	-TG	N	S	DATA FIELD			LC	123456(Y)
	- 1	6	1				A	T
	- 1	6	1	.			A	T
	N2	1	1	00	==	90	A	T
	N2	1	3	00	==	59	A	T
▶	N2		5	00	==	59	A	T
	1		7	N S ◯			-	(F)
								FAIL

-TG	N	S	DATA FIELD			LC
- 1	6	1				A
- 1	6	1	.			A
N2	1	1	00	==	90	A
N2	(2)	3	00	==	59	A
1	1	7	N S			-

The indicated row of the table could have been combined with the preceeding one, thus shortening the table to the form presented at left. By stipulating that the numeric range (T=N) of a 2-character group (G=2) be tested <u>two times</u> (N=2), starting at the third position of the input string (S=3), the same format checks would be accomplished.

THREE SOLUTIONS

TO THE HANDLING OF EDIT CHECKS FOR 'COUNTRY CODES'

(1) Long, unsophisticated table
(2) Compact Format table
(3) Fast working table arrangement

(1)

1 -TG	2 3 N S	4 5 6 7 8 12345678900	9 LC
A	1 1	AC AF	•
A	1 1	AG AH	•
A	1 1	AI AJ	•
A	1 1	AK AL	•
A	1 1	AM AN	•
A	1 1	AO AP	•
A	1 1	AQ AR	•
A	1 1	AS AT	•
A	1 1	AU AV	•
A	1 1	AW AX	•
A	1 1	AY AZ	•
A	1 1	BA BC	•
A	1 1	BD BE	•
A	1 1	BF BG	•
A	1 1	BH BL	•
A	1 1	BM BP	•
A	1 1	BQ BR	•
A	1 1	BS BV	•
A	1 1	BW BX	•
A	1 1	BY CA	•
A	1 1	CB CD	•
A	1 1	CE CF	•
A	1 1	CG CI	•
A	1 1	CJ CK	•
A	1 1	CL CM	•
A	1 1	CN CO	•
A	1 1	CP CS	•
A	1 1	CT CU	•
A	1 1	CV CX	•
A	1 1	CY CZ	•
A	1 1	DA DD	•
A	1 1	DE DK	•
A	1 1	DL DM	•
A	1 1	DN DO	•
A	1 1	DP DS	•
A	1 1	DT DV	•
A	1 1	DW DX	•
A	1 1	DY DZ	•
A	1 1	EA EB	•
A	1 1	EC ED	•
A	1 1	EE EF	•
A	1 1	EG EH	•
A	1 1	EI EJ	•
A	1 1	EK EL	•
A	1 1	EM EN	•
A	1 1	EO EP	•
A	1 1	EQ ER	•
A	1 1	ES ET	•
A	1 1	EU EV	•
A	1 1	EW EX	•
A	1 1	EY EZ	•
A	1 1	FA FB	•
A	1 1	FC FD	•
A	1 1	FE FF	•
A	1 1	FG FH	•
A	1 1	FI FJ	•
A	1 1	FK FL	•
A	1 1	FM FN	•
A	1 1	FO FP	•
A	1 1	FQ FR	•
A	1 1	FS FT	•
A	1 1	FU FV	•
A	1 1	FW FX	•
A	1 1	FY FZ	•
A	1 1	GA GB	•
A	1 1	GC GD	•
A	1 1	GE GF	•
A	1 1	GG GH	•
A	1 1	GI GJ	•
A	1 1	GK GL	•
A	1 1	GM GN	•
A	1 1	GO GP	•
A	1 1	GQ GR	•
A	1 1	GS GT	•
A	1 1	GU GV	•
A	1 1	GW GX	•
A	1 1	GY GZ	•
A	1 1	HA HB	•
A	1 1	HC HD	•
A	1 1	HE HF	•
A	1 1	HG HH	•
A	1 1	HI HJ	•
A	1 1	HK HL	•
A	1 1	HM HN	•
A	1 1	HO HP	•
A	1 1	HQ HR	•
A	1 1	HS HT	•
A	1 1	HU HV	•
A	1 1	HW HX	•
A	1 1	HY HZ	•
A	1 1	IA IB	•
A	1 1	IC ID	•
A	1 1	IE IF	•
A	1 1	IG IH	•
A	1 1	II IJ	•
A	1 1	IK IL	•
A	1 1	IM IN	•
A	1 1	IO IP	•
A	1 1	IQ IR	•
A	1 1	IS IT	•
A	1 1	IU IV	•
A	1 1	IW IX	•
A	1 1	IY IZ	•
A	1 1	JA JB	•
A	1 1	JC JD	•
A	1 1	JE JF	•
A	1 1	JG JH	•
A	1 1	JI JJ	•
A	1 1	JK JL	•
A	1 1	JM JN	•
A	1 1	JO JP	•
A	1 1	JQ JR	•
A	1 1	JS JT	•
A	1 1	JU JV	•
A	1 1	JW JX	•
A	1 1	JY JZ	•
A	1 1	KA KB	•
A	1 1	KC KD	•
A	1 1	KE KF	•
A	1 1	KG KH	•
A	1 1	KI KJ	•
A	1 1	KK KL	•
A	1 1	KM KN	•
A	1 1	KO KP	•
A	1 1	KQ KR	•
A	1 1	KS KT	•
A	1 1	KU KV	•
A	1 1	KW KX	•
A	1 1	KY KZ	•
A	1 1	LA LB	•
A	1 1	LC LD	•
A	1 1	LE LF	•
A	1 1	LG LH	•
A	1 1	LI LJ	•
A	1 1	LK LL	•
A	1 1	LM LN	•
A	1 1	LO LP	•
A	1 1	LQ LR	•
A	1 1	LS LT	•
A	1 1	LU LV	•
A	1 1	LW LX	•
A	1 1	LY LZ	•
A	1 1	MA MB	•
A	1 1	MC MD	•
A	1 1	ME MF	•
A	1 1	MG MH	•
A	1 1	MI MJ	•
A	1 1	MK ML	•
A	1 1	MM MN	•
A	1 1	MO MP	•
A	1 1	MQ MR	•
A	1 1	MS MT	•
A	1 1	MU MV	•
A	1 1	MW MX	•
A	1 1	MY MZ	•
A	1 1	NA NB	•
A	1 1	NC ND	•
A	1 1	NE NF	•
A	1 1	NG NH	•
A	1 1	NI NJ	•
A	1 1	NK NL	•
A	1 1	NM NN	•
A	1 1	NO NP	•
A	1 1	NQ NR	•
A	1 1	NS NU	•
A	1 1	NV NZ	•
A	1 1	OA OB	•
A	1 1	OC OE	•
A	1 1	OF OG	•
A	1 1	OH OI	•
A	1 1	OK OL	•
A	1 1	OM ON	•
A	1 1	OO OP	•
A	1 1	OQ OR	•
A	1 1	OS OT	•
A	1 1	OU OV	•
A	1 1	OW OX	•
A	1 1	OY OZ	•
A	1 1	PA PB	•
A	1 1	PC PE	•
A	1 1	PF PK	•
A	1 1	PM PO	•
A	1 1	PP PQ	•
A	1 1	PU QA	•
A	1 1	QE RH	•
A	1 1	RP RQ	•
A	1 1	RR SA	•
A	1 1	SB SC	•
A	1 1	SE SF	•
A	1 1	SG SH	•
A	1 1	SL SM	•
A	1 1	SN SO	•
A	1 1	SP ST	•
A	1 1	SU SW	•
A	1 1	SY SZ	•
A	1 1	TE TO	•
A	1 1	TR ZA	•
A	1 1	US01US02	•
A	1 1	US03US04	•
A	1 1	US05US06	•
A	1 1	US07US08	•
A	1 1	US09US10	•
A	1 1	US11US12	•
A	1 1	US13US14	•
A	1 1	US15US16	•
A	1 1	US17US18	•
A	1 1	US19US20	•
A	1 1	US27US28	•
A	1 1	US29US30	•
A	1 1	US31US32	•
A	1 1	US33US34	•
A	1 1	US35US36	•
A	1 1	US37US38	•
A	1 1	US39US40	•
A	1 1	US41US42	•
A	1 1	US43US44	•
A	1 1	US45US46	•
A	1 1	US47US48	•
A	1 1	US49US50	•
A	1 1	US51US52	•
A	1 1	US53US54	•
A	1 1	US55US56	-

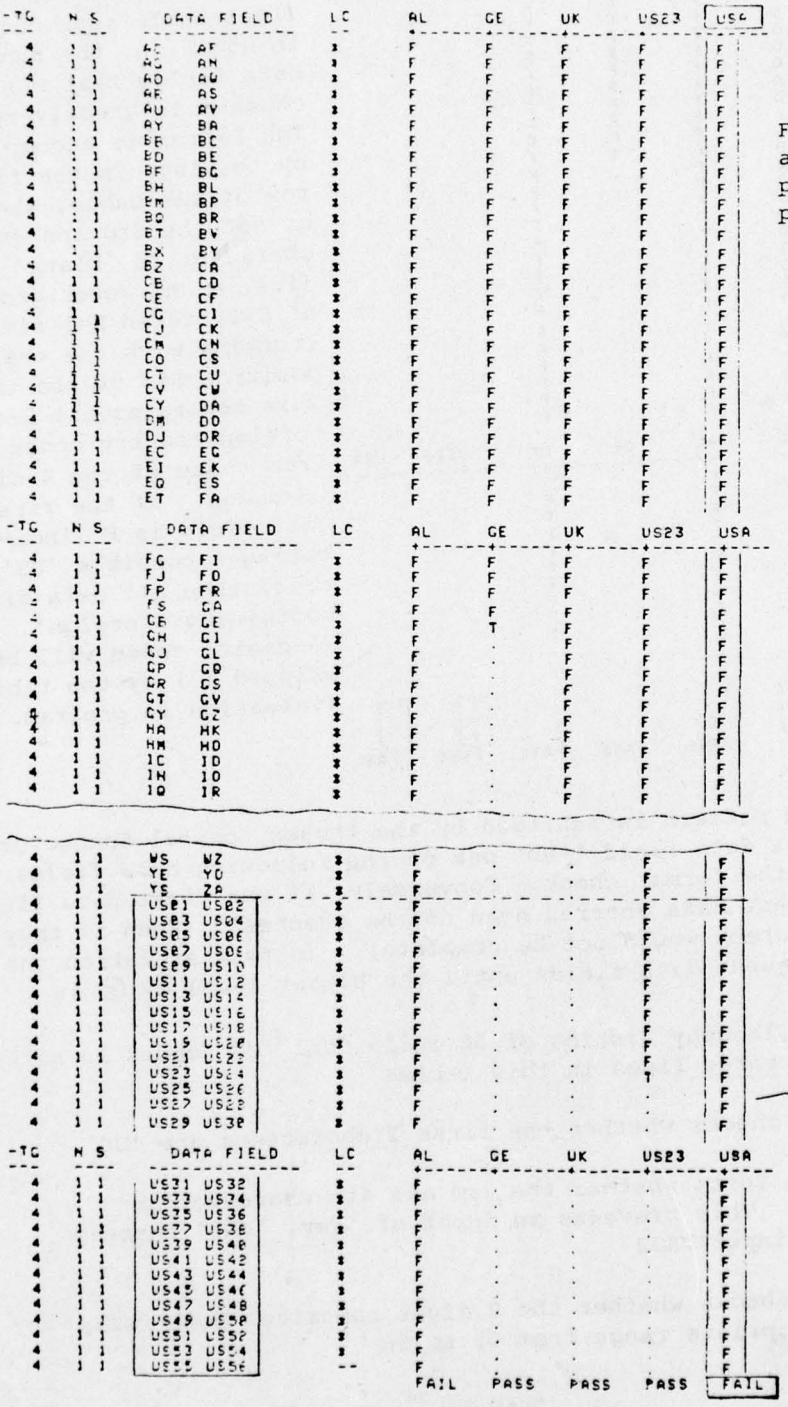
(2)

1 -TG	2 3 N S	4 5 6 7 8 12345678900	9 LC
2	1 3		A
2	1	ACAFAGANAD	0
2		AQARASAUAV	0
2		AYBABHBNCBD	0
2		BFBFBGBHBI	0
2		BKBPBQBRBT	0
2		BVBXBYBZCA	0
2		CBCDCFCFCG	0
2		CICJCKCMCN	0
2		COCSCTCUCV	0
2		CWCYDADMDO	0
2		DJDPDEDFDI	0
2		EKENESETEA	0
2		EGEIEJEOEP	0
2		ERESGAGBGE	0
2		GHGIGJGLGP	0
2		GQGRGSGTGV	0
2		GYGZHAHKHM	0
2		HOICIDINIU	0
2		IQIRISITIV	0
2		IYIZJAJMJO	0
2		JSKFKSKTKD	0
2		LALFLILSLT	0
2		LULYMAMBMC	0
2		MHMIMLMNMO	0
2		MPMQMRMTMU	0
2		MVMXMYMZNA	0
2		NCNENFNGNH	0
2		NINLNONPNR	0
2		NSNUNZPAPC	0
2		PEPGPKPNPO	0
2		PRPQPUPARE	0
2		RHRPRQRWSA	0
2		SBSCSESFSG	0
2		SHSLSMSNSO	0
2		SPSTSUSWSY	0
2		SZTCTDTHTK	0
2		TLTNTOTPTQ	0
2		TSTUTVTWTZ	0
2		UGUKUOUPUC	0
2		VEVIVMVQVT	0
2		WAWEWIWJWS	0
2		WZYEYOYSZA	*
2		US	A
R1	2 3	0 9	A
R2	1	01 56	-

(3)

1 -TG	2 3 N S	4 5 6 7 8 12345678900	9 LC
2	3		&
1	1	A	A
1	2	CEGNOQRSUV	0
1	2	Y	•
1	1	H	A
1	2	ABCDEFGHLM	0
1	2	PQRTVXYZ	•
1	1	C	A
1	2	ABDEFGIJKN	0
1	2	NOSTUVWY	•
1	1	D	A
1	2	AMOJW	•
1	1	E	A
1	2	CGIKQST	•
1	1	F	A
1	2	AGIJOPRS	•
1	1	G	A
1	2	ABEHIJLPQR	0
1	2	STVYZ	•
1	1	H	A
1	2	AKMU	•
1	1	I	A
1	2	CDNOQRSTVY	0
1	2	Z	•
1	1	J	A
1	2	AMOS	•
1	1	K	A
1	2	ESTU	•
1	1	L	A
1	2	AEISTUY	•
1	1	M	A
1	2	ABCHILNOPQ	0
1	2	RTUVXYZ	•
1	1	N	A
1	2	ACEFGHILOP	0
1	2	RSUZ	•
1	1	P	A
1	2	ACEGKNOPQU	•
2	1	QA	•
1	1	R	A
1	2	EHPUW	•
1	1	S	A
1	2	ABCEFGHLMA	0
1	2	OPTUWYZ	•
1	1	T	A
1	2	CDHKLNOPQS	0
1	2	UVWZ	•
1	1	U	A
1	2	GKVY	•
1	1	V	A
1	2	CEIMQT	•
1	1	W	A
1	2	AFIOSZ	•
1	1	Y	A
1	2	EOS	•
2	1	ZA	X
2	1	US	A
R1	2 3	0 9	A
R2	1 3	01 56	--

(For greater detail see the following pages)



-TG	N	S	DATA FIELD	LC	AL	GE	UK	US23	USA
4	1	1	AC AF	1	F	F	F	F	F
4	1	1	AG AN	1	F	F	F	F	F
4	1	1	AO AQ	1	F	F	F	F	F
4	1	1	AR AS	1	F	F	F	F	F
4	1	1	AU AV	1	F	F	F	F	F
4	1	1	AY BA	1	F	F	F	F	F
4	1	1	BB BC	1	F	F	F	F	F
4	1	1	BD BE	1	F	F	F	F	F
4	1	1	BF BG	1	F	F	F	F	F
4	1	1	BH BL	1	F	F	F	F	F
4	1	1	BM BP	1	F	F	F	F	F
4	1	1	BQ BR	1	F	F	F	F	F
4	1	1	BT BV	1	F	F	F	F	F
4	1	1	BX BY	1	F	F	F	F	F
4	1	1	BZ CA	1	F	F	F	F	F
4	1	1	CB CD	1	F	F	F	F	F
4	1	1	CE CF	1	F	F	F	F	F
4	1	1	CG CI	1	F	F	F	F	F
4	1	1	CJ CK	1	F	F	F	F	F
4	1	1	CM CN	1	F	F	F	F	F
4	1	1	CO CS	1	F	F	F	F	F
4	1	1	CT CU	1	F	F	F	F	F
4	1	1	CV CW	1	F	F	F	F	F
4	1	1	CY DA	1	F	F	F	F	F
4	1	1	DM DO	1	F	F	F	F	F
4	1	1	DJ DR	1	F	F	F	F	F
4	1	1	EC EG	1	F	F	F	F	F
4	1	1	EI EK	1	F	F	F	F	F
4	1	1	EQ ES	1	F	F	F	F	F
4	1	1	ET FA	1	F	F	F	F	F
-TG	N	S	DATA FIELD	LC	AL	GE	UK	US23	USA
4	1	1	FC FI	1	F	F	F	F	F
4	1	1	FJ FO	1	F	F	F	F	F
4	1	1	FP FR	1	F	F	F	F	F
4	1	1	FS GA	1	F	F	F	F	F
4	1	1	GB GE	1	F	T	F	F	F
4	1	1	GH GI	1	F	.	F	F	F
4	1	1	GJ GL	1	F	.	F	F	F
4	1	1	GP GQ	1	F	.	F	F	F
4	1	1	GR GS	1	F	.	F	F	F
4	1	1	GT GV	1	F	.	F	F	F
4	1	1	GY GZ	1	F	.	F	F	F
4	1	1	HA HK	1	F	.	F	F	F
4	1	1	HM HO	1	F	.	F	F	F
4	1	1	IC ID	1	F	.	F	F	F
4	1	1	IN IQ	1	F	.	F	F	F
4	1	1	IQ IR	1	F	.	F	F	F
4	1	1	WS WZ	1	F	.	.	F	F
4	1	1	YE YO	1	F	.	.	F	F
4	1	1	YS ZA	1	F	.	.	F	F
4	1	1	US01 US02	1	F	.	.	F	F
4	1	1	US03 US04	1	F	.	.	F	F
4	1	1	US05 US06	1	F	.	.	F	F
4	1	1	US07 US08	1	F	.	.	F	F
4	1	1	US09 US10	1	F	.	.	F	F
4	1	1	US11 US12	1	F	.	.	F	F
4	1	1	US13 US14	1	F	.	.	F	F
4	1	1	US15 US16	1	F	.	.	F	F
4	1	1	US17 US18	1	F	.	.	F	F
4	1	1	US19 US20	1	F	.	.	F	F
4	1	1	US21 US22	1	F	.	.	F	F
4	1	1	US23 US24	1	F	.	.	T	F
4	1	1	US25 US26	1	F	.	.	.	F
4	1	1	US27 US28	1	F	.	.	.	F
4	1	1	US29 US30	1	F	.	.	.	F
-TG	N	S	DATA FIELD	LC	AL	GE	UK	US23	USA
4	1	1	US31 US32	1	F	.	.	.	F
4	1	1	US33 US34	1	F	.	.	.	F
4	1	1	US35 US36	1	F	.	.	.	F
4	1	1	US37 US38	1	F	.	.	.	F
4	1	1	US39 US40	1	F	.	.	.	F
4	1	1	US41 US42	1	F	.	.	.	F
4	1	1	US43 US44	1	F	.	.	.	F
4	1	1	US45 US46	1	F	.	.	.	F
4	1	1	US47 US48	1	F	.	.	.	F
4	1	1	US49 US50	1	F	.	.	.	F
4	1	1	US51 US52	1	F	.	.	.	F
4	1	1	US53 US54	1	F	.	.	.	F
4	1	1	US55 US56	--	F	.	.	.	F
					FAIL	PASS	PASS	PASS	FAIL

(1) UNSOPHISTICATED TABLE
'COUNTRY CODES'

For space considerations only a part of the table (with test pattern for five inputs) is presented here.

Data Field Evaluations:

T - True (match with input)
F - False(no match)
. - Skip Data Field Evaluation

Relatively few data fields can be skipped with this table arrangement, resulting in longer processing times.

Note in particular the considerable amount of comparisons required for all 'US' codes.

-TG	N	S	DATA FIELD	LC	AL	GE	UK	USØ3	USA
2	1	3		Ⓐ	T	T	T	F	F
2		1	AC AF AG AH AD	0	F	F	F	.	.
2			AØ AR AS AU AV	0	F	F	F	.	.
2			AY BA BB BC BD	0	F	F	F	.	.
2			BE BF BG BH BL	0	F	F	F	.	.
2			BM BP BQ BR BT	0	F	F	F	.	.
2			BV BX BY BZ CA	0	F	F	F	.	.
2			CB CD CE CF CG	0	F	F	F	.	.
2			CI CJ CK CM CN	0	F	F	F	.	.
2			CO CS CT CU CV	0	F	F	F	.	.
2			CW CY DA DK DØ	0	F	F	F	.	.
2			DJ DR EC EG EI	0	F	F	F	.	.
2			EK EL ES ET FA	0	F	F	F	.	.
2			FG FI FJ FO FP	0	F	F	F	.	.
2			FR FS GA GE GE	0	F	T	F	.	.
2			GH GI GJ GL GP	0	F	.	F	.	.
2			GQ GR GS GT GV	0	F	.	F	.	.
2			GY GZ HA HK HM	0	F	.	F	.	.
2			HO IC ID IH ID	0	F	.	F	.	.
2			IØ IR IS IT IV	0	F	.	F	.	.
2			IY IZ JA JM JO	0	F	.	F	.	.
2			JS KE KS KT KU	0	F	.	F	.	.
2			LA LE LI LS LT	0	F	.	F	.	.
2			LU LY MA ME MC	0	F	.	F	.	.
2			MH MI ML MN MØ	0	F	.	F	.	.
2			MP MQ MR MT MU	0	F	.	F	.	.
2			MV MX MY MZ NA	0	F	.	F	.	.
2			NC NE NF NG NH	0	F	.	F	.	.
2			NI NL NØ NP NR	0	F	.	F	.	.
2			NS NU NZ PA PC	0	F	.	F	.	.

-TG	N	S	DATA FIELD	LC	AL	GE	UK	USE3	USA
2			PE PG PK PM PØ	0	F	.	F	.	.
2			PP PQ PU QA RE	0	F	.	F	.	.
2			RH RP RQ RW SA	0	F	.	F	.	.
2			SB SC SE SF SG	0	F	.	F	.	.
2			SH SL SM SN SØ	0	F	.	F	.	.
2			SP ST SU SW SY	0	F	.	F	.	.
2			SZ TC TD TH TK	0	F	.	F	.	.
2			TL TN TØ TP TQ	0	F	.	F	.	.
2			TS TU TV TW TZ	0	F	.	F	.	.
2			UG UK UV UY VC	0	F	.	T	.	.
2			VE VI VM VØ VT	0	F	.	.	.	.
2			WA WF WJ WØ WS	0	F	.	.	.	.
2			WZ YE YØ YS ZA	Ⓧ	F	.	.	.	.
2			US	A	F	.	.	T	T
R1	2	3	2 -- 9	A	.	.	.	T	F
R2	1		Ø1 -- 56	-	.	.	.	T	.
					FAIL	PASS	PASS	PASS	FAIL

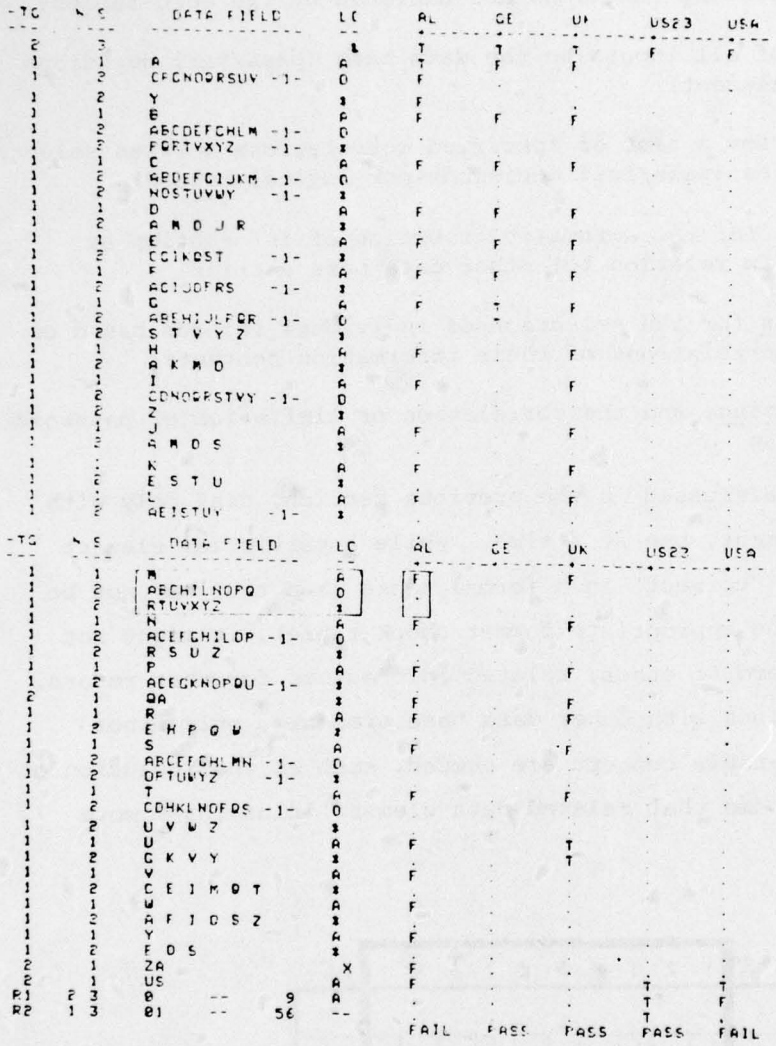
(2) COMPACT TABLE 'COUNTRY CODES'

This table is much shorter (full table reproduction). In addition, the number of data fields that must be checked is greatly reduced. The latter is accomplished by testing, in the first row of the table, whether or not the 3rd and 4th character is 'blank' (i.e. a character group of G=2, taken N=1 times, starting with the character position S=3 of the input). This separates all 2-digit foreign country codes from those of the 4-digit US codes. If the first data field is F (indicating a possible 'US' code) then all data fields containing 'foreign' (2-digit) codes will be skipped (.) by the table evaluation in program.

The improved evaluation pattern is achieved by the chosen Logical Connectors Ⓐ through Ⓧ. The first data field 'AND' one of the following data fields must be 'T' to satisfy the format check. Conversely, if the first data field is 'F' then the subsequent data entries need not be checked. (Even if they were right, the format check would not be complete). In this situation the program skips all subsequent data fields until the higher level OR Ⓧ is encountered.

Furthermore, the rather lengthy listing of 56 individual 'US' codes in solution (1) has been reduced to three lines in this table:

- The first line checks whether the first 2 characters are 'US'
- The second line tests whether the 3rd and 4th character are truely numbers. This prevents an input of, say, 'US3' instead of the correct input USØ3
- The third line checks whether the 2 digit numerical code falls within the appropriate range from 01 to 56



-TG	N	S	DATA FIELD	LC	AL	GE	UK	US23	USA
2		3		&	T	T	T	F	F
1		1	A	A	T	F	F		
1		2	CFGNOQRSUV -1-	O	F				
1		2	Y	*	F				
1		1	B	B	F	F	F		
1		2	ABCDEFGHLM -1-	O					
1		2	FQRTVXYZ -1-	*					
1		1	C	A	F	F	F		
1		2	ABDEFGIJKM -1-	O					
1		2	NOSTUVWY -1-	*					
1		1	D	A	F	F	F		
1		2	A M O J R	*					
1		1	E	A	F	F	F		
1		2	CGIKOST -1-	*					
1		1	F	A	F	F	F		
1		2	ACIJOPRS -1-	*					
1		1	G	A	F	T	F		
1		2	ABEHIJLPQR -1-	O		-			
1		2	S T V Y Z	*					
1		1	H	A	F		F		
1		2	A K M O	*					
1		1	I	A	F		F		
1		2	CDNOQRSTVY -1-	O					
1		2	Z	*					
1		1	J	A	F		F		
1		2	A M O S	*					
1		1	K	A	F		F		
1		2	E S T U	*					
1		1	L	A	F		F		
1		2	AEISTUV -1-	*					

-TG	N	S	DATA FIELD	LC	AL	GE	UK	US22	USA
1		1	M	A	F		F		
1		2	ABCHILNOPQ	O					
1		2	RTUVXYZ	*					
1		1	N	A	F		F		
1		2	ACEFGHILOP -1-	O					
1		2	R S U Z	*					
1		1	P	A	F		F		
1		2	ACEGKNOPQU -1-	*					
2		1	QA	*	F		F		
1		1	R	A	F		F		
1		2	E H P Q W	*					
1		1	S	A	F		F		
1		2	ABCEFGHLMN -1-	O					
1		2	OPTUWYZ -1-	*					
1		1	T	A	F		F		
1		2	CDHKLNOPQS -1-	O					
1		2	U V W Z	*					
1		1	U	A	F		T		
1		2	C K V Y	*			T		
1		1	V	A	F				
1		2	C E I M O T	*					
1		1	W	A	F				
1		2	A F I O S Z	*					
1		1	Y	A	F				
1		2	F O S	*					
2		1	ZA	X	F				
2		1	US	A	F			T	T
R1	2	3	0 -- 9	A				T	F
R2	1	3	01 -- 56	--				T	
					FAIL	PASS	PASS	PASS	FAIL

(3) FAST WORKING TABLE 'COUNTRY CODES'

This table design goes one step further in that the two character combinations of the foreign country codes have been split up to achieve greater processing economy.

For example, take the first three lines of the second part of the table:

M	A
ABCHILNOPQ	O
RTUVXYZ	*

The first character 'M' together (AND) with one of the following characters (OR) of the next two lines form the acceptable country code combinations:

MA MB MC MH MI ML . . . MZ

However, if the first character of the given input does not match 'M', then the characters of the following two lines need not be checked. This arrangement of course, further speeds up the evaluation of this format check table (although the table itself is somewhat longer than the one in solution (2).

Note that all edit tables (1), (2), and (3), despite their different structures and processing performance always lead to the same format test results.

3.4.4.7 Logical Check Tables

At present, the following functions are achieved by the edit tables:

1. Format Checks of all inputs to the data base (pass/fail decisions per data base element)

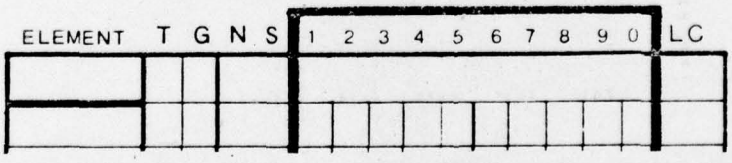
2. Logical Checks for a test of specified correlations between selected data base entries (pass/fail decisions per logical record)

3. Computed Inputs for the automatic insertion of information as triggered (and in relation to) other data base entries

4. Boolean Searches for the selection of individual records based on specific intercorrelations of their information contents

5. Password Processing, and the correlation or limitation of password/analyst functions

The format checks, discussed in the previous section, deal only with a specific data base element, one at a time. While a particular element entry or update might be 'correct' in a formal sense (and thus may not be 'detected as false' by the appropriate format check table), it might not make much sense with regard to other, related information for that record. To test those interrelations with other data base elements, only minor extensions to the format-table concept are needed, such as the inclusion of an 'Element' column (to name that related data element) plus three more logical connectors.



IF - IF
TH - THEN
SE - SET

The choice of IF, TH, and SE as addition to the array of logical connectors has been deliberate. It makes the design of logical tables quite simple by letting the procedures closely resemble the logic inherent in the English language (if, then, set). It also is a completely sufficient extension to boost the available table capabilities from mere format checks to the most complex logical procedures imaginable.

In this context one should note that the above list of functions only represents the current use of such tables. Much more is possible, from problems such as proper hyphenation to complex language processing or graphic design tasks based on sets of correlations between given information and constraints. In fact, it is recommended to reserve for the table approach all those logical functions of the system that the user wishes ultimately to control. The resulting software (using such tables) is relatively uncomplicated, and also very stable in time since all that may change are those tables (which are treated as data and not as programs!). Thus the cost of 'experimenting' is greatly reduced, and the user can truly attain maximum utility of his system in a convenient, on-line environment that lets him try, decide, and continue to bring out the best.

3.4.4.8 Sample of Logical Checks

The following four sample sheets concern the data base element GF27 "CATEGORY CODE." They show the spearation between format and logical checks, and how the latter can be formulated, implemented and tested to fulfill the requirements specified by the user.

ELT	ELEMENT NAME	REQUIRED EDITS
GF27	Category Code	Alpha-Numeric. Allowable codes are: 80000, 80010, 80020, 80040, 80050, 80060, 80070, 80080, 80090, 80100, 80110, 80120, 80130, 80150, or 5 blanks. Blanks are only allowed when first two characters of GF23 are 'US'. If entry is 80010, 80020 or 80040: GD12 must be 'I' If entry is '80060' or '80080': AR02 (Primary Rwy) must be ≥ 7070 AR05 must be 'P' If entry is '80070' or 80090': AR02 (Primary Rwy) must be 4970-7069 AR05 must be 'P' If entry is '80100': AR02 (Primary Rwy) must be 3970-4969 AR05 must be 'P' OR AR02 (Primary Rwy) must be ≥ 4970 AR05 must be 'T' or 'N' If entry is '80110': AR02 (Primary Rwy) must be 2970-3969 AR05 must be 'P' OR AR02 (Primary Rwy) must be 2970-4969 AR05 must be 'T' or 'N' If entry is '80120: AR02 (Primary Rwy) must be < 2970 GD12 must be 'I'

Format Checks for GF27

Logical Checks for GF27

The update information for a specific data element has already passed a pertinent Format Table before entering the Logical Check table. At this point the input must be considered 'true', at least from a 'format' point of view.

There are many instances, however, where a format-correct input could be 'false' (or make no sense) if correlated with other information in the record. The 'Required Edits' of the table to the right specify in detail the valid interrelations between various data base elements.

To implement and test those conditions, software for the input, processing, and testing of 'Logical Tables' have been written. The following pages show and discuss implementation and test sessions of the 'GF27' Logical Table.

TRANSLATION OF EDIT REQUIREMENTS INTO TABULAR FORM

	ELEMENT	T	G	N	S	1	2	3	4	5	6	7	8	9	0	LC	
*	GF27		5													IF	1st line must be IF or SE
			5													TH	IF GF27 is 'blank' THen GF23 must be 'US'
	GF23		2			U	S									IF	
			5			8	Ø	Ø	1	Ø	8	Ø	Ø	2	Ø	0	IF GF27 is 80010, 80020 OR 80040 THen GD12 must be 'I'
			5			8	Ø	Ø	4	Ø						TH	
	GD12		1			I										IF	
			5			8	Ø	Ø	6	Ø	8	Ø	Ø	8	Ø	TH	
	AR02		5				7	Ø	7	Ø	2	Ø	Ø	Ø	Ø	A	
	AR05		1			P										IF	
			5			8	Ø	Ø	7	Ø	8	Ø	Ø	9	Ø	TH	
	AR02	N	5				4	9	7	Ø		7	Ø	6	9	A	
	AR05		1			P										IF	
			5			8	Ø	1	Ø	Ø						TH	IF GF27 is 80100 THen AR02 Num. Range: 3970-4969 And AR05 must be 'P' Or AR02 Num. Range: 4970-20000 And AR05 must be 'T' or 'N'
	AR02	N	5				3	9	7	Ø		4	9	6	9	A	
	AR05		1			P										*	
	AR02	N	5				4	9	7	Ø	2	Ø	Ø	Ø	Ø	A	
	AR05		1			T	N									IF	
			5			8	Ø	1	1	Ø						TH	
	AR02	N	5				2	9	7	Ø		3	9	6	9	A	
	AR05		1			P										*	
	AR02	N	5				2	9	7	Ø		4	9	6	9	A	
	AR05		1			T	N									IF	
			5			8	Ø	1	2	Ø						TH	
	AR02	N	5							Ø		2	9	7	Ø	A	
	GD12		1			I										IF	

*In case of unspecified element fields the program substitutes the element name of the first row.

10 1	1	2 3	4 5 6 7 8	9
ELEM	-TG	N S	1234567890	LC
----	---	----	----------	--
GF27	5			IF
	5			TH
GF23	2		US	IF
	5		8001080020	O
	5		80040	TH
GD12	1		I	IF
	5		8006080080	TH
AR02	5		707020000	A
AR05	1		P	IF
	5		8007080090	TH
AR02	N5		4970 7069	A
AR05	1		P	IF
	5		80100	TH
AR02	N5		3970 4969	A
AR05	1		P	*
AR02	N5		497020000	A
AR05	1		TN	IF
	5		80110	TH
AR02	N5		2970 3969	A
AR05	1		P	*
AR02	N5		2970 4969	A
AR05	1		TN	IF
	5		80120	TH
AR02	N5		0 2970	A
GD12	1		I	IF
	5		80130	TH
GD10	1		I	A
GD12	1		I	IF
	5		80150	TH
GD12	1		I	--

The table design of the previous page may be implemented on-line via the terminal using the program 'LHEAD' (which provides header lines and sets the appropriate tabs); or via the card reader using program 'CDT'.

The hand-drawn lines facilitate interpretation of the various IF - THEN 'sentences' Note that within each IF and THEN group the full range of logical connectors can be employed, from low to high level AND's and OR's. This allows the creation of logical structures within each IF or THEN sequence, providing ample capabilities for handling numerous complex logical intercorrelations.

The basic rules for the use of logical tables are simple, however. Only if the listed element matches the specified IF condition (T) must the subsequent THEN condition(s) also be tested and fulfilled (T) in order to 'PASS' the logical table. In other words whenever the IF condition is (F), the subsequent TH entries need not be tested at all. This can readily be seen by inspecting the test session print out of the following page.

LC - LOGICAL CONNECTORS

IF - IF
TH - THEN

O - OR Low Level
A - AND

* - OR
& - AND

X - OR
E - AND High Level

NAME OF TABLE -- XGF27

TEST INPUTS: -- 80000 -- 80020 -- 80100 -- 80110 --

Up to 5 test inpust may be used

```
AR02  -- 4000
AR05  -- T
GD10  -- I
GD12  -- I
GF23  -- US
GF27  -- **
```

The test program considers all elements mentioned in the table. After sorting -- and the elimination of all duplicate entries -- the remaining element labels are displayed for data input by the test personnel

A '**' input allocates the five inputs to the specified element (GF27)

ELEM	-IG	N S	DATA FIELD	LC	80000	80020	80100	80110
GF27	5			IF	.	.	.	.
	5			TH	F	F	F	F
GF23	2		US	IF	.	.	.	.
	5		80010 80020	0	F	(T)	F	F
	5		80040	TH	F	.	F	F
GD12	1		I	IF	.	T	.	.
	5		80060 80080	TH	F	F	F	F
AR02	5		7070 20000	A	.	.	.	.
AR05	1		P	IF	.	.	.	.
	5		80070 80090	TH	F	F	F	F
AR02	N5		4970 == 7069	A	.	.	.	.
AR05	1		P	IF	.	.	.	.
	5		80100	TH	F	F	(T)	F
AR02	N5		3970 == 4969	A	.	.	T	.
AR05	1		P	*	.	.	F	.
AR02	N5		4970 == 20000	A	.	.	(F)	.
AR05	1		T N	IF	.	.	.	.
	5		80110	TH	F	F	.	(T)
AR02	N5		2970 == 3969	A	.	.	.	F
AR05	1		P	*	.	.	.	.
AR02	N5		2970 == 4969	A	.	.	.	T
AR05	1		T N	IF	.	.	.	T
	5		80120	TH	F	F	.	F
AR02	N5		0 == 2970	A	.	.	.	.
GD12	1		I	IF	.	.	.	.
	5		80130	TH	F	F	.	F
GD10	1		I	A	.	.	.	.
GD12	1		I	IF	.	.	.	.
	5		80150	TH	F	F	.	F
GD12	1		I	--	.	.	.	.
					PASS	PASS	FAIL	PASS

3.4.4.9 Sample of Computed Inputs

The next four sheets give an example for a computed input to the data base. The rules of intercorrelation with other data are so complex, that the user wanted to eliminate human input altogether in order to achieve an accurate categorization for NNØ1, "Navigation/Communication Aids." The edit requirements for this data field depend upon the entries in NNØ2 through NN19.

The computer sets (Logical Connector 'SE') the data base element NNØ1 to the information listed in the Data Field, if the preceding IF-sentence is 'true' (T). By starting with the entries of lowest priority it is ensured that the last 'setting' of NNØ1 is indeed that of highest priority for the record in question. To simulate the three groups A, B, C of the edit requirements listed, variables named AAA, BBB, and CCC have been used in the following tables. They are 'initialized' to 'Ø' at the start of the table to ensure their proper functions.

SUBCATEGORY: NAVIGATION/COMMUNICATION AIDS

ELT	ELEMENT NAME	REQUIRED EDITS
NNØ1	N/C Aids	Alpha-Numeric. Generated.

GROUP	If one or more of these elements contain an 'A'	This code will be generated
A	NNØ8, NNØ9, NN14	1
B	NN1Ø, NN11, NN12, NN13, NN19	2
C	NN16, NN17, NN18	3

If one or more of the elements in this group contains an 'A'	AND one or more of the elements in this group contains an 'A'	This code will be generated
A	B	4
A	C	5
B	C	6

If one or more of the elements in all groups contain an 'A' the generated code will be 7.

If none of the groups contain an 'A' the highest priority availability code will be generated. Descending priority is 'E', 'U' then 'N'.

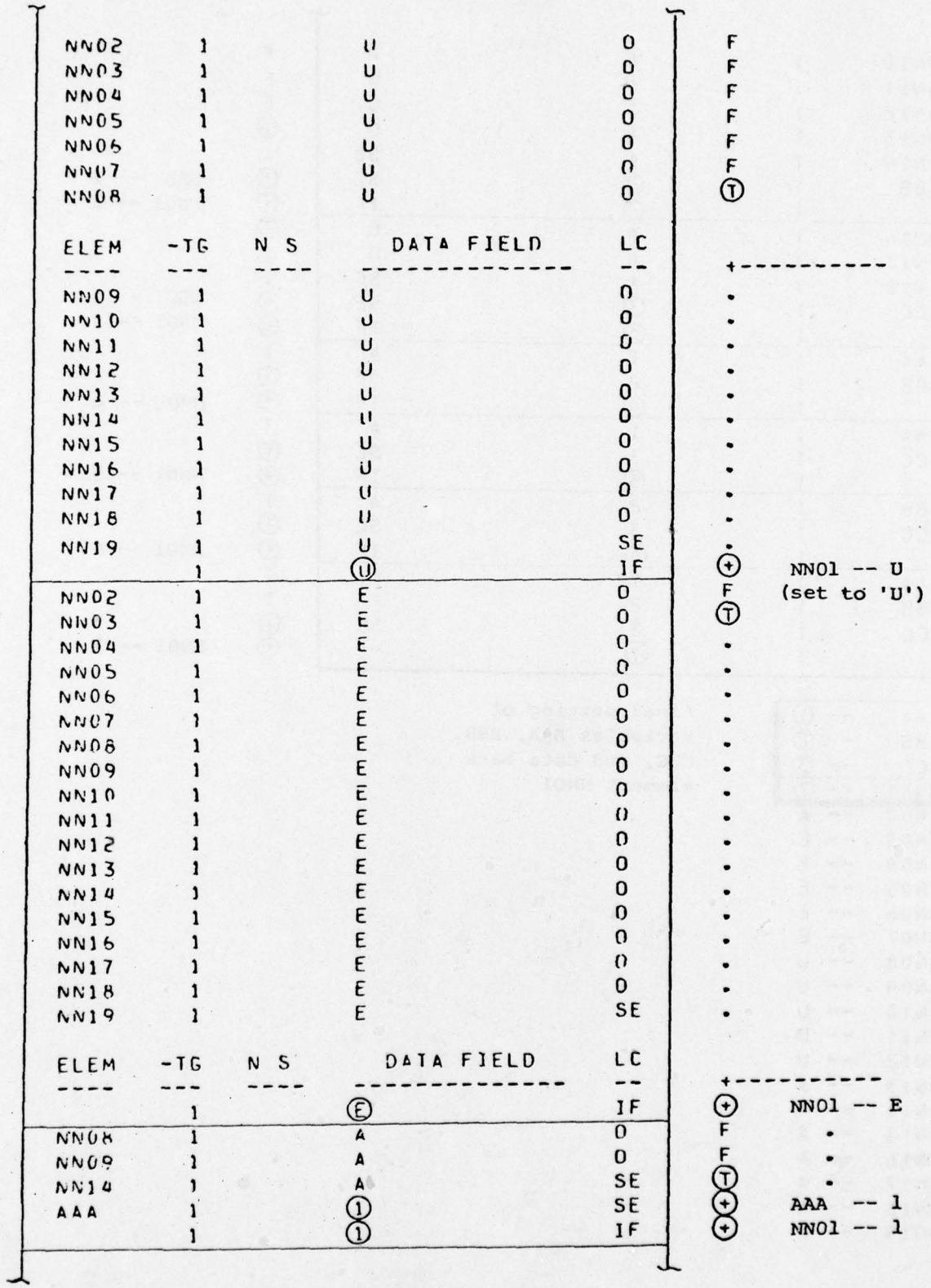
NAME OF TABLE -- LNN01

AAA	--	9
BBB	--	9
CCC	--	9
NN01	--	9
NN02	--	A
NN03	--	E
NN04	--	E
NN05	--	E
NN06	--	E
NN07	--	E
NN08	--	U
NN09	--	U
NN10	--	U
NN11	--	U
NN12	--	U
NN13	--	A
NN14	--	A
NN15	--	A
NN16	--	A
NN17	--	A
NN18	--	A
NN19	--	A

For this test elements NN02 through NN19 of the data base are presumed to be at the the values indicated at left.

NN01 must be set -- and with it variables AAA, BBB, CCC -- (Computed inputs). Their current values are presumed to be an (impossible) '9', i.e. a residual computer value at these memory locations.

ELEM	-TG	N S	DATA FIELD	LC	
NN01	1			SE	.
AAA	1		(0)	SE	⊕ Initialization of
BBB	1		(0)	SE	⊕ AAA, BBB, CCC
CCC	1		(0)	IF	⊕ to 'Ø'
NN02	1		N	0	F
NN03	1		N	0	F
NN04	1		N	0	F
NN05	1		N	0	F
NN06	1		N	0	F
NN07	1		N	0	F
NN08	1		N	0	F
NN09	1		N	0	F
NN10	1		N	0	F
NN11	1		N	0	F
NN12	1		N	0	F
NN13	1		N	0	F
NN14	1		N	0	F
NN15	1		N	U	F
NN16	1		N	0	F
NN17	1		N	0	F
NN18	1		N	0	F
NN19	1		N	SE	F
	1		N	IF	.


NN02 1 U 0 F
NN03 1 U 0 F
NN04 1 U 0 F
NN05 1 U 0 F
NN06 1 U 0 F
NN07 1 U 0 F
NN08 1 U 0 T
ELEM -TG N S DATA FIELD LC
NN09 1 U 0
NN10 1 U 0
NN11 1 U 0
NN12 1 U 0
NN13 1 U 0
NN14 1 U 0
NN15 1 U 0
NN16 1 U 0
NN17 1 U 0
NN18 1 U 0
NN19 1 U SE
1 U IF + NN01 -- U
NN02 1 E 0 F (set to 'U')
NN03 1 E 0 T
NN04 1 E 0
NN05 1 E 0
NN06 1 E 0
NN07 1 E 0
NN08 1 E 0
NN09 1 E 0
NN10 1 E 0
NN11 1 E 0
NN12 1 E 0
NN13 1 E 0
NN14 1 E 0
NN15 1 E 0
NN16 1 E 0
NN17 1 E 0
NN18 1 E 0
NN19 1 E SE
ELEM -TG N S DATA FIELD LC
1 E IF + NN01 -- E
NN08 1 A 0 F
NN09 1 A 0 F
NN14 1 A SE T
AAA 1 1 SE + AAA -- 1
1 1 IF + NN01 -- 1

NN10	1	A	0	F	
NN11	1	A	0	F	
NN12	1	1	0	F	
NN13	1	A	0	T	
NN19	1	A	SE	•	
BBB	1	②	SE	⊕	BBB -- 2
	1	②	IF	⊕	NN01 -- 2
NN16	1	A	0	T	
NN17	1	A	0	•	
NN18	1	A	SE	•	
CCC	1	③	SE	⊕	CCC -- 3
	1	③	IF	⊕	NN01 -- 3
AAA	1	1	A	T	
BBB	1	2	SE	T	
	1	④	IF	⊕	NN01 -- 4
AAA	1	1	A	T	
CCC	1	3	SE	T	
	1	⑤	IF	⊕	NN01 -- 5
BBB	1	2	A	T	
CCC	1	3	SE	T	
	1	⑥	IF	⊕	NN01 -- 6
AAA	1	1	A	T	
BBB	1	2	A	T	
CCC	1	3	SE	T	
	1	⑦	--	⊕	NN01 -- 7

AAA	-- ①
BBB	-- ②
CCC	-- ③
NN01	-- ⑦
NN02	-- A
NN03	-- E
NN04	-- E
NN05	-- E
NN06	-- E
NN07	-- E
NN08	-- U
NN09	-- U
NN10	-- U
NN11	-- U
NN12	-- U
NN13	-- A
NN14	-- A
NN15	-- A
NN16	-- A
NN17	-- A
NN18	-- A
NN19	-- A

Final setting of variables AAA, BBB, CCC, and data base element NN01

3.4.4.10 Boolean Retrieval

The logical tables may also be used to perform Boolean Searches over parts of the data base, or the full data base (worldwide search). This is accomplished by listing the desired conditions (or value ranges) of selected data base elements in an IF sentence, followed by a SET command for an output variable. In this way the record identification (airfield) can be listed in a special retrieval file for output processing (report generation or type) whenever the output variable has been set.

For a simple example of a Boolean Search table, see the following page.

SAIR REQUESTS:

From full AAFIF file print all airfields with hard surface runway of 5000 feet or greater in the following countries:

1. South Africa (SF)
2. Rhodesia (RH)
3. Uganda (UG)

The above boolean search request involves the data base elements:

GF23 - COUNTRY CODES: SF RH UG
AR02 - RUNWAY LENGTH: 5000 to 20000 (Max)
AR04 - SURFACE COMPOSITION: ASP BIT CON (Hard)

The following simple table thus expresses all conditions for the above boolean search request.

ELEMENT	T	G	N	S	1 2 3 4 5 6 7 8 9 0	LC
						IF
GF23		3		1	S F R H U G	A
AR02	N	5			5 0 0 0 2 0 0 0	A
AR04		3			A S P B I T C O N	SE
OUT		1			1	--

Please note that the output variable OUT gets set to '1' only if all logical conditions specified are met (the table rows are interconnected by logical AND's). If OUT is set to '1', the program triggers the output of the record in question.

3.4.4.11 Edit Tables Loaded to the System

The following is the beginning of the list of edit tables as currently loaded to the Air Facilities System. The printed list EREPRT has been created by program EPRT while converting the edit tables, filed in RDOS, into the final form and loading them to INFOS. The printout also contains error messages if formal errors have been detected. The presented pages represent only a small sample.

The table design has been based on the specified user requirements as outlined in the document "AAFIF EDIT REQUIREMENTS, GENERAL INFORMATION", 12 December 1977.

FORMAT TABLES

LOGICAL TABLES

```
   1  RDOS FORMAT TABLE FGF03
1      2 3     4 5 6 7 8     9
-TG    N S     1234567890    LC
---    ----    ----------    --
  1      1     ABCDEGHX      -
```

GF03

```
                1  RDOS LOGICAL TABLE LGF03

10 1     1       2 3     4 5 6 7 8     9
ELEM     -TG     N S     1234567890    LC
----     ---     ----    ----------    --
GF03       1                           IF
GF03       1             ABC           TH
MM01     -01             UN            A
GF10     -01             N             A
GF27     -05             8013080150    A
GD02     -01             D             A
GD03     -01             34D           IF
GF03       1             D             TH
GF10     -01             N             IF
GF03       1             E             TH
GC01       1             N             A
GF10     -01             N             IF
GF03       1             G             TH
GC01       1             N             A
GF10       1             N             A
GD10       1             I             A
GD11       1             I             A
GD12       1             I             A
GD13       1             I             A
GF27       5             80130         A
N401       1             N             A
```

```
AA03      1                N                A
AA04      1                N                A
AA05      1                N                A
AA06      1                N                TF
GF03      1                H                TH
GF27      5                80150            IF
GF03      1                X                TH
GF27      5                80130            A
GF15      1                D                IF
GF27      5                                 TH
GF23      2         1      US               IF
GF27      3         3      010020040        TH
GD12      1         1      I                IF
GF27      3         3      060080           TH
AR02     N5         1      0707099999       A
AR02    -01      5  1      .                IF
GF27      3      1         070090           TH
AR02     N5         1      0497007069       A
AR02    -01      5  1      .                A
AR05      1      1  1      P                IF
GF27      3         3      100              TH
AR02     N5         1      0397004969       A
AR02    -01      5  1      .                A
AR05      1      1         P                *
AR02     N5                0497099999       A
AR02    -01      5  1      .                A
AR05      1      1  1      TN               IF
GF27      3         3      110              TH
AR02     N5         1      0297003969       A
AR02    -01      5  1      .                A
AR05      1      1         P                *
AR02     N5                0297004969       A
AR02    -01      5  1      .                A
AR05      1      1         TN               IF
GF27      3         3      120              TH
AR02     N5         1      0000002969       A
AR02    -01      5  1      .                A
GD12      1      1  1      I                IF
GF27      3         3      130              TH
GD10      1         1      I                A
GD12      1         1      I                A
GF03      1         1      G                IF
GF27      3         3      150              TH
GD12      1         1      I                A
GF03      1         1      H                --
```

```
    2  RDOS FORMAT TABLE GF04
1       2 3     4 5 6 7 8     9
-TG     N S     1234567890    LC
---     ----    ----------    --
  1       1     U             A
  2       2                   *
  1       1     CS            A
  2       2       .FE4E.E     0
  2       2     E-EGEEECEJ    0
  2       2     FP.I          --
```

```
    3  RDOS FORMAT TABLE GF05
1       2 3     4 5 6 7 8     9
-TG     N S     1234567890    LC
---     ----    ----------    --
  4     1 1     000E          *
-04             000.          A
 N3     1 1     000   179     A
-01     3       .             A
  1     1 4     E.            --
```

```
    4  RDOS FORMAT TABLE GF06
1       2 3     4 5 6 7 8     9
-TG     N S     1234567890    LC
---     ----    ----------    --
  1               .S          --
```

```
                4  RDOS LOGICAL TABLE LGF06
;
10 1    1       2 3     4 5 6 7 8     9
ELEM    -TG     N S     1234567890    LC
----    ---     ----    ----------    --
GF06      1                           IF
          1             M             TH
GD11      1             A             A
GD13      1             I             IF
          1             S             TH
GD02      1             ABCDU         A
GD03      1             1234U         A
GD11      1             I             A
GD13      1             A             -
```

3.4.4.12 Edit Table Processing

Summing up the detailed descriptions and samples given in previous chapters, it can be stated that the following four classes of functions are handled by the edit tables:

- ✓ Format Checks of all inputs to the data base (pass/fail decisions per data base element).
- ✓ Logical Checks for a test of specified correlations between selected data base entries (pass/fail decisions at the conclusion of an airfield update).
- ✓ Computed Inputs for the automatic insertion of information as triggered (and in relation to) other data base entries.
- ✓ Boolean Searches for the selection of individual records based on specific intercorrelations of their information contents.

There are three programs for edit table inputs (RDOS):

FHEAD - Header and tab settings for FORMAT table inputs.

LHEAD - Header and tab settings for LOGICAL table inputs.

CEDT - Program for 'wholesale' table inputs via the card reader.

and two major software packages:

ET - For input conversion, system loading to INFOS on-line testing, and the modification of specific edit tables.

EPRT - For the 'wholesale' INFOS loading of all current edit tables in RDOS required by the system control information contained in the 'Label Book.'

3.4.4.12.1 ET - Edit Table Test Package

The ET package allows the functional verification of all edit tables as they are entered into the system. It displays the actual processing and response to various classes of inputs by showing in detail whether, where, and why a table lets correct inputs pass while incorrect ones are rejected.

The built-in features serve two functions: (1) that of proper table design with an implicit training of those analysis who are responsible for establishing and maintaining these system controls and (2) that of verifying and documenting the editing functions as originally laid down in the document AAFIF Edit Requirements, December 12, 1977.

The edit table evaluation programs used by the Air Facilities On-Line System are a subset of the ET package. Essentially the special test input and documentation features are removed while the table evaluation logic remains the same. In ths way it is guaranteed that the editing functions of the On-Line System are identical with those of the special test and verification package 'ET'.

Initialization Sequence:

```
DIR NEHL                ; get directory
ET                      ; load ET program package
```

System Response:

ENTER FILE NAME FOR STORING THE ON-LINE SESSION

START FILE NAME WITH S-- '''''

If a new file name is entered in the space provided (''''') a new file will be created for storing the on-line test session. If an old file name is entered, the old file will be appended (added on).

Although any entered filename (not used for another purpose) would be acceptable, it is recommended to begin all session file names with 'S', and to use the second character for identification of the systems analyst. In this way a CLI command for a sorted listing of all session files can readily be obtained.

For example:

```
LIST/S SN-              ; Sorted list of all session files
                          of analyst 'N'

      SNØ1
      SNØ2              ; The possible system response
      SNØ3                would indicate that 'N' has three
                          session files in the system.
```

'SNØ4' would be the next 'new' filname for analyst 'N'.

In the case of inconsequential or 'play' sessions, use of a common filename, e.g., SPLAY is recommended. This allows periodic deletions and thus avoids the proliferation of inconsequential files kept on disk. The command DELETE SPLAY

at the beginning of a session would remove previous contents and make 'SPLAY' a 'new' file.

Output

A complete RDOS back-up file of the on-line session will be created by the ET package. This back-up may be used for subsequent viewings at the Datagraphix terminal by entering the CLI command:

TYPE SNØ4 (filename);

or the production of a hardcopy printout:

PRINT SNØ4 (filename).

The printout can be used as a detailed working paper for (off-line) inspections, improvements, table changes, and preparation of the next on-line session at the system analyst's desk. Such use could significantly alleviate a crowded terminal schedule, and could improve both speed and quality of the 'next session' by allowing for a better preparation than would otherwise be possible.

All tables tested during an on-line session will automatically be loaded to INFOS for system use. A printout of the last test session (for a particular table) should consequently be kept, serving as a system's load and test evaluation document.

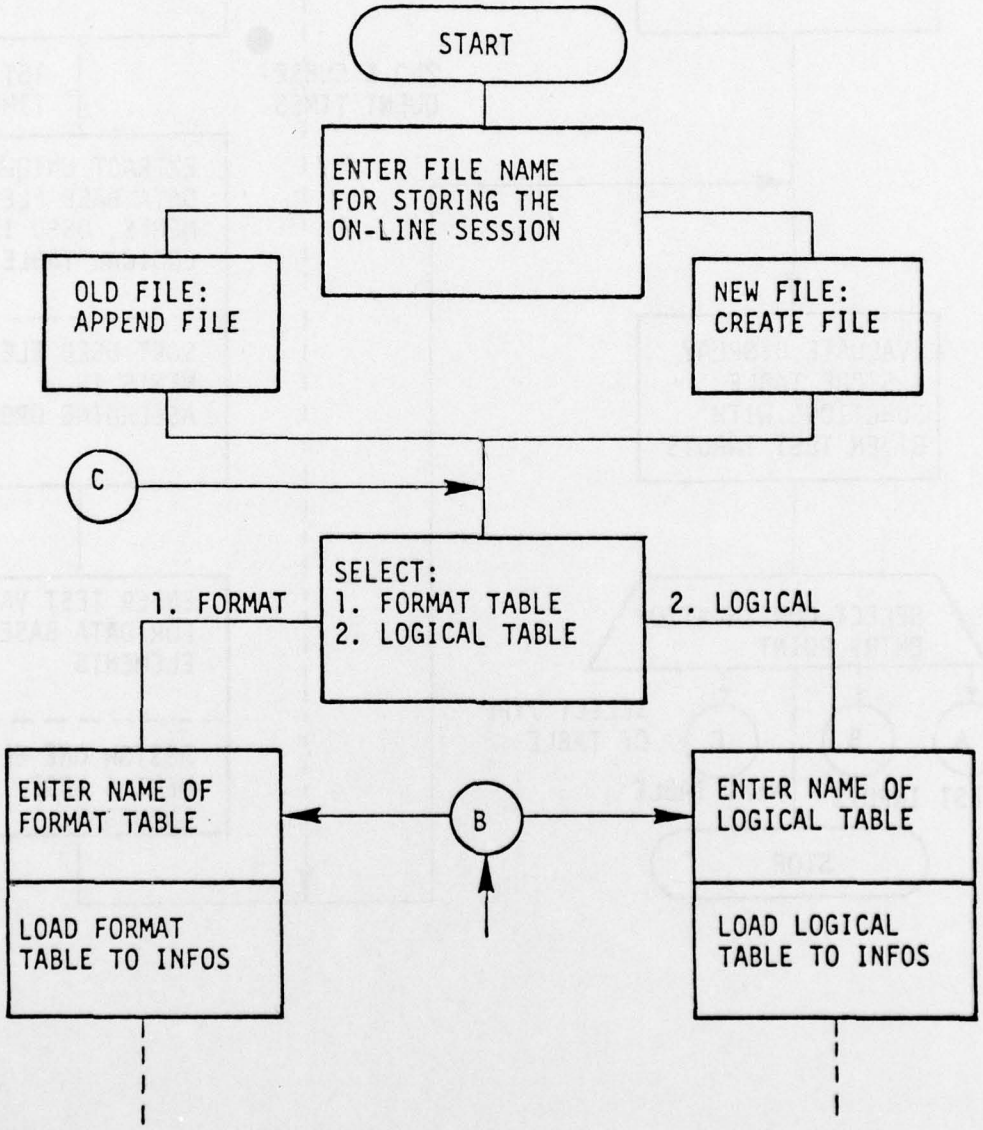
Tables/Files Accessed

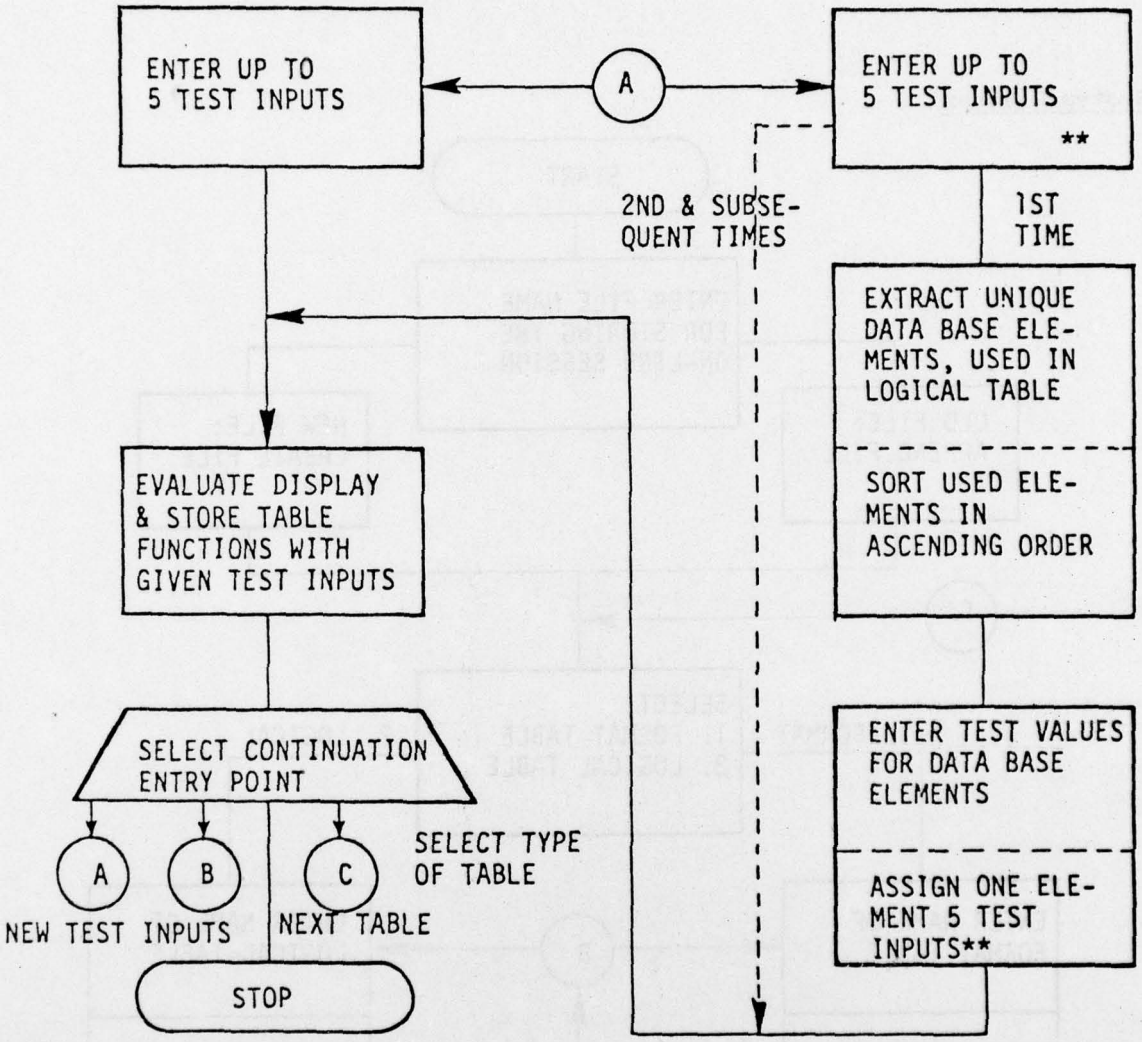
A logical record of the Air Facilities System has about 650 database elements, comprised of approximately 575 'fixed format' and 75 'free text' items. As a consequence, there will be about 575 format tables in the operating AF system for editing each updated or newly entered fixed format item.

In addition, there will be about 60 logical tables for the automatic insertion of 'computed inputs', or for testing whether the correlation of several database elements are within prescribed bounds and follow specified rules (i.e., make sense). While the format tables test each entry immediately upon completion in the operating AF system, the logical tables will be tested only at the end of a record update (i.e., when a new logical record is requested by the user).

All format and logical tables are entered as RDOS files. They can be displayed, tested, changed and reloaded to INFOS individually through the Edit Table program ET (for a wholesale loading of edit tables see program package EPRT).

System Diagram




ENTER UP TO
5 TEST INPUTS
A
ENTER UP TO
5 TEST INPUTS
**
2ND & SUBSE-
QUENT TIMES
1ST
TIME
EXTRACT UNIQUE
DATA BASE ELE-
MENTS, USED IN
LOGICAL TABLE
SORT USED ELE-
MENTS IN
ASCENDING ORDER
EVALUATE DISPLAY
& STORE TABLE
FUNCTIONS WITH
GIVEN TEST INPUTS
SELECT CONTINUATION
ENTRY POINT
ENTER TEST VALUES
FOR DATA BASE
ELEMENTS
A
B
C
SELECT TYPE
OF TABLE
NEW TEST INPUTS
NEXT TABLE
ASSIGN ONE ELE-
MENT 5 TEST
INPUTS**
STOP

3.4.4.12.2 EPRT - Edit Table Systems Loading

This program package converts into the final form and loads all format and Logical edit tables as required by the Label Book load module 'INLABEL'. It detects missing tables and formal table input errors.

Besides the loading to INFOS two output files 'ETLIST' and 'EREPRT' are created.

'ETLIST' provides a comprehensive checklist of loaded and missing tables. 'EREPRT' prints the RDOS names of all tables that have been loaded along with detected formal table errors. If the print option (display request at the start of the run) is invoked, the full edit table will also be printed. Format tables are printed on the left, Logical tables on the right-hand side of the printout.

3.4.5 Tables for Interactive Communications

The interactive section of the Air Facility system performs the function of permitting the user to access system on-line capabilities. Basically it consists of a sequence of displays printed at the Datagraphics 132A terminal. These displays request information from the user regarding system function selection, parameter insertion, data entry, as well as present error message display and further selection processing. Responses by the user are entered by depressing the appropriate key(s) at the keyboard.

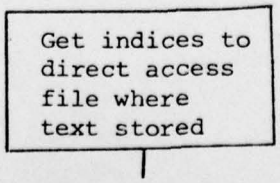
To perform these capabilities the following software modules have been implemented:

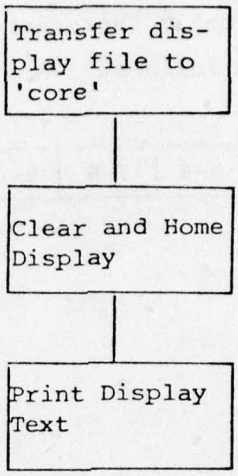
- o A Display Generator
- o A Character Read/Echo Module
- o A set of logical error check modules
- o An Error Communications Display Generator

Of these modules, the Display Generator and the Error Communications Display Generator make extensive use of tables. The data contained in these tables implement important program control functions. For this reason, their use and content is outlined in this section.

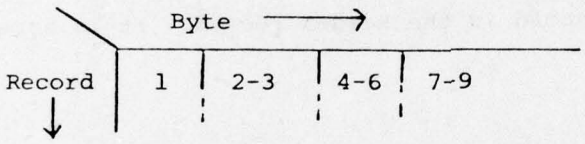
3.4.5.1 Display Generator

The Display Generator is a generalized software module that prints display text on the Datagraphics terminal. Its major capability is to be able to print any text stored in a formatted disk file, thus separating text from a source program. The macro algorithm for the module is as follows:





The indices for the displays are stored in a sequential access file with the following characteristics:



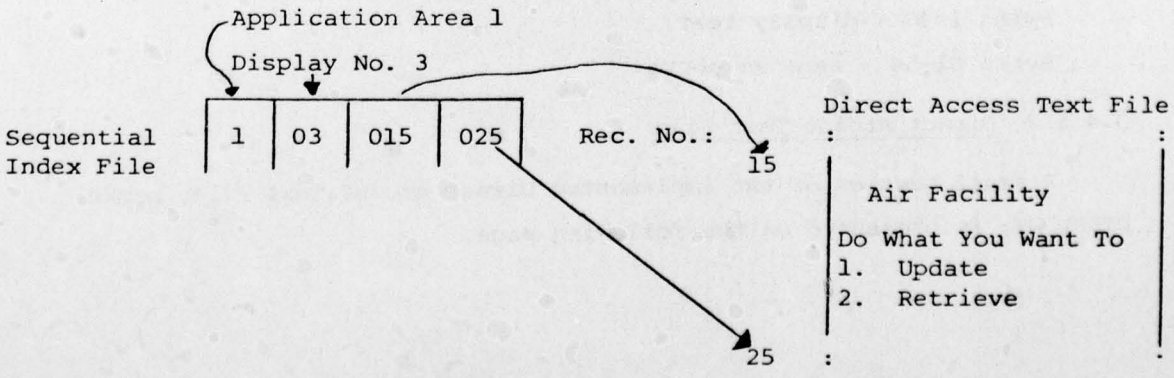
Byte 1 - character representing the particular application area, e.g., update, retrieval

Bytes 2-3 - character representing the particular display that is within an application

Bytes 4-6 - characters representing the record number in the direct access files where the particular display text starts

Bytes 7-9 - characters representing the record number in the direct access file where the particular display text ends.

For example:



The records that logically represent a display are stored on the Direct Access Text File as follows:

START

HEADER	1-4	5-6	7-8	9-40	41-46	47-52	52-54
TEXT	1-40				4 -46	47-52	52-54

.
.
.
.
.
.
.

END

The start or initial record is the header record. It is arranged as follows:

Bytes 1-4 - Display ID

Bytes 5-6 - Number of physical records in a logical display record.

Bytes 7-8 - Number of 'reads' to be performed from this display.

Bytes 9-40 - Display header information, e.g., 'AIR FACILITY UPDATE.'

Bytes 41-46 - Cursor position on the Datagraphics screen where the bytes are to be printed.

Bytes 47-52 - The cursor position where, if required, a read or data input is to occur. If not, it is filled with 'X's.'

Bytes 52-54 - The number of characters to be inputed if a read is to occur.

The actual text of the display is stored in records header +1 to the end as follows:

Bytes 1-40 - display text.

Bytes 41-54 - same as above.

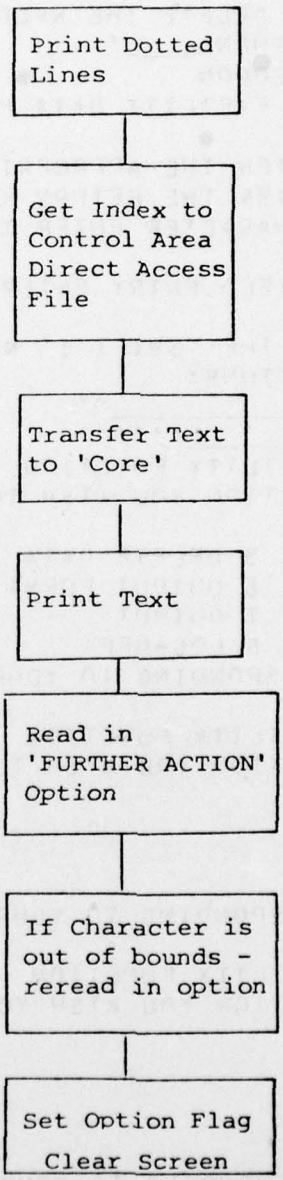
3.4.5.2 Direct Access Test File

A small section of the implemented Direct Access Text File, named DISPL.DT, is presented on the following page.

```
C---+----1----+----2----+----3----+----4----+----5----+
02010503AIR FACILITY DELETE                          002041XXXXXX00
TO DELETE AIRFIELD DATA ENTER :                      005021XXXXXX00
WAC NUMBER:_____'''''                                007021007047O4
INSTALLATION NUMBER:____''''''                       009021009045O6
PASSWORD_____''''''                                  011021XXXXXX00
02030301AIR FACILITY DELETE                          002041XXXXXX00
DEPRESS THE CR KEY TO DELETE THE RECORD              005021XXXXXX00
OR, ENTER A  R  TO RETURN:____'                      007021007051O1
01011302AIR FACILITY LOGON                           002041XXXXXX00
***WELCOME TO THE AIR FACILITY DATA MANA             004029XXXXXX00
GEMENT SYSTEM***                                     004069XXXXXX00
FOR REQUESTED DATA ENTER THE APPROPRIATE             008021XXXXXX00
KEYS,WHEN FINISHED ENTER THE RETURN KEY.             008062XXXXXX00
TO CORRECT THE LAST CHARACTER ENTER THE              010026XXXXXX00
'X' KEY.                                             010066XXXXXX00
TO CORRECT THE FULL FIELD ENTRY ENTER                012026XXXXXX00
THE 'SHIFT X' KEY.                                   012065XXXXXX00
TO ESCAPE ENTRY ENTER THE 'SHIFT !' KEY.             014026XXXXXX00
TO ACCESS SYSTEM FUNCTIONS:                          018021XXXXXX00
ENTER YOUR GEC-ID:_____''                            020031020062O2
ENTER YOUR PASSWORD:_____''''''                      022031022058O8
01020901SELECT ATR FACILITY FUNCTION                 002041XXXXXX00
PLEASE SELECT THE FUNCTION YOU WISH TO               004021XXXXXX00
PERFORM                                              004060XXXXXX00
1 DISPLAY DATA          5 DELETE DATA                006021XXXXXX00
2 UPDATE DATA           6 OUTPUT FORMAT              008021XXXXXX00
3 RETRIEVE DATA         7 OUTPUT                     010021XXXXXX00
4 ADD A AIRFIELD        8 LOG-OFF                    012021XXXXXX00
ENTER THE NUMBER CORRESPONDING TO YOUR               016021XXXXXX00
CHOICE:_____'                                        016061016073O1
01030801SELECT ATR FACILITY FUNCTION                 002021XXXXXX00
PLEASE SELECT THE FUNCTION YOU WISH TO               004021XXXXXX00
PERFORM                                              004060XXXXXX00
1 UPDATE DATA                                        006026XXXXXX00
2 DISPLAY DATA                                       008026XXXXXX00
3 ADD A RECORD                                       010026XXXXXX00
ENTER THE NUMBER CORRESPONDING TO YOUR               012021XXXXXX00
CHOICE_____'                                         012060012015O1
01040901SELECT ATR FACILITY FUNCTION                 002021XXXXXX00
PLEASE SELECT THE FUNCTION YOU WISH TO               004021XXXXXX00
PERFORM                                              004060XXXXXX00
1 UP???? DATA                                        006026XXXXXX00
2 DISPLAY DATA                                       008026XXXXXX00
3 ADD A RECORD                                       010026XXXXXX00
4 DELETE A RECORD                                    012026XXXXXX00
ENTER THE NUMBER CORRESPONDING TO YOUR               014021XXXXXX00
CHOICE_____'                                         014060014074O1
```

3.4.5.3 Error Communications Display Generator

The Error Communications Display Generator is a generalized software module that prints error messages and 'Further Action' prompts on the Datagraphics screen. It is accessed when a user inserts an invalid response to a prompt or when he enters 'shift ■ ' in character input. The macro algorithm for the module is as follows:



AD-A076 106 SYNECTICS CORP ROME N Y F/G 5/2
AUTOMATED AIR INFORMATION PRODUCTION SYSTEM, PHASE I, VOLUME II--ETC(U)
SEP 79 W NEHL , P MOULDER F30602-77-C-0065
UNCLASSIFIED RADC-TR-79-179-VOL-3 NL

2 OF 2
AD
A076106

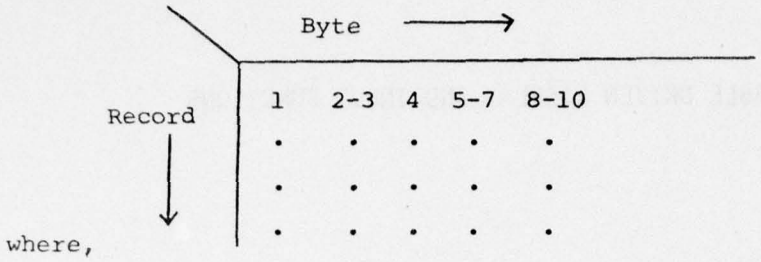
END
DATE
FILMED
11-79
DDC

TABLE DRIVEN DISPLAY AND INPUT FUNCTIONS

```
ENTER YOUR GEO-ID:_____''          02003102006202
ENTER YOUR PASSWORD:_____''''''    02203102205306
01020901SELECT AIR FACILITY FUNCTION   002041XXXXXX00
PLEASE SELECT THE FUNCTION YOU WISH TO 004021XXXXXX00
PERFORM                                004060XXXXXX00
1 DISPLAY DATA          5 DELETE A RECORD006021XXXXXX00
2 UPDATE DATA           6 OUTPUT FORMAT 008021XXXXXX00
3 RETRIEVE DATA         7 OUTPUT        010021XXXXXX00
4 ADD A RECORD          8 LOG-OFF       012021XXXXXX00
ENTER THE NUMBER CORRESPONDING TO YOUR 016021XXXXXX00
CHOICE:_____'                          01606101607301
01030801SELECT AIR FACILITY FUNCTION   002021XXXXXX00
PLEASE SELECT THE FUNCTION YOU WISH TO 004021XXXXXX00
PERFORM                                004060XXXXXX00
1 UPDATE DATA                          006026XXXXXX00
2 DISPLAY DATA                         008026XXXXXX00
3 ADD A RECORD                         010026XXXXXX00
ENTER THE NUMBER CORRESPONDING TO YOUR 012021XXXXXX00
CHOICE_________'                       01206001201501
```

- FUNCTION/AREA IDENTIFICATION
- DISPLAY TITLE
- DISPLAY TEXT
- DISPLAY SCREEN POSITIONING
- CURSOR POSITIONING
- CHARACTER INPUT CONTROL

The indices for the displays are stored in a sequential access file with the following characteristics:



where,

Byte 1 - character representing the particular application areas, e.g., update retrieval.

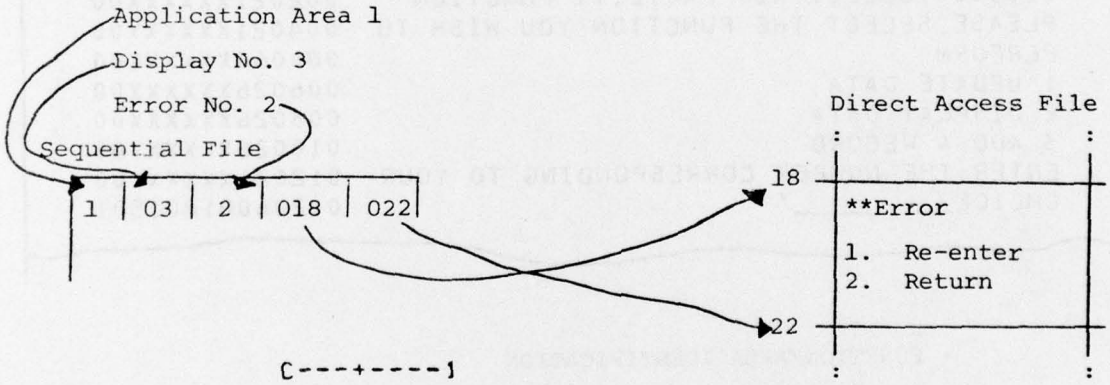
Bytes 2-3 - characters representing the particular display that is within an application.

Bytes 4 - character representing the particular error display for a display.

Bytes 4-7 - characters representing the record in the direct access file where the particular error text starts.

Bytes 8-10 - characters representing the record number in the direct access file where the particular error text ends.

For example:



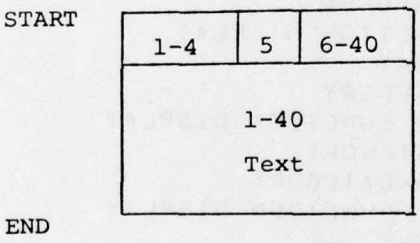

```
C---+----1
2011001003
2012004006
2021007009
1011010012
1012013016
1021017019
4021020022
6012023025
```

Index File INDIS.DT

```
6031026028
6011029031
7011032034
7012036040
2014041044
7013035037
6013045048
6014049052
7014053055
7015056058
7016059061
7017062064
7018065067
7019068070
7021071073
7022074076
7023077080
7024081084
7025085088
7026090093
7027094097
6013098100
3011101103
```

3.4.5.4 Error Text File

The records that logically represent an error area are stored on the *direct access file* as follows:



The start or intial record is the header record. It is arranged as follows:

Bytes 1-4 represent the error area ID.

Byte 5 represents the number of available 'Further Action' options.

Bytes 6-40 represent error message text.

Subsequent records store error message text.

Bytes 1-4 represent the error area ID.

Byte 5 represents the number of available 'Further Action' options.

Bytes 6-40 represent error message text.

Subsequent records store error message text. The following is a sample of the first part of the Error Text File, loaded to the system:

```
C---+----1----+----2----+----3----+----4
20112**INVALID MAC
DO     1 RE-ENTER MAC
       2 RETURN TO FUNCTION DISPLAY
20123**MAC/INS. DOES NOT MATCH GEO-ID
DO     RE-ENTER  1 MAC/INS.  2 GEO-ID
     3 EXIT AIR FACILITY SYSTEM
20212**INCORRECT DELETE PASSWORD
DO     1 RE-ENTER DELETE PASSWORD
       2 RETURN TO FUNCTION DISPLAY
10112**INVALID GEO-ID
DO     1 RE-ENTER GEO-ID
       2 EXIT AIR FACILITY SYSTEM
10123**INVALID PASSWORD
DO     1 RE-ENTER PASSWORD
       2 RE-ENTER GEO-ID AND PASSWORD
       3 EXIT AIR FACILITY SYSTEM
10212***INCORRECT RESPONSE
DO     1 RE-SELECT FUNCTION
       2 EXIT AIR FACILITY SYSTEM
40212**INCORRECT RESPONSE
DO     1 RE-SELECT RESPONSE
       2 RETURN TO FUNCTION DISPLAY
60122**INVALID CATEGORY
DO     1 RE-SELECT CATEGORY
       2 RETURN TO THE FUNCTION DISPLAY
60312**INVALID SUB-CATEGORY
DO     1 RE-SELECT SUB-CATEGORY
       2 RETURN TO THE FUNCTION DISPLAY
60112**INVALID RESPONSE
DO     1 RE-ENTER RESPONSE
       2 RETURN TO THE FUNCTION DISPLAY
70116**INCORRECT ENTRY
DO     1 RE-ENTER RESPONSE 2 RETURN
3 NEXT FIELD 4 COPY 5 NEW DATA 6 SAVE
70136**END OF PAGE
DO     1 TEXT       2 RETURN
3  COPY  4 CONTINUE  5 NEW DATA  6 SAVE
70123**FURTHER MULTIPLES
DO     1 PROCESS NEXT MULTIPLE
       2 DONT PROCESS MULTIPLE  3  COPY
```

```
20143**INVALID INSTALLATION
DO     1 RE-ENTER WAC
       2 RE-ENTER INSTALLATION
       3 RETURN
60133**INVALID SUB-CATEGORY
DO    1 RE-ENTER CATEGORY
      2 RE-ENTER SUB-CATEGORY
      3 RETURN TO FUNCTION DISPLAY
60144**INVALID ELEMENT(S)
DO    RE-ENTER: 1 CATEGORY
      2 SUB-CATEGORY
      3 ELEMENTS    4 RETURN
70145**END OF PAGE
DO     1 TEXT  2 RETURN
3 COPY  4 CONTINUE  5 NEW DATA
70153**END OF PAGE
DO     1 CONTINUE  2 RETURN
       3 COPY
70162**FILE ALREADY EXISTS
DO     1 ENTER NEW FILE NAME
       2 RETURN TO FUNCTION DISPLAY
70172**FILE DOESNT EXIST
DO     1 ENTER NEW FILE NAME
       2 RETURN
70182**INVALID SUB-CATEGORY
DO     1 RE-ENTER SUB-CATEGORY
       2 RETURN TO FUNCTION DISPLAY
70194**INCORRECT ENTRY
DO     1 RE-ENTER RESPONSE  2 RETURN
       3 NEXT FIELD         4 COPY
70212**INCORRECT START COLUMN
DO     1  RE-ENTER START COLUMN
       2  RETURN TO FUNCTION DISPLAY
70222**ELEMENTS DO NOT HAVE SORT KEY 1
DO     1 RE-ENTER ELEMENT
       2 RETURN TO FUNCTION DISPLAY
70233**LATITUDE OUT OF RANGE
DO     1 RE-ENTER LATITUDE
       2 RE-SELECT DELIMITING CRITEREA
       3 RETURN TO FUNCTION DISPLAY
70243**LONGTITUDE OUT OF RANGE
DO     1 RE-ENTER LONGTITUDE
       2 RE-SELECT DELIMITING CRITEREA
       3 RETURN TO FUNCTIONS DISPLAY
70254**MINUTES OUT OF RANGE
DO     1 RE-ENTER MINUTES
       2 RE-ENTER LAT/LONG  3 RETURN
       4 RE-SELECT DELIMITING CRITEREA
       5 RETURN TO FUNCTIONS DISPLAY
```

3.4.6 On-Line Programs

The on-line capabilities of the Air Facilities Subsystem are supported by special on-line programs, series of tables with software characteristics in function areas vital to the user, and interactive modules for the handling of basic display and input functions. The special on-line software is outlined in greater detail in this section. It provides a program framework that enables the data base management funcions of:

- o System Security with Log/on, Log/off procedures, GEO-ID/Password checks and their correlations.
- o Data Retrieval from the AAFIF data base and display on a page-by-page, group, or element level.
- o Updates with extensive format and logical checks, including automatic updates (computed inputs).
- o Adds and Deletes of the information for a whole airfield.

The programs for the on-line data base management system consist of the following modules:

3.4.6.1 AFF - Air Facilities Executive

The program AFF serves as the on-line Air Facilities executive and performs the functions of Log-on/Log-off. The interactive, on-line mode provides for a full repertoire of error checking of user inserted data as well as a complete retinue of meaningful, diagnostic error messages with a unique steering capability to allow the user to act upon these errors.

The program initially asks the user to enter a GEO-ID and password. These values serve as "keys" to enter the program which then accesses a subroutine that asks a user to select a system function. The functions include ADD, DELETE, UPDATE, DISPLAY, RETRIEVE, and LOG-OFF. It then will initate the execution of the selected function. Upon completion the user will have the option of re-selecting a function.

Input
The inputs to the AFF program are the entrance 'keys' of the GEO-ID and password as well as the selected function.

Output
The outputs from the program are the updated user's log as well as a completed function.

Tables/Files Accessed

The following tables/files are accessed by the program:

✓ Validate GEO-ID and password:

VGEO.DT

✓ Transaction Log:

TLG.DT

✓ Display File and Indices:

DISPL.DT
INCD.DT

✓ Communications File and Indices

CTL.DT
INCT.DT

✓ Password File

"QXFA"

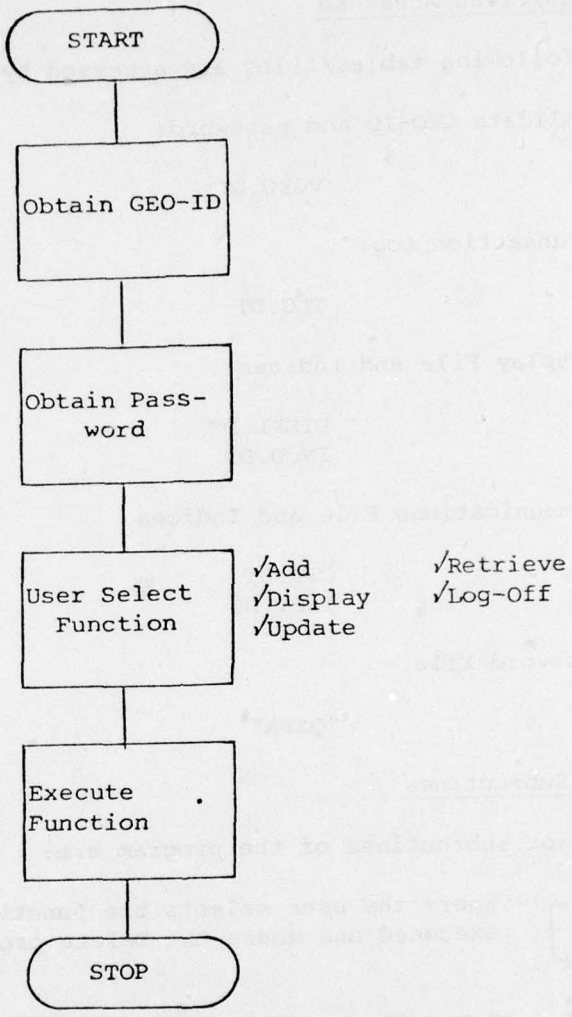
Major Subroutines

The major subroutines of the program are:

✓ FDO, DLE, DLE1 - Where the user selects the function to be executed and where the Delete programs

✓ UPD1 - Subroutines that perform the Update functions
UPD2
UPD3
UPD31
UPD32
UPD4
UTXT
UTXTS
UTXTA
CHSTRG
STRSCH
UPD5

System Diagram



3.4.6.2 DISPLAY

The Display Subroutines permit a user to display and view selected contents of an Air Facilities record. The program initially asks the user to enter the WAC and installation numbers. These parameters serve to identify the record to be displayed. The program then asks the user to select the portion of the record that he desires to display. The full record, a full category, a full subcategory of a category, specific elements, or a range of elements may be chosen. Next the user is given the choice of having the programs displayed at the terminal or the line printer or both. If the line printer option is selected, the selected portions of the record are printed in sequence at the line printer. If the terminal option is chosen the program displays all data items selected, if necessary on a page-by-page basis. Due to their character lengths, text fields are processed in separate displays. Upon completion the user is asked if he desires to continue. If so, the above process is repeated.

Input

The inputs to the program are:

- ✓ User inserted parameters that are the retrieval indexes for the record, i.e., WAC and Installation Number.
- ✓ The positional parameters that specify which portions of the record are to be displayed, i.e., the full record, a category, a subcategory of a category, specific elements, or a range of elements.
- ✓ The actual data in the record that is to be displayed.

Output

The output from the program is the display of data on the screen or a printout of the data at the line printer.

Tables/Files

The following table/files are accessed by the DISPLAY program:

✓ Air Facilities data base

✓ Label Book Tables

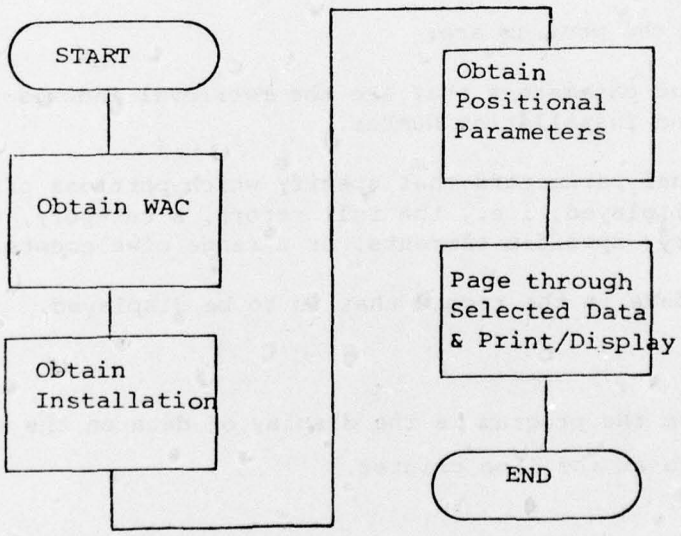
✓ WAC validation file: VWC.DT

✓ Category/Sub-Category/Element Validation File: VPP.DT

✓ Display File and Indices: DISPL.DT
INCD.DT

✓ Communication File and Indices: CTL.DT
INCT.DT

System Diagram



3.4.6.3 UPDATE

The Update subroutines permit a user to update selected contents of an Air Facilities record. The program initially asks the user to enter the WAC and installation numbers. These parameters serve to identify the record to be updated. The program then asks the user to select the portion of the record that he desires to update. The full record, a full category, a full subcategory, specific elements, or a range of elements may be chosen. The program then sequences through the selected portion of the record. At each page the present contents of the data elements are printed and the user simply inserts the desired changes. These inputs are made subject to format checks with corrective actions requested by the system when necessary.

Text fields, due to their character lengths, are processed following the regular page update. Upon completion of this process, logical checks are performed that correlate the entered, new data with other data elements of the airfield. If correct, the update process will be completed by automatically writing the new data to the data base. Otherwise, the user will be notified by the system for remedial actions. Finally, the user is asked if he desires to update another record. If so, the above described process is repeated.

Input

The inputs to the program are:

- ✓ User inserted parameters that are the retrieval indexes for the record, i.e., WAC and installation.
- ✓ The positional parameters that specify which portions of the record are to be updated, i.e., the full record, a category, a subcategory of a category, specific elements, or a range of elements.
- ✓ The actual, current data in the record that is to be updated

Output

The output of the Update program is the user inserted values that replace the old element values.

Tables/Files

The following tables/files are accessed by the Update programs:

✓ Air Facilities data base

✓ Label Book Tables

✓ Format and Logical Check Tables

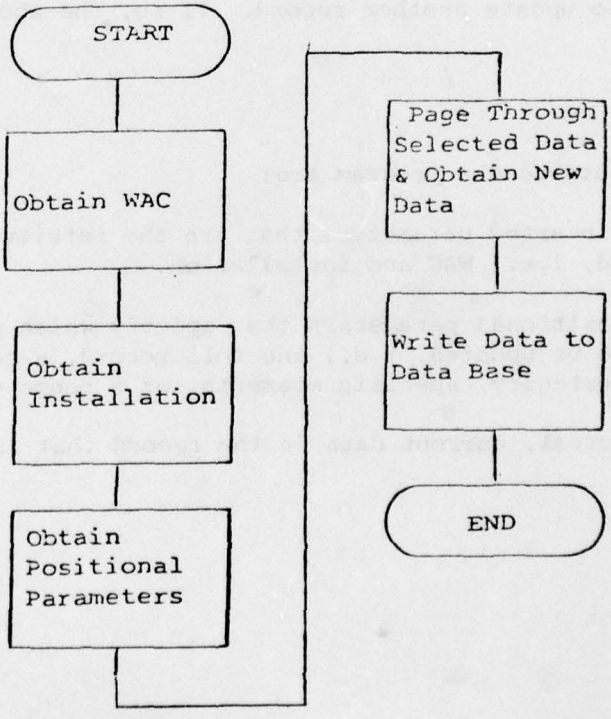
✓ WAC Validation File: VWC.DT

✓ Category/Subcategory/Element Validation File: VPP.DT

✓ Display File and Indices: DISPL.DT
INCD.DT

✓ Communication File and Indices: CTL.DT
INCT.DT

System Diagram



3.4.6.4 ADD

The Add Programs permit a user to insert or add a full Air Facilities record to the data base. The program initially asks the user to insert the record index parameters of WAC, installation number, and coordinates. The program then pages through each page of elements of a subcategory of all categories. Due to their character lengths, text fields are processed following the addition of regular page element fields. Upon completion of entering a record the program will interactively ask if the user desires to enter another one. If so, the above described process is repeated.

Input

The inputs to the program are the data inserted by the user via keyboard entry. There are two types:

- ✓ Input parameters, that are the index parameters for the record. They include WAC, installation, and geographic coordinates.
- ✓ Record data that is the field element data including text that the user inserts.

Output

The output from the program is a full Air Facilities record that may be further updated or accessed by other Air Facility capabilities.

Tables/Files

The following table/files are accessed by the ADD program:

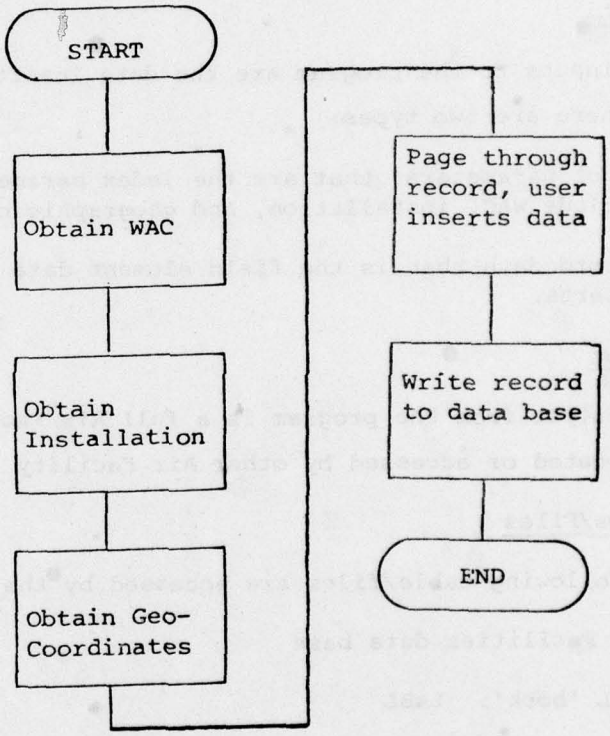
- ✓ Air Facilities data base
- ✓ LABL 'book': LABL
- ✓ WAC Validation File: VWC.DT
- ✓ Category/Subcategory/Element Validation File: UPP.DT
- ✓ Display File and Indexes: DISPL.DT
  INCD.DT

✓ Communication File and Indices: CTL.DT
INCT.DT

✓ Format and Logical Test Files: FMT.DT
LGT.DT

As with the previously discussed Update program, format checks will be conducted by the system upon completion of each data field. The logical checks will commence only after data for all elements in the airfield have been entered.

System Diagram



3.4.6.5 DELETE

The Delete Subroutines permit a user to delete an Air Facilities data record. It is performed in an interactive, on-line mode. It provides for a full repertoire of error checking of user inserted data as well as a complete retinue of meaningful diagnostic error messages and unique steering capability to allow the user to act upon these errors.

The program initially asks the user to enter WAC and installation numbers. These parameters serve to identify the record to be deleted. The program then asks the user to enter or delete password. The user is given only three chances to correctly enter the password. Then the record is deleted and the user is informed via a display of such. The user may also delete a multiple element value. He simply inserts the field identifier, and then the mulitple number.

Input

The inputs to the program are the WAC installation record identifiers and the record to be deleted.

Output

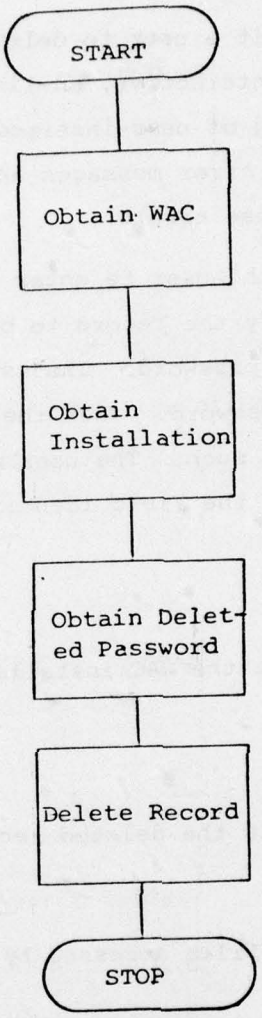
The output from the program is the deleted record.

Tables/Files Accessed

The following are the tables/files accessed by the DELETE program:

- ✓ Air Facilities data base
- ✓ WAC Validation File: VWC.DT
- ✓ DELETE Password File: QXZH
- ✓ Display File and Indices: DISPL.DT
  INCD.DT
- ✓ Communication File and Indices: CTL.DT
  INCT.DT

System Diagram



3.4.6.6 RETRIEVE

The RETRIEVE programs permit a user to retrieve Air Facility records according to user specified parameters. It provides for a full repertoire of error checking of user inserted data as well as a complete retinue of meaningful diagnostic error messages and a unique steering capability to allow the user to act upon these errors.

The parameters are initially a table that creates a Boolean query logic. These tables are then delimited to the full data base, a geographic grid area, a WAC range, country range, installation range, or list of previously retrieved records. The program then will retrieve the records and pass them to an output module. The program is entitled RTL and is called from the LOG program. It is stored in the file RTL.FR. Its binary code is stored in the file RTL.RB and the RLDR sequence is stored in the file RTL.MC. It is, of course, stored in the AF Directory.

Input

The inputs to the RETRIEVE program are the user inserted and created Boolean logic table, and search delimiters. These include WAC range, installation numbers, grid area, previous subset, country range, or full data base.

Output

The output from the program is the set of retrieved records.

Tables/Files Accessed

The following are the tables/files that the program accesses:

✓ Boolean Logic Table: R- .DT

✓ GRID Search File: GRID
RFLWC

✓ WAC Validation File: VWAC.DT
RFWC.DT

✓ Any specified, previously derived subset: - .DT

✓ The list of retrieved records: RFIL.DT

✓ Display file and indices: DISPL.DT
INCD.DT

✓ Communications File and Indices: CTL.DT
INCT.DT

Major Subroutines

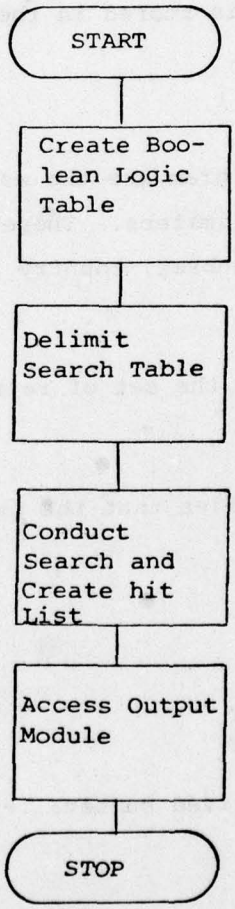
✓ RTL2 - conducts the grid search

✓ DISPL - prints the interactive displays

✓ CTL - prints the communication displays

✓ DREAD - reads in and echoes user keyboard inputs

System Diagram



3.4.6.7 Interactive Modules

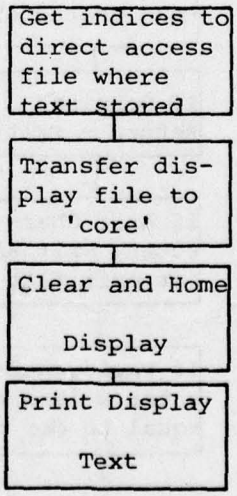
The interactive section of the Air Facility system performs the function of permitting the user to access system on-line capabilities. Basically, it consists of a sequence of displays printed at the Datagraphics 132A terminal. These displays request information from the user regarding system function selection, parameter insertion, data entry, as well as present error message display and further selection processing. Responses by the user are entered by depressing the appropriate key(s) at the keyborad.

To perform these capabilities the following software modules have been implemented:

- A Display Generator
- A Character Read/Echo Module
- A set of logical error check modules
- An error communications display generator

3.4.6.7.1 Display Generator

The Display Generator is a generalized software module that prints display text on the Datagraphics terminal. Its major capability is to be able to print any text stored in a formatted disk file, thus separating text from a source program. The macro algorithm for the module is as follows:



The display generator is accessed by a Fortran call to DISPL (N, N1, ARRAY) where:

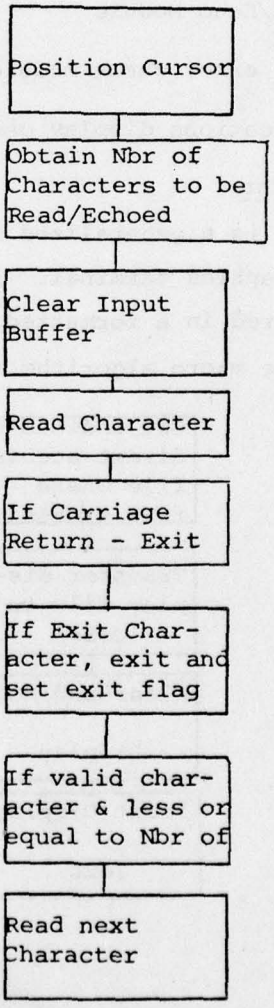
N = application area

N1 = display number in the application area

ARRAY = is the array that contains the 'core' image of the disk file text

3.4.6.7.2 Read/Echo Module

The Character Read/Echo Module will, as titled, read and echo a character(s) from the keyboard to the Datagraphics terminal. The module also provides cursor positioning, character delete, line delete, input escape, and no-echoing of excess input of characters. The macro algorithm is as follows:



(characters to be read in-
Echo it to the terminal
(if greater than maximum
allowable characters it is
not echoed).) '

The valid character set is that contained in the AF data base. Special control characters are as follows:

 - character delete

shift  - line delete

CR - exit

shift - exit and set flag

The two independent subroutines that compromise the module are:

DREAD (ARRAY, N, INBUF)

where:

ARRAY is the display text containing cursor positions and number of characters to be read in.

N = specific line of display text.

INBUF is the vector that contains inputted characters.

RDWR (NCHARS, NREAD, INBUF)

This routine does the actual reading, checking, and echoing.

NCHARS = maximum number of characters to be read in.

NREAD = actual number of characters read in.

INBUF = vector containing characters read in.

3.4.6.7.3 Error Check Routines

The logical error check subroutines determine whether the user's inserted characters are valid with respect to the specific application. Identified error checks include:

- WAC Validation
- Installation Validation
- GEO-ID/Validation Password

- Update Format Check
- Update Logic Check
- Positional Parameter Validation

These checks are implemented via the use of subroutines with the following format:

Subroutine CHECK (INBUF, IER)

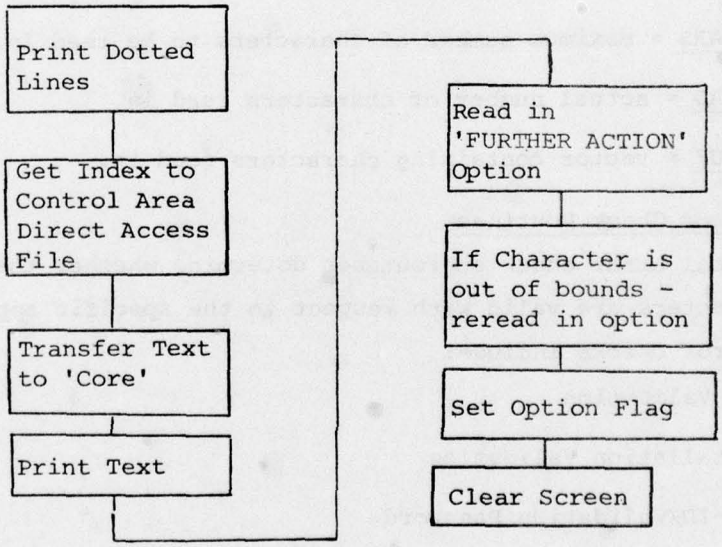
where:

CHECK = the name of the validation routine.

INBUF = the vector containing the data to be validated.

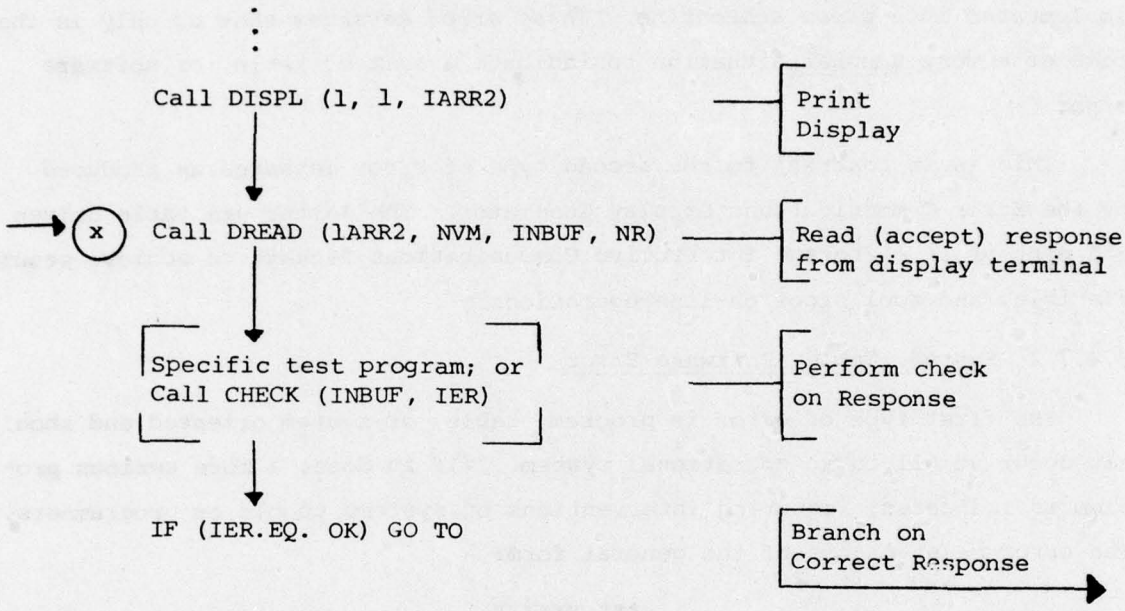
IER = an error flag signifying whether data is valid or invalid.

The Error Communications Display Generator is a generalized software module that prints error messages and 'Further Action' prompts on the Datagraphics screen. It is accessed when a user inserts an invalid response to a prompt or when he enters 'shift ■' in character input. The macro algorithm for the module is as follows:

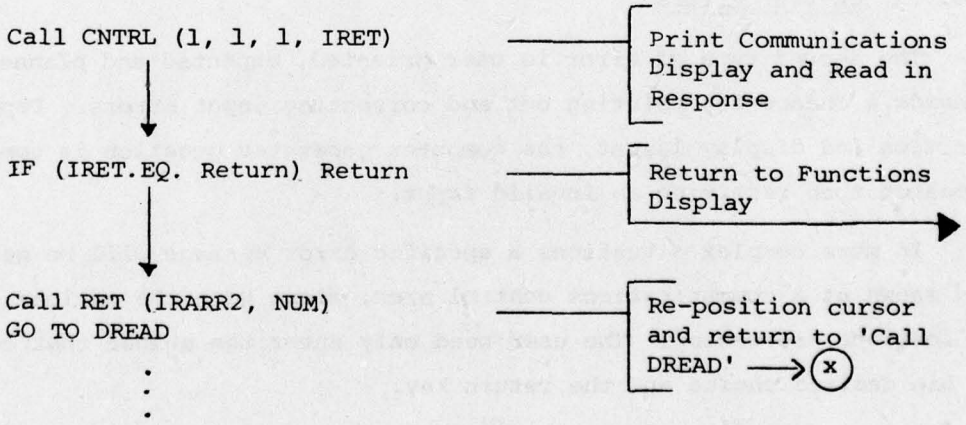


3.4.6.7.4 Application Principles of Interactive Routines

Use of the outlined modules can be outlined in simplified form by the following sample of an interactive program sequence:



In the case of error:



3.4.7 Error Messages

Two types of error messages may occur. The first is contained in the software itself and triggered by a most unusual condition tested for and implemented in a given subroutine. These error messages show up only in the case of a very unusual situation to indicate a system, table, or software error.

This is in contrast to the second type of error messages as produced by the Error Communications Display Generator. The latter are table driven and part of an elaborate Interactive Communications Package to achieve secure, flexible, and fool proof on-line operations.

3.4.7.1 System, Table, Software Error

The first type of error is program, table, or system oriented and should not occur at all in an operational system. (If it does, a more serious problem is indicated, requiring interventions of systems people or programmers.) The error messages are of the general form:

*** XXXXXX

with 'XXXXXX' standing for a specific message such as 'FORMAT TABLE MISSING.' They will be displayed on the screen starting at the current cursor position.

3.4.7.2 On-Line Errors

The second type of error is user oriented, expected and planned for to provide a chance for pointing out and correcting input errors. Depending upon function and display layout, the computer generated question is sometimes just repeated upon receiving an invalid input.

In more complex situations a specific error message will be generated and shown at a communications control area, along with the options available to continue processing. The user need only enter the number that corresponds to his desired choice and the return key.

3.4.7.3 List of Implemented On-Line Messages

**Invalid WAC

Do 1 Re-enter WAC
2 Return to function display

**WAC/INS. Does not match GEO-ID Re-enter

Do 1 WAC/INS
2 GEO-ID
3 Exit Air Facility System

**Incorrect Delete Password

Do 1 Re-enter delete password
2 Return to function display

**Invalid GEO-ID

Do 1 Re-enter GEO-ID
2 Exit Air Facility System

**Invalid Password

Do 1 Re-enter Password
2 Re-enter GEO-ID and Password
3 Exit Air Facility System

**Incorrect Response

Do 1 Re-select function
2 Exit Air Facility System

**Incorrect Response

Do 1 Re-select response
2 Return to function display

**Invalid Category

Do 1 Re-select category
2 Return to the function display

**Invalid Subcategory

Do 1 Re-select subcategory
2 Return to the function display

**Invalid Response

Do 1 Re-enter response
2 Return to the function display

```
**End of page

            Do  1  Text
                2  Return
                3  Copy
                4  Continue
                5  Save

**Further Multiples

            Do  1  Process next multiple
                2  Don't process multiple
                3  Copy


**Invalid Installation

            Do  1  Re-enter WAC
                2  Re-enter Installation
                3  Return


**Invalid Sub-Category

            Do  1  Re-enter Category
                2  Re-enter Sub-Category
                3  Return to function display


**Invalid Element(s)  Re-enter

            Do  1  Category
                2  Sub-category
                3  Elements
                4  Return


**End of Page

            Do  1  Text
                2  Return
                3  Copy
                4  Continue
                5  New Data


**End of Page

            Do  1  Continue
                2  Return
                3  Copy
```

**File Already Exists

Do 1 Enter a new file name
2 Return

**File Doesn't Exist

Do 1 Enter new file name
2 Return

**Invalid Element

Do 1 Re-enter Element
2 Return

**Incorrect Entry

Do 1 Re-enter response
2 Return
3 Copy

**Incorrect Start Column

Do 1 Re-enter start column
2 Return to function display

**Elements Do Not Have Sort Key 1

Do 1 Re-enter element
2 Return to function display

**Latitude Out of Range

Do 1 Re-enter latitude
2 Re-select delimiting criteria
3 Return to function display

**Longtitude Out of Range

Do 1 Re-enter longtitude
2 Re-select delimiting criteria
3 Return to functions display

**Minutes Out of Range

Do 1 Re-enter minutes
2 Re-enter Lat/Long
3 Return
4 Re-select delimiting criteria
5 Return to functions display

**Incorrect Direction

Do 1 Re-enter direction
2 Lat/long
3 Minutes
4 New criteria
5 Return

**GRID Finished

Do 1 Further grids
2 New delimiting criteria
3 Search
4 Return

3.4.8 Application Programs

The Application Programs usually involve the processing of larger sections of the data base. For that reason a 'wholesale' off-line processing during the off hours (nights, weekends, holidays) is indicated. Nevertheless, the on-line system is frequently used to initiate and to simplify those off-line procedures (e.g. the on-line creation of Boolean search tables for the off-line processing of large SAIR requests).

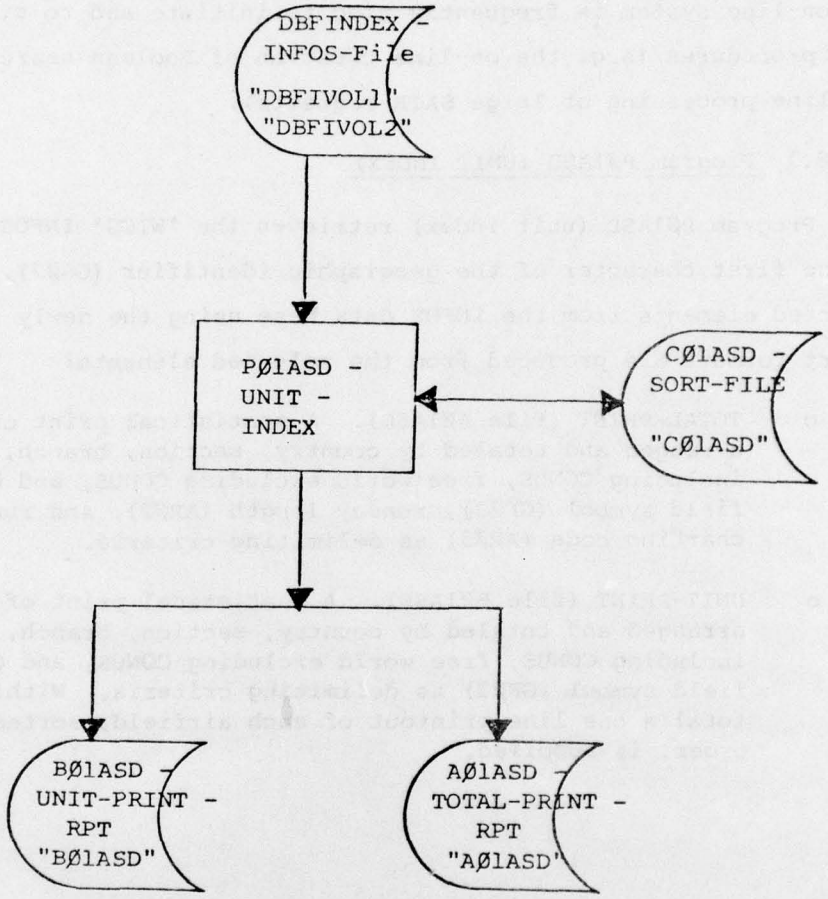
3.4.8.1 Program PØ1ASD (UNIT INDEX)

Program PØ1ASD (unit index) retrieves the 'WICG' INFOS index, sorts it by the first character of the geographic identifier (GGØ7), and retrieves selected elements from the INFOS data base using the newly sorted index. Two report formats are produced from the selected elements:

- TOTAL-PRINT (file AØ1ASD). A statistical print of all airfields arranged and totaled by country, section, branch, free world including CONUS, free world excluding CONUS, and CONUS using airfield symbol (GFØ3), runway length (ARØ2), and runway surface charting code (ARØ5) as delimiting criteria.

- UNIT-PRINT (file BØ1ASD). A statistical print of all airfields arranged and totaled by country, section, branch, free world including CONUS, free world excluding CONUS, and CONUS using airfield symbol (GFØ3) as delimiting criteria. Within each country total a one line printout of each airfield, sorted in B.E. Number order, is supplied.

SYSTEM FLOWCHART



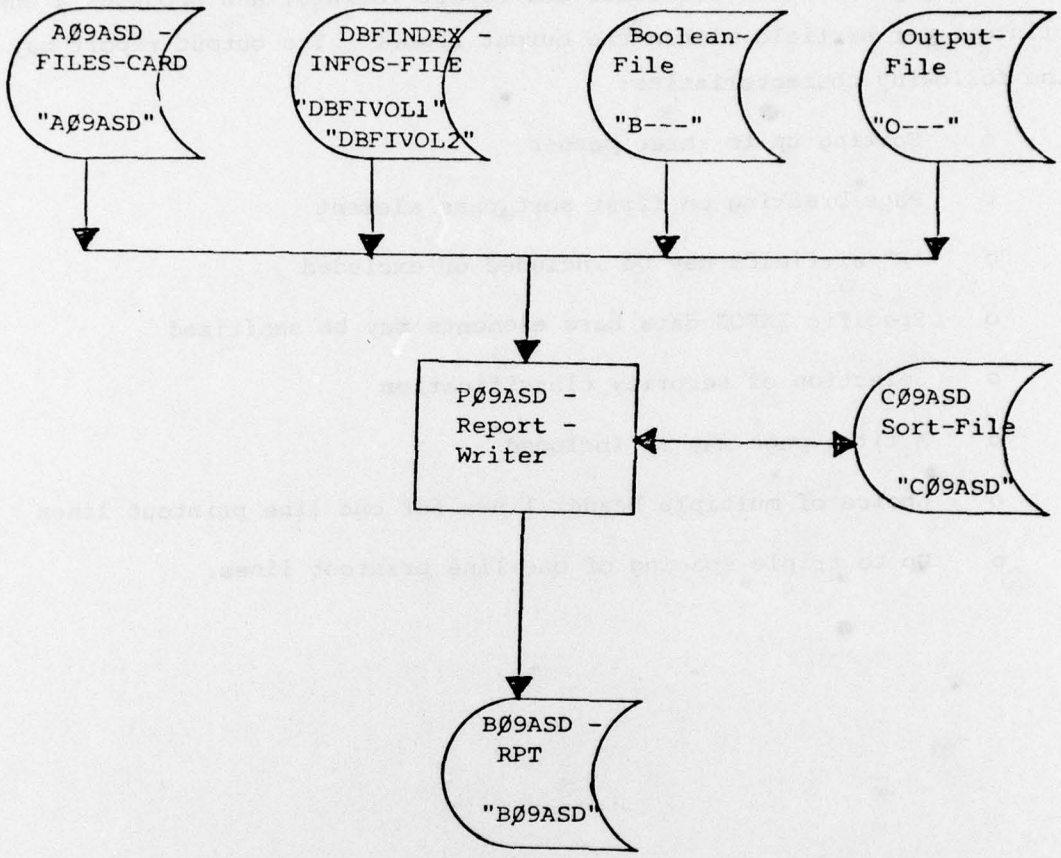
Program PØ1ASD

3.4.8.2 Program PØ9ASD (SUBSAR)

Program PØ9ASD (SUBSAR) retrieves selected elements from the INFOS data base using pre-defined airfields and report formats, and produces a one line printout per airfield within the output report. The output report may have the following characteristics:

- Sorting up to three passes
- Page breaking on first sort pass element
- "R" airfields may be included or excluded
- Specific INFOS data base elements may be sanitized
- Selection of security classification
- A title page may be included
- Choice of multiple header lines for one line printout lines
- Up to triple spacing of one line printout lines.

SYSTEM FLOWCHART



Program PØ9ASD

3.4.8.3 NAME CHANGE

This program lists changes that have occurred against selected elements in the AAFIF data base on a weekly and semiannual basis. Two report formats are output and sorted according to user preference: 'NAMCHG' and 'CHGSLIP.' The report outputs are unclassified and used in the maintenance of certain FLIP products, NAV and planning charts, and DoD air facility library records.

3.4.8.4 HISTDUP

This program produces a card code formatted dump to magnetic tape of all airfields in the AAFIF data base or uses the Boolean retrieval file as input to dump selected airfields.

3.4.8.5 AIFUPD

This program creates the Defense Intelligence Agency Automated Intelligence file (DIA-AIF) that reflects air facility intelligence changes, additions, and deletions for a select set of DIA AAFIF data base elements. Input to this program is the Air Facility Subsystem transaction log and output is a magnetic tape reflecting the necessary transactions.

3.4.8.6 ASSOTW

This program builds the Air and Sea Plane Stations of the World (ASSOTW) magnetic tape file by retrieving and formatting airfield data from the AAFIF data base using the ASSOTW transaction log as input. The ASSOTW magnetic tape file is then processed on the Electron Beam Recorder (EBR).

3.4.9 Operating Instructions

This section describes the operator interface with the system for operational as well as system loading and test procedures.

- To list a program use the CLI command:

  LIST **.FR

  Where ** is the file name.

- To print a program at the terminal use the following CLI command:

  TYPE **.FR

- To print a program at the line printer use the following CLI command:

  PRINT **.FR

- To compile a program use the following CLI command:

  FORT **.FR

3.4.9.1 Operational Programs

- To run on-line a set of programs execute the following 'MC' files.

  AFS.MC On-Line Data Management Programs

  RET.MC Edit Table Programs

3.4.9.2 Directory Selection

The preceding programs represent the 'operational' version of the developed software which has been collected and assembled in one directory (DIR PETE). There also exists a 'developmental' and 'debug' version in another directory (DIR NEHL) which has many of the built in software tests and error messages left activated. The latter also contains the programs for table generations, testing, and systems loading.

To select the desired directory use the following CLI commands:

DIR PETE (Operational Programs)
DIR NEHL (Test and System Loading Programs)

3.4.9.3 Table Generation and Loading Programs

To produce header and tab settings for the on-screen generation of label book tables type:

LDISPLAY

To produce a hardcopy printout of all label book tables use the MC file:

PLABL

To process all label pages and to load the created load file to INFOS type (MC file):

CLBL

To produce header and tab settings for the on-screen generation of Format tables and Logical tables use programs:

FHEAD (Format tables)

LHEAD (Logical tables)

For 'wholesale' inputs of edit tables via the cardreader use program:

CEDT

For the on-line testing and modification of a specific edit table (with automatic, subsequent reloading to INFOS) use the program package:

ET

For a documentation of the test session through subsequent hardcopy output use the CLI commands:

PRINT SNAME (hardcopy)

TYPE SNAME (screen review)

(Substitute NAME with the actual name of the session file as requested by ET and input during the test session.)

For the 'wholesale' loading to INFOS of all current edit tables use:

EPRT

This program also produces two RDOS files:

- ETLIST, a comprehensive listing of all loaded and missing tables (edit table system summary), and
- EREPRT, a listing of all formal errors detected during the loading cycle. An on-line generated output option allows the user to expand this listing to include the contents of each loaded table in its original RDOS form.

To obtain a hardcopy output of those created load files use:

PRINT ETLIST (edit table loading summary)

PRINT EREPRT (detected errors and table contents)

3.4.9.4 Function Analysis of Logical Tables

In DIR NEHL a special version of the AFF program exists that contains detailed error and diagnostic messages for the UPDATE function. It also provides a complete hardcopy output of the Logical check procedures at the end of the Update cycle. Since all tables, their load segments, and the evaluation results are printed, a complete analysis of the many (otherwise unseen) functions can be made.

The system is very protective against 'wrong' inputs, to the point that it might not be possible to update certain portions of an airfield. The provided capability is useful therefore in specifically answering the question which of the following items have caused an update rejection:

- o unacceptable (though format-correct) user input
- o 'old' data base entry that does not conform with a 'new' logical edit requirement
- o wrong logical table entry

Before large-scale operations commence, it is anticipated and recommended that a thorough break-in period be used to vigorously exercise the system in order to detect and to resolve most of these potential differences. To initiate the provided analysis capability use the program:

AFL (for UPDATE with printout of Logical edit checks)

SECTION IV

CONCLUSIONS AND RECOMMENDATIONS

4.0 General

Main purpose of the pilot effort was to demonstrate the feasibility of the design on a limited 1/15 data base volume. The conducted functional and volume tests thus constituted a major evaluation criteria for determining whether to continue Phase II of the program and, in the case of program continuation, to pinpoint potential areas of improvement.

4.1 Conclusions

The tests, as conducted during the Air Facilities' test and evaluation period, conclusively demonstrated that the functions designed and implemented match the requirements outlined in the Statement of Work.

Three types of tests were performed. The inspection tests proceeded as conceived and were accepted as is. They dealt primarily with vendor supplied equipment and systems software. The function tests dealt with the required processing functions as designed by Synectics and implemented through specific software. These tests proceeded as conceived and were accepted as each function test was demonstrated, with exceptions and suggestions for modifications noted. The major conclusion from the volume tests was that the new on-line system, in a 2-week period using one analyst, successfully accomplished the work of the 100-150 analysts who would generate the same volume in a 2-week period with the old, card-oriented processing system.

The prime conclusion that can be drawn from the Air Facilities' test and evaluation exercise is that the functions, as implemented in the pilot system, in general performed as proposed, or better, over a 1/15 data base and transaction volume.

4.2 Recommendations

It is recommended that the functions, as demonstrated during the Air Facilities Pilot system be extended and implemented to cover the full Air

Facilities data base and processing volume. Major emphasis in this Phase II effort should be placed on an extended data base structure with automatic loading support, improved processing capabilities for application programs that produce (off-line) reports and tapes, and a full expansion to remote, multi-user access for the on-line data base maintenance system.