SURVEY OF MANAGEMENT INFORMATION SYSTEMS AND THEIR LANGUAGES

MAY 1968

MTP-313

J. FRY

J. GOSDEN

DATA MANAGEMENT SERIES NO. 1

THE MITRE CORPORATION

32

MTP-313

SURVEY OF MANAGEMENT INFORMATION SYSTEMS AND THEIR LANGUAGES

J. FRY
J. GOSDEN

MAY 1968

THE MITRE

The work reported in this document was sponsored by the Defense Communications Agency under Contract AF 19(628)5165

ABSTRACT

Significant data management systems available on third generation hardware are considered. Three types of user interface are discussed: Own DML (Data Management Language) Systems, Forms Controlled Systems and POL Embedded Systems. Typical systems within each category are presented and their salient features are highlighted.

TABLE OF CONTENTS

1.0 INTRODUCTION

In this short paper it is not possible to do justice to the variety of data management systems that are available or imminent. Rather, this paper presents an overview of several major data management systems selected as typical in three categories of user languages. A survey of the detailed characteristics (languages, data structures, and facilities) of some dozen systems is presented in another paper.[1] That paper is a formal, organized, and consistent tabular restatement of the claimed characteristics of the data management systems and does not represent an evaluation.

Other writers have discussed the characteristics and properties of other systems but we particularly recommend the articles by Canning[2, 3] as fine descriptive reports on current systems. We are also assuming that our audience is generally familiar with data management systems and the important data structure concepts being developed. For a general reference manual offering extensive bibliographies and first-class reviews rather than opinions, we recommend the American Documentation Institute's Annual Reviews.[4]

It is interesting that the major thrust of the development of data management systems has not come from the hardware manufacturers, but rather from military command and control systems and user groups. Indeed the trend has continued so that now the software houses are making major contributions in this area, and we have selected most of our examples from that source.

Data management systems are really an outgrowth from RPGs (report program generators). Thus one of the early ancestors was the set of report generators produced for the IBM 702. The first of these, the MARK I Generalized Report Generator, was completed about 1956 and the second, MARK II Generalized

Report and File Maintenance Generator was a 1957 vintage system. This was followed later by 9PAC on the IBM 709 and 7090 in 1958/59, while at about the same time the TUFF-TUG-IRS implementation on the 704 began the series of military data management systems (sometimes called command and control systems) that have led to FFS and NIPS which in turn seem to be the immediate ancestors of GIS (Generalized Information System) being developed by IBM. Meanwhile, 9PAC has been the main ancestor for most nonmilitary implementations.

2.0 MAJOR CLASSES OF LANGUAGE

In general, the scope of capabilities provided in major data management systems does not vary very much. The major differences lie in the form of the user languages, which are the user interface to the system. Such languages cover several major functional areas:

describing data
input
update
reorganization
retrieval
sorting
presentation

The general user is most concerned with queries to a data base which utilize the last three areas. In many cases, sorting and presentation are automatic or separate functions and the major language differences are typified by the languages used to define the retrieval required.

Retrieval languages divide into three major categories:

Own DML (data management language)
Forms Controlled
POL Embedded

The three categories can be briefly contrasted as follows: Own DML Systems are imitative of conventional POL's; Forms Controlled Systems use well tried "forms" techniques common in everyday business and public life ; and POL Embedded Systems take advantage of existing powerful programming languages.

3.0 OWN DML SYSTEMS

These systems are imitative of procedure-oriented languages (POLs). They used the style and approach of POLs but used a new format and verbs specifically developed for data managemert. Just as FORTRAN was oriented toward "scientific" computation and COBOL toward "business" needs, so DMLs were oriented toward "data management." They followed the example of COBOL in separating the input for various functional areas, e.g., data description, retrieval conditions, and output format. Indeed they developed separate languages for each functional area taking advantage of the fact that different kinds of users are involved. For example, data description languages for systems specialists to establish the file structure, maintenance languages for clerks who nurture the data base, and query languages for casual users. They also simplified the programming skills needed, first by restricting themselves to basic functions such as sum, count, and average, and second, by eliminating many housekeeping chores, especially input/output control, and also the specific counting through records repeated sets, and individual fields. In fact, some parts of DMLs, especially query languages, are problem oriented languages, a specialised type of POL. They are used to specify what is to be done, not how it is to be done. In particular, most query languages are "content" oriented. They provide "content retrieval"; e.g., what is the division of the man whose name is G. Parker? The NIPS and TDMS systems described below are typical members of this category.

3.1 National Military Command System/NIPS

3.1.1 Background

The NMCS Information Processing System (NIPS/360) is in essence the Formatted File System, FFS, converted to the IBM 360/50 computer system and operational under Operating System 360/Option 2. Historically, the Formatted File Systems began with the SAC 438L system on the IBM 7090 in 1961. The NIPS predecessor on the IBM 1410 was created in 1963 to satisfy the mission of a Naval Fleet Intelligence Center. Since then, the system has been employed at several intelligence centers and commands throughout the world. Basically, the system has evolved in two separate communities : Command and Control; and Intelligence. The Intelligence system, the IDHS FFS, has emphasized efficient processing capabilities and improved output options for their many-volumed data sources. On the other hand, the Command and Control system, NIPS, has expanded logical file maintenance, improved the query language, and built an on-line retrieval mechanism. Both the second-generation FFS systems operate under the modified 1410 operating system OPSYS (1410-PR-155).

The 1410 FFS is being converted to an S/360 model 50H (eight magnetic tape drives, two 2311 discs, one 2304 disc, 2260/1053, and associated card/read/punch printer equipment) in two primary phases. Phase I components include file structuring, file maintenance, retrieval and sort, output, remote inquiry processing, system formatted output, and necessary utilities. Included in Phase II are expanded file maintenance including multi-file capabilities, file revision, on-line update, as well as additional components and utilities. The target date for the operational capabilities is January 1968, for Phase I and June 1968, for Phase II.

3.1.2 Summary of Capabilities

NIPS/360 is a general-purpose file handling system operating under O/S 360 and performing the traditional functions of structuring, maintaining, revising, and retrieving from a set of data files. While maintaining a high degree of external compatibility with its 1410 predecessor, the internal processing methods have been modified to exploit the capabilities of the third-generation hardware S/360, and operating system O/S 360. Adhering to the strict O/S 360 programming conventions for communication, base register usage, linkage, etc., the conversion is being done using the COBOL language down to the subroutine level of software. At the subroutine level, either COBOL or assembly language will be used depending on which is necessary to efficiently utilize O/S 360 software and S/360 hardware capabilities.

Compatibility with the existing 1410 FFS system was a major design criterion which is being achieved through the ability to use existing FFS control cards, query and summary decks, file maintenance decks, and analyst's procedures.

3.1.3 Specific Features

The new system does much to alleviate the physical restrictions of the old system. Specifically, the physical size restriction of a logical record to 2701 characters has been relieved by making each individual group instance a physical record. However, because of compatibility requirements, some of the logical shortcomings are still apparent in the system.

In the new system, each File Format Table (FFT is the computer form of the data description or dictionary) will be stored with its file. Thus the NIPS files will be "self-describing" and no longer have a physically remote file description which can easily be misplaced or lost. This should be a major factor in making the transfer of data possible among various installations and is an important development. The S/360 NIPS is primarily a direct access oriented system utilizing the capabilities of the disc through the Index Sequential Access Method (ISAM) of O/S 360.

The query language for NIPS, called RASP (Retrieval and Sort Processor) is a good example of its kind. Figure 1 shows the rules for the IF clause which illustrates the POL-like structure and the wide variety of alternatives possible.

3.2 System Development Corporation/TDMS

3.2.1 Background

The Time-Shared Data Management System (TDMS) is a general-purpose system for managing data in a time-sharing environment. It is currently being developed by the Technology Directorate of the System Development Corporation (SDC) as a result of work sponsored in part by the Advanced Research Projects Agency (ARPA) of the Department of Defense. TDMS is being designed and implemented for use on the IBM System 360 Model 50 TDMS is an outgrowth of TSS-Lucid System also developed by SDC for the AN/FSQ-32 computer.

3.2.2 Summary of Capabilities

TDMS will permit the user to describe and generate a file as well as retrieve and display data from the fiie on a cathode-ray tube (CRT) device. It also will provide the capabilities to update and maintain the file and to generate hard copy reports. TDMS operates under the SDC provided ADEPT operating system. ADEPT

IF	FIELD	EQUALS		VALUE	
CONNECTOR	**FIELD**	**OPERATOR**		**DATA BASE VALUE**	
IF	FFT	EQ	EQUAL	$	ANY VALUE
AND	OR GROUP	EQA	EQUAL α	--	BLANK
OR	NAME	EQN	EQUAL N	+ FIELD	FILE FIELD NAME
		LT	LESS THAN	VALUE	VALUE
		LTA	LESS THAN α	==VALUE #	CANCELS SUB
		LTN	LESS THAN N		CONVERSION

FIGURE I
NIPS "IF" STATEMENT

provides a "time-sharing" environment which we define as an environment having "on-line access" and "multiple access" and "rapid response."

3.2.3 Specific Features

To achieve rapid response in any large data base it is necessary to be able to provide an access mechanism based on content. Inverted files or indexes and associative memories are two techniques used to do this. TDMS uses a type of inverted file.

In TDMS a file is stored in a series of associated tables. One group of tables holds the actual data element values. There is a separate table for each defined data element, and only unique data element values are stored in these tables (i.e., no duplicate values). Another table contains the names of the data elements in the files. A third type of table describes the individual data elements and their logical relationships. It also points to tables which contain pointers arranged in sorted order, which point to the tables containing the actual data element values.

This system provides a variety of on-line services oriented to users who are not programmers.

There are two capabilities in TDMS which can be used to retrieve data. One of these, QUERY, can produce only relatively simple outputs. A typical example is "PRINT JOB TITLE WHERE POSITION EQUALS PROJECT HEAD." The second, COMPOSE/PRODUCE, can produce rather sophisticated outputs. Within COMPOSE, the user describes any number of report formats to TDMS. Each report format has a name and requires several statements to describe the data which is to be output and the way the output will look when it has been produced as a report. PRODUCE provides the user with the capability of requesting any of the report formats previously generated in COMPOSE to produce actual reports.

Through two additional programs, TDMS has capabilities for modifying data element values and for maintaining files. The UPDATE program allows the ucer to add, delete or change data element values. Th. MAINTAIN program provides for merging, subsetting, extracting, ordering, and restructuring of files.

4.0 FORMS CONTROLLED SYSTEMS

These systems are dominated by a design goal to provide a "simple to use" interface. Well-designed forms can contain self-explanatory user instructions, reduce the human input, and thus reduce errors in language use and input to the system. This is possible because data management systems do not need all of the facilities of a POL. As the variety of facilities in such systems has increased, so the number of forms needed has increased. The COGENT and MARK IV systems described below are typical members of this category.

It is interesting to note that there is a very strong relation between forms controlled languages and interactive on-line query languages. For every "box" on a form there is a pre-printed question or instruction to the user and the user enters his "response." Sometimes the response is made easy by a "multiple choice" format. Most interactive query languages use the same technique presenting one "box" at a time to the user who enters his response. In on-line mode the housekeeping actions such as "If you answer yes to question 7, ignore questions 8 and 9 and go to question 10" is automated. Thus simple forms controlled systems should be easy to convert to on-line.

The two systems described below also typify the advent of software houses providing "off-the-shelf" generalized software packages. In order to appeal to a wide market, they have extended forms control to include "own-code, " usually in a conventional POL, and have made provisions to accept files and data descriptions directly from other POLs. Thus we begin to see the merging of all three types of retrieval language that we identified earlier: Own DML, Forms Controlled, and POL Embedded.

4.1 Computer Sciences Corporation/COGENT III

4.1.1 Background

COGENT III, a COBOL compatible generalized file management system, is an advanced general-purpose data management system developed by Computer Sciences Corporation (CSC) for the IBM/360 family of equipment. COGENT III represents the next development in a series of generalized file management systems produced by CSC. It follows COGENT II, also developed for the IBM/360 family, and a series of COGENT systems developed for the IBM 7090, UNIVAC 1107/1108, IBM 7044, and RCA SPECTRA 70.

COGENT III is modular in construction and is designed to function on a wide range of System/360 configurations. It provides for use of unit record, magnetic tape, and direct access devices, as well as teleprocessing equipment. The system employs and is compatible with O/S 360 and the related COBOL compiler.

4.1.2 Summary of Capabilities

COGENT III generalizes the major data processing functions associated with information storage, maintenance, retrieval, and presentation. The system allows for use of a common data base by various groups of users. This facility is provided under the control of a single, comprehensive data directory which describes all data used or referenced by the system, as well as describing the relationships between data sets, records, and data fields in the data base.

COGENT III consists of an information system language and a language processor (implemented in COBOL) which interprets this language and generates functional COBOL programs for maximum machine independence to perform the information system tasks requested. In addition, an interactive inquiry retrieval and storage language is provided to meet rapid response requirements. This language is processed by the COGENT III Interpretive Processor which is implemented in assembly language employing re-entrant code. The system provides for specification and generation of program(s) for a simple function, an entire data processing application, and/or a series of integrated applications in addition to interpretive processing of dynamic requests for information storage and retrieval including on-line data input.

4.1.3 Specific Capabilities

The information system language consists of fixed-form data descriptions and functional specifications as well as an interactive inquiry and storage language. The language elements are designed to allow specification of the information needs of information collection, storage, maintenance, retrieval, and presentation for various levels of users, such as management, operating personnel, and system analysts.

In general the language allows the user to specify the following:

(1) information to be stored in the data base

(2) source of the information

(3) how and under what conditions the information is created and maintained

(4) what relationships exist between different collections of information (hierarchical structure and integration between data sets)

(5) who can have access to modify or retrieve the information

(6) how and under what conditions the information is to be retrieved

(7) how the information is to be presented

(8) on-line, off-line, batched, real-time or demand processing of tasks

The Data Descriptions allow the user to describe such items as data fields, relationships between data fields, security codes for maintenance and retrieval, decoding and encoding association tables, physical characteristics of data elements, and presentation format of data elements, which fields may be used as access keys for content retrieval, and which fields are record identifiers.

The Functional Specifications allow the user to specify:

(1) the task control information for creation, maintenance, sorting, and reporting

(2) the processing mode - direct, demand, or batch

(3) the file structure - serial or direct

The interactive inquiry retrieval and storage language allows the user to access the data base via "ad hoc" queries and execution of previously defined queries. The user can obtain immediate response or can batch quer'es.

Hierarchical data structures are provided on both serial, direct access data sets as well as between multiple data sets. In most cases, hierarchical records physically follow their associated higher level records. In the case of multiple data sets, different levels of hierarchical records may be physically stored on different devices (direct access only).

When stored on direct access devices, the record keys as defined by the Data Description language are employed to build a record key index. Tł's is one method of content access used to provide rapid response. Records are indexed by hierarchical group only. Individual records within the hiera:chy are extracted in core from physical blocks containing a hierarchical group.

An Access Key index is automatically constructed for each data field defined as an access field in the Data Description language. The index contains an entry for each possible field value with pointers (record key) to all physical blocks which contain the value. This is a selective form of content access.

4.2 Informatics/MARK IV

4.2.1 Background

The MARK IV file management system is an advanced general-purpose software system developed for the IBM/360 series of equipment. It is the fifth in a series of data management systems designed and developed by Informatics. MARK IV operates under either the Standard Disc Operating System (DOS) or Operating System (OS). The initial system will be available during the latter part of 1967 and a final version will be available early in 1968.

4.2.2 Summary of Capabilities

MARK IV is a data processing system designed primarily for business or file applications. The system provides for the generation, maintenance, and retrieval of information from tape or disc oriented files. A comprehensive report facility is also provided.

The system generates computer programs based upon specifications given to it by the user. The user employs one or more structured forms to prepare the specifications which become the MARK IV source input. No on-line facilities are now provided by the system.

4.2.3 Specific Features

MARK IV generates a program to perform the user-specified functions from precoded routines which are stored in the MARK IV library. The precoded routines provide for many of the common data processing tasks, thus reducing the amount of code that has to be generated.

The user specifies the logical organization of his files and the data are stored on tapes or on disc. The data are accessed via the Sequential Access Method (SAM) if the storage medium is tapes, and by the SAM or the Index Sequential Access Method (ISAM) if the storage medium is disc.

The system can create files from various data sources; maintain files by performing changes, additions, and deletions; select data records based on selected criteria; make computations on the data in these records; extract data items from selected records; and produce new files, parts of files, and combinations of files. Various report formats and data arrangements, as well as the facility to use preprinted forms, are available.

In addition to the standard set of programs provided by MARK IV, the user also is provided with the facility for calling up user-coded routines, "own-code," which may be required for special processing. The system also has the facility for saving a generated program in its library and calling upon the program if the user results are required on a periodic basis.

5.0 POL EMBEDDED SYSTEMS

These systems, instead of inventing new languages, have utilized and extended existing POLs, mainly COBOL and recently PL/1. Therefore, they provide all the power of a POL without the need for special linkages. The IDS, ICS (an expansion of DL-1), and DM-1 systems described below are typical but surprisingly different members of this category.

5.1 General Electric IDS

5.1.1 Background

Integrated Data Store (IDS) is a data management system designed and developed by the General Electric Company in 1965. It has been implemented in the GE200, 400, and 600 series of computers.

5.1.2 Summary of Capabilities

IDS, in concert with COBOL, provides the capability for generating, maintaining, and retrieving data from disc resident files. IDS operates under the control of the GE multiprogramming monitor which provides an on-line remote terminal inquiry capability.

5.1.3 Specific Features

IDS is embedded in COBOL. All the features available in COBOL are available to the IDS user. Some of the COBOL procedures statements have been modified, and other statements have been added to the basic language in order to accommodate IDS structured files. All maintenance, retrieval, and outputting is accomplished by using COBOL. Theoretically, a distinct set of procedures, a unique COBOL program, might be required for each type of update or query.

IDS provides the facility for placing data in the file and retrieving from it. The functions performed by IDS itself can be likened to that of special input and output subroutines.

A variety of physical and logical file structures are possible under IDS. The user has the traditional capability of specifying the logical structure of his data; he also has some control over the physical structure. IDS provides the capability of linking together any combinations of master or repeating groups to form an entry. During file maintenance, linkages are changed to reflect any additions or deletions to the file. Neither groups nor entries need to be physically stored sequentially. They may be stored in a random order, since the links connect all the groups constituting an entry and an entry is randomized on its key element(s).

IDS pioneered the use of the notion of "sets" typified by list-processing and ring structures. This is a powerful data structure concept, hierarchies can be represented in it but it cannot be represented in physical hierarchical structures. This kind of power is becoming more important as we begin to implement "integrated" data bases and was essential to the effective development of on-line graphics as exemplified in Sketchpad.[5]

IDS on the GE400 and 600 series of computers operates under the multi-programming monitor and functions like any other processor (e. g., FORTRAN) in the environment.

There are other basic differences in the versions of IDS available on the GE200, 400, and 600 series. The 200 series does not use COBOL, but an assembly language. The differences between the 400 and 600 series are less pronounced, but they do exist primarily in the COBOL language modifications and additions.

5.2 North American Aviation/DL-1

5.2.1 Background

Data Language-1 (DL-1) is a data management system developed jointly by the Space Division of North American Aviation, Inc. (NAA) and International Business Machines Corporation (IBM) in 1966. The DL-1 effort is a continuing joint effort by IBM and NAA based on their previous experience with a system called DATE, which was implemented on the IBM 7010 computer. The first version of DL-1 will contain all the features of 7010 DATE system. DL-1 is written for the IBM 360 family of computers and requires Operating System/360 and Data Management to operate. The system also uses COBOL and PL/I as programming languages.

An on-line facility, called Information Control System (ICS) is also being planned. This system will interface with DL-1 and permit simultaneous execution of teleprocessing and conventional batch processing programs. ICS will include a routine to handle 60 remote terminals and 50 different types of messages. The system will operate under the OS/360 Interim Sequential Partitioned System, and is scheduled for implementation during the early part of 1968.

5.2.2 Summary of Capabilities

DL-1 provides a user with the means to organize and describe a hierarchically structured data base. The system creates a set of directories and data description tables that are required when the data base is initially loaded into the system, and when the data is retrieved and manipulated during the execution of the user's applications programs. Data base definition is external to the applications program. DL-1 also provides the means for applications programs to interface with the system in order to create, maintain and retrieve data from the data base. The system also allows for more than one applications program to access the same data base, and allows logical records to span one or more physical disc tracks if necessary.

Storage media for the data base can be either tapes or discs. The data in tape files are accessed by the Sequential Access Method (SAM) while disc files are accessed via the Index Sequential Access Method (ISAM).

5.2.3 Specific Features

DL-1 is an interface between problem oriented compilers (COBOL and PL/I) and the IBM operating system OS/360. The primary point of contact of DL-1 with OS/360 is the Data Management section of OS/360, which is an Input/Output system that provides such data handling services as buffering and storage device control.

The system does not have its own retrieval languages. Each user of the data base is required to prepare an applications program in COBOL or PL/I in order to manipulate the data base. The applications programs contain interface statements to the PL/I system, and a communications table called a Program Communication Block (PCB). The interface statements consist of COBOL or PL/I "calls" to DL-1 followed by a series of parameters which indicate the type of operation on the data base that is desired. The actual operation on the data base will be performed by the DL-1 system.

An output formatting program is not available in the first version of the system, but is being planned for future versions. Theoretically, this facility could be provided by a COBOL or PL/I applications program.

This system also incorporates strong separate terminal and data file protection.

5.3 Auerbach/DM-1

5.3.1 Background

Data Manager-1 (DM-1) is a generalized data management software system designed by Auerbach Corporation as a proprietary package. The system is being implemented on two distinctly different hardware systems: on a Univac 1218 (Mil Spec 418) for the U. S. Air Force under contract through the Rome Air Development Center; and on the IBM 360/50 for the Western Electric Corporation.

5.3.2 Summary of Capabilities

The DM-1 system provides program and job library services, data storage and access services, and job execution control. The job library contains a number of general-purpose system jobs for building, maintaining, and querying the data base and job library. User jobs can be added to this library. The structure of the data pool provides logically for definition of items and defines a structure for a set of directory tables.

Basically DM-1 provides a nucleus system with emphasis on data definition, data structuring, and data access methods. Features such as generalized report generators, user query and maintena[illegible]ages, and recording services are not defined in the DM-1 design. It is intended that these items be developed for each customer as separate units to be integrated into the system and currently COBOL can be used as a retrieval language. Programs (generalized report generators) would be added to the job library with no unique status. The syntax of system languages (query, maintenance) can be defined by the user through a meta language (similar to ALGOL).

5.3.3 Specific Features

DM-1 has been designed independently of a specific computer. However, the system is designed to utilize fully the capabilities of whatever operating system that DM-1 is to run under. A standard interface is provided to the computer through the use of the data service routines, which are theoretically the only computer dependent routines of the system.

DM-1 has a very comprehensive logical file structuring capability. The system has been designed to provide a very powerful data structure facility that can provide complex hierarchies of logical relationships in the form of general linked-tree structures. The data description language permits the use of variable length data elements, optional data elements, and nested structures.

The DM-1 data pool is a series of fixed length data segments containing an unformatted stream of bits. The segment size of the data pool is independent of the logical file organization and theoretically utilizes the maximum buffer size of the operating system. The DM-1 access mechanism, which utilizes the IOCS, transfers the data between the data pool and core. The segmented data stream is interpreted in core by the data service routines with the aid of the system directories and the segment index.

REFERENCES

(1) The MITRE Corporation, A Survey of Data Management Systems, MTR-5036, 7 March 1968.

(2) EDP Analyzer, Vol. 5, No. 12, December 1967, Canning Publications, Inc.

(3) EDP Analyzer, Vol. 6, No. 1, January 1968, Canning Publications, Inc.

(4) American Documentation Institute, Annual Review of Information Science and Technology, Carlos A. Cuadra, ed., Interscience Publishers, Div. of John Wiley & Sons, 1967.

(5) Sutherland, I. E., "Sketchpad: A Man-Machine Graphical Communication System" AFIPS Conf. Proc. (1963 Spring Joint Computer Conf.)

BIBLIOGRAPHY

Bachman, C. W., "Software for Random Access Processing, "Datamation, April 1965.

Bachman, C. W., Williams, S. B., "A General Purpose Programming System for Random Access Memories, "Proceedings Fall Joint Computer Conference, 1964

Bleier, R. E., "Treating Hierarchical Structures in the SDC Time-Shared Data Management System (TDMS)," SP 2750, 20 August 1967.

Computer Sciences Corporation, "COGENT II Specifications," 1967.

Computer Sciences Corporation, "Introduction to COGENT II," 1967.

Defense Intelligence Agency, "IDHS 1410 Formatted File System: Programming Maintenance Manual," Washington, D. C., 20 January 1967, (DIAM 65-9-4) (AD 648 020).

Defense Intelligence Agency, "IDHS 1410 Formatted File System: File Maintenance and File Generation Manual," Revised ed. Washington, D. C., 1 August 1966, DIAM-65-9-1, (AD 637 018).

Defense Intelligence Agency, "IDHS 1410 Formatted File System: Retrieval and Output Manual," Revised ed. Washington, D. C., 1 August 1966, DIAM-65-9-2, (AD 637 018).

Defense Intelligence Agency, "IDHS 1410 Formatted File System Operator's Manual", Washington D. C., 1 January 1966, DIAM-65-9-3, (AD 637 019).

Franks, E. W., "A Data Management System for Time-Shared File Processing Using a Cross-Index File and Self-Defined Entries", Proceedings Spring Joint Computer Conference, 1966.

General Electric, "Integrated Data Store," CPB-483 (5C 10-16).

General Electric, "Introduction to Integrated Data Store," CPB-1048, April 1965.

North American Aviation, Inc., and IBM, Data Language No. 1 (DL-1) Encyclopedia, Publication 2541-F, April 1967.

North American Aviation, Inc., and IBM, ICS: Information Control System.

Postley, J. A., "Informatics MARK IV File Management System: Summary of Characteristics," AIS 66-016.

Postley, J. A., "MARK IV System," Datamation, January 1968.

Rauncher, V., Schwimmer, H. S., "The Basic Language Specification of TDMS, Phase II: Report Production, "System Development Corporation, TM-3370/008/00.

Rauncher, V., "The Language Specifications for the Query Operation of TDMS, " System Development Corporation, TM-3370/004/00.

Rauncher, V., "The Language Specifications for the Update Operation of TDMS," System Development Corporation, TM-3370/005/00.

Reynolds, J. D., Bosak, N., Shiban, J. R., "The Language Specifications for the Load Operation of TDMS," System Development Corporation, TM-3370/007/00.

Sable, J. et al., "Design of Reliability Central Data Management Subsystem," Final report, Auerbach Corporation, Philadelphia, Pa., July 1965, Vol. 2, RADC TR-65-189-Vol. 2, (AD 469 269).

Sable, J., Crowley, W., Rosenthal, M., Forst, S., Harper, P., 'Reliability Central Automatic Data Processing Subsystem," Vol. 1 Design Specification Report. Final Report Auerbach Corp., Philadelphia, Pa., August 1966, (Report no. 1280-TR-Vol. 1) RADC TR-66-474-Vol. 1.

Sable, J., Crowley, W., Rosenthal, M., Forst, S., Harper, P., "Reliability Central Automatic Data Processing Subsystem," Vol. 2: Design Specification Report (Cont'd). Final Report, Auerbach Corp., Philadelphia, Pa., August 1966, (Report no. 1280-TR-Vol. 2) RADC TR-66-474-Vol. 2 (AD 489 667).

Sable, J. Minker, J., "Reliability Central Automatic Data Processing Subsystem," Vol. 3: Data Management System Survey. Final Report, Auerbach Corporation, Philadelphia, Pa., August 1966, (Report no. 1280-TR-Vol. 3) RADC TR-66-474-Vol. 3 (AD 489 668).

UNCLASSIFIED
Security Classification

DOCUMENT CONTROL DATA - R & D

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

1. ORIGINATING ACTIVITY *(Corporate author)*

The MITRE Corporation
1820 Dolley Madison Boulevard
McLean, Virginia 22101

2a. REPORT SECURITY CLASSIFICATION
UNCLASSIFIED

2b. GROUP
N/A

3. REPORT TITLE

Survey of Management Information Systems and Their Languages

4. DESCRIPTIVE NOTES *(Type of report and inclusive dates)*
N/A

5. AUTHOR(S) *(First name, middle initial, last name)*

James P. Fry
John A. Gosden

6. REPORT DATE
May 1968

7a. TOTAL NO. OF PAGES
30

7b. NO. OF REFS
29

8a. CONTRACT OR GRANT NO.
AF 19(628)-5165

b. PROJECT NO.
34.2.3

c.

d.

9a. ORIGINATOR'S REPORT NUMBER(S)
MTP-313

9b. OTHER REPORT NO(S) *(Any other numbers that may be assigned this report)*
None

10. DISTRIBUTION STATEMENT

This document has been approved for public release and sale; its distribution is unlimited.

11. SUPPLEMENTARY NOTES

Data Management Series No. 1

12. SPONSORING MILITARY ACTIVITY Dep. Dir., NMCSTS
Defense Communications Agency, South Court
House Road, Arlington, Virginia 22204,
Attn: Mr. Reynold Thomas, Jr., Code 912

13. ABSTRACT

Significant data management systems available on third-generation hardware are considered. Three types of user interface are discussed: Own DML (Data Management Language) Systems, Forms Controlled Systems and POL Embedded Systems. Typical systems within each category are presented and their salient features are highlighted.

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Data Management Systems						
Computers						
Data Structure						
Query Languages						
Information File Processing						
Retrieval						
Compilers						
Data Bases						
Data Management						
File Indexing						
Information Systems						
File Maintenance						
Data Validation						
POL						