SPERRY UNIVAC
Operating System/3 (OS/3)

Introduction to Data Management

SPERRY UNIVAC

UP-8031 Rev. 1

H

This document contains the latest information available at the time of publication. However, Sperry Univac reserves the right to modify or revise its contents. To ensure that you have the most recent information, contact your local Sperry Univac representative.

Sperry Univac is a division of Sperry Rand Corporation.

AccuScan, FASTRAND, PAGEWRITER, SPERRY UNIVAC, UNISCOPE, UNISERVO, and UNIVAC are trademarks of the Sperry Rand Corporation.

 PRINTED IN U.S.A.

preface

This manual is one in a series designed to introduce the software available with the SPERRY UNIVAC Operating System/3 (OS/3). The actual programming procedures required to use the software described are not included in this introductory series. Such detailed information is beyond the scope and intent of these manuals and is included in the appropriate OS/3 user guide or programmer reference manual.

data management-essentially

Computer software can be categorized as control programs, service programs, and problem programs. The control programs are the operating system; the service programs may be processing programs (like the language processors — the assembler or the FORTRAN compiler, for example), or utility programs (like the linkage editor and the librarian); and problem programs are user programs. All must function together to resolve data processing requirements in the most efficient and economical manner, and achieving this economy and efficiency requires that all these programs be well designed and carefully programmed. This is especially true of the operating system programs.

A brief description of the nature and functions of the operating system is contained in the *UNIVAC Operating System/3 Introduction to the Supervisor.* Each of the operating system's functions is the responsibility of one operating system component and each component is described in a document within the UNIVAC OS/3 introductory series. The subject of this manual is the UNIVAC OS/3 Data Management.

Defining data management as a component of the operating system might raise some objections from those who insist that data management is a service program. But service program or control program, data management in your computer system is essential – just as essential as management always is, whether it's running a department store or supervising a production line. And to manage operations, it is always necessary to consider the physical plant, the facilities organization, and the actual control procedures. These are also the things entailed in data management for your computer. To manage the data your system uses, to keep order in all that information – information that starts and ends as names, addresses, part numbers, zip codes, pay rates, descriptions, floating-point numbers, man-hours, registers, counters, codes, and so on, ad infinitum; information that appears as printed characters at one point, binary digits at another, magnetic codes here, and holes in paper cards there – that is the function of the data management facility.

Data management is always essential, but only in assembly language programming is the scope of that need obvious. In COBOL or RPG, your compiler translates your statements and specifications into instructions for the data management routines. In assembly language programming, however, the

programmer must provide those levels of physical, organizational, and procedural description directly to the data management facility. Therefore, this booklet approaches data management from an assembly language programmer's point of view.

The data management facility supplies the interface between your programs and the input and output devices of your computer system. It provides methods for organizing and accessing your data; it provides you with control and protection for your files; and it contributes to the efficiency that is necessary to realize the benefits of electronic data processing.

It is necessary for data management to know what peripheral devices, which file organizations, and what input and output procedures are to be used. This is the information required by data management.

WHAT INPUT AND OUTPUT DEVICES?

The UNIVAC OS/3 Data Management provides support for a wide variety of peripheral devices — everything from card readers, punches, and printers to paper tape, magnetic tape, and disc devices.

To inform the data management facility about the peripheral devices your program will need and how these devices are to be assigned in your system, you must use job control statements and declarative macro instructions. The job control language and its uses are described in the *UNIVAC Operating System/3 Introduction to Job Control;* for now, just remember that the operating system components — supervisor, job control, and data management — must operate together in an intimate and interdependent relationship. And remember that the UNIVAC OS/3 software is designed to make the achievement of this balance a simple task for the user.

The declarative macro instructions, however, are the assembly language user's lines of communication to the data management facility. They allow you to specify parameters that the data management facility uses to define such things as record formats (whether fixed-length or variable, whether ASCII variable-format or undefined) and file names (which you assign to your data files).

These macro instructions always use an operation code that begins with DTF and includes an abbreviation for the particular file; for instance, DTFPR, for Define The File for the Printer. These

DTFs are the only way to provide descriptive information to the data management facility. The DTFs are tables which must contain the necessary information in precise locations relative to the beginning of the table. The DTF declarative macro instructions make it easy for you to direct the construction of the tables, without having to know the table layout.

WHAT FILE ORGANIZATION?

Besides providing the data management with descriptions of the input and output devices, you must also describe the file organization you want to use. In most cases, this is automatic because the peripherals can be used only with a sequential file. All printer, card reader, paper tape, and magnetic tape devices use sequential organization, but with direct access devices — like your disc subsystems — you have a choice.

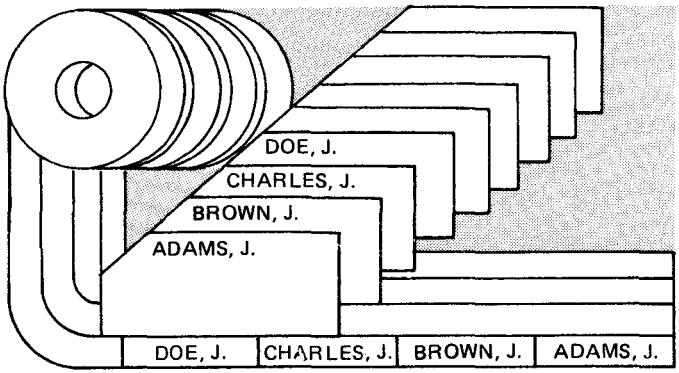
- Sequential Access Method

  By using the declarative macro instruction provided for sequential devices, or by using DTFSD for disc files, the sequential access method (SAM) processing routines are linked to your program. Then sequential files can be created or read. Output files can be prepared (prepped) automatically, which provides them with the necessary labels.

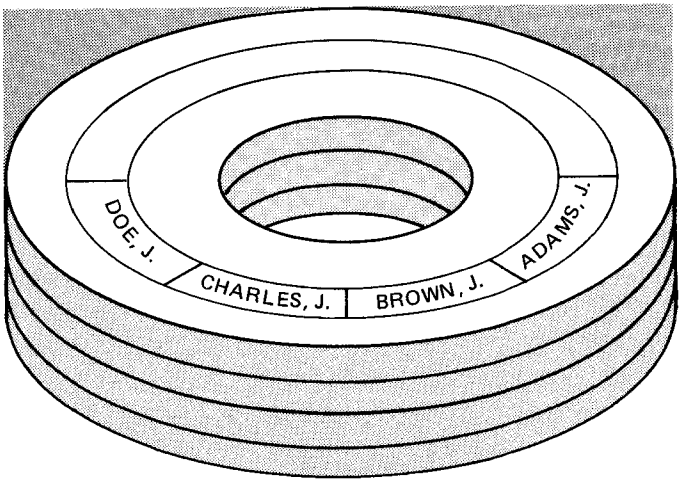
Sequential files have the simplest construction of all; one record follows another in the file. This makes it easy to access the next record in the file because there is no need for a key or an index to find it, but it also means that every record in the file must be read, starting with the first and continuing to the ten thousandth, if this is the record needed.

Problems can arise when adding and deleting records in a sequential file since the entire file may have to be duplicated to preserve the correct sequence, but the duplicates themselves provide a measure of file security and a means of double checking for errors. As you can see, there are both advantages and disadvantages to the sequential file organization. However, these are also dependent on the device type.

Card and magnetic tape files are excellent examples of files that are organized sequentially.



But disc files may be organized sequentially as well.



- Indexed Sequential Access Method

Disc files can use the indexed sequential file organization. These files are constructed so that records originally are written in sequence by record identification (or key) and then indexed as well. The advantage is that records then can be retrieved sequentially (when a consecutive group is to be accessed) or randomly by key when a specific record is needed.

A further advantage is that new records can be inserted so that they are logically (but not physically) in their rightful places between existing records. This maintains the file as a single orderly series of records.

With this file organization you have the advantages of both sequential and random access of your data records. If you want to use this arrangement for your disc files, you specify it by using the ISAM declarative macro instruction DTFIS.

- Direct Access Method

  Another file organization that is possible with disc subsystems is the direct organization. The direct access method (DAM) is used to create and use such files, and DAM processing routines are defined by the DTFDA declarative macro instruction.

  In a common form of direct access, blocked records are specified, and there are no keys in the ISAM sense. Records are retrieved by asking for record number 275, or number 62, for example. This is relative record addressing, involving no search; and data management merely reads the proper block, and points to the position in the buffer. However, DAM will also perform track and cylinder searches when blocks have been created in the hardware searchable format. Unlike ISAM, DAM does not monitor the key sequence at storage time; so the user is responsible for selecting the cylinders for search.

All of these file organization methods are available to UNIVAC OS/3 users, whether the direct access devices you use are the sophisticated variable-sector disc, or the economical fixed-sector disc. Each method of disc file construction has its advantages. Which method you choose will depend on your particular data processing needs and your preferences of file organization.

- Nonindexed Disc Access Method

  A combination of the SAM and DAM processing has been provided in OS/3 with a nonindexed macro instruction and processing module. With this access method, you can create a file sequentially, with or without keys, or you can create it randomly. An existing file can be accessed sequentially or randomly by key or relative record number. The nonindexed disc access method also allows you to divide a file into several partitions. This is useful in those cases where it is advantageous to allow for different block specifications for various segments of a file.

WHAT PROCESSING FUNCTIONS?

The last of the three kinds of information that must be supplied to the data management facility concerns

the processing functions. While the descriptions of the physical devices and the logical organizations used are accomplished by the declarative macro instruction (or DTFs), it is the imperative macro instructions that control the actual processing of the file, including reading and writing records.

Naturally the repertoire of imperative macro instructions is different for each type of device: printer, punch, disc, or whatever. For each, a full range of functions is provided. Through the data management imperatives, you can position a tape, skip a print line, retrieve a disc record — whatever your program requires. All the imperative macro instructions are described in your user's guide.

data management-logically

The data management facility that provides the services we've described does so by combining logical input and output processing modules, transient routines, interpretive software, and service routines to give you optimum control and protection of your data files without imposing great programming burdens.

- Logical IOCS

  The logical input/output control system (IOCS) consists of software modules to perform I/O functions demanded by your programs. These modules are resident in main storage to conserve program execution time, and in all but the smallest UNIVAC OS/3 configurations, the IOCS modules are designed to provide support in a multiprogramming environment.

- Transients

  Many data management functions are infrequently used — but still absolutely necessary; for example, a file is usually opened and closed only once in one execution of a program. These, and similar functions, are written as transient routines for the UNIVAC OS/3 Data Management. This means that a small area of main storage can be set aside just for transient functions — and saving in storage results.

- Interpretive Software

  Of course a major portion of the data management facility is designed to interpret the

declarative macro instructions you use to describe the files of your programs. This interpretive ability relieves the user of most of the tasks inherent in accepting various forms of data, controlling its storage, and outputting information.

- Disc Space Management

  Disc space management service routines are included as part of data management to provide you with an efficient and completely automatic space accounting and maintenance feature, which relieves you of the responsibility of keeping precise control of the contents of your direct access volumes. These routines allocate space, release unused space, scratch and rename files, and obtain label and extent information. They also resolve competing demands for allocation, establish standard interfaces, and maintain the volume table of contents for all system, permanent, and temporary files.

Special note should be made of the ability of the data management facility to provide any disc access method for your system, whether you use fixed- or variable-sector disc subsystems. All three commonly used methods of accessing disc-stored data are

provided: sequential (SAM), direct access (DAM), and search by key (ISAM). This result has been achieved by applying new methods within the data management modules, so as to use disc functions that are common to both types of subsystems.

The UNIVAC OS/3 ISAM is basically an indexed sequential method that enables you to program as if you were using a variable-length record ISAM. The UNIVAC OS/3 ISAM provides file maintenance and usage in the traditional ISAM fashion; i.e., combinations of random and sequential retrieval are possible. However, internal changes have eliminated track push-down, and unblocked overflow records. This results in more rapid processing of record 'ADDS', and contributes to the flexibility of UNIVAC OS/3 ISAM in several ways. Among these are the handling of variable-length records and the facility for an alternate sequential access method (ASAM) where files have neither an index nor any keys. This access method allows you to insert new data between records of a previously formed file. The added data is chained from prime records to produce a form of header-trailer file. Records are randomly accessed by means of relative block number plus position within block.

The combination of UNIVAC OS/3 ISAM and the fixed-sector disc subsystems gives the UNIVAC 90/30 System user a very efficient interface with low cost,

high volume disc storage that can, in some instances, surpass the performance of the standard ISAM with the more expensive disc devices.

data management-especially

Just like any other data management facility in use today, the UNIVAC OS/3 Data Management is designed to provide its services with a minimum of user programming effort. But this facility is not identical to all the other data management facilities in use; the UNIVAC OS/3 Data Management differs because it offers features of great advantage to the user:

- Custom configurations
- Efficient resource use
- Data file versatility
- High compatibility

These are the features that demonstrate that the UNIVAC OS/3 Data Management facility has a design consistent with all the UNIVAC OS/3 software – software that is meant to offer flexibility and power without increasing costs.

DATA MANAGEMENT CONFIGURATIONS

Your data management facility is custom tailored for your computer system to match your data processing needs. Each customer installation uses only those parts of the software that will give maximum service and ensure the maximum performance of your hardware. Yet, because the data management software is modular, the facility can be extended later to support any additional equipment you may attach to your system.

RESOURCE USE

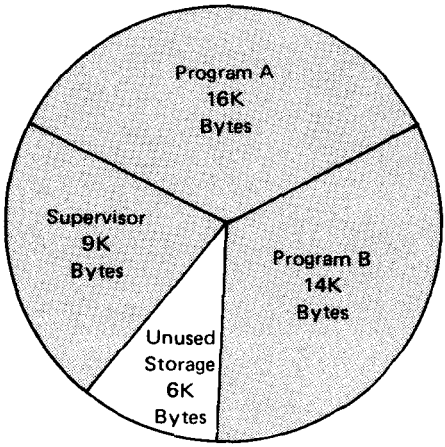
The design of the UNIVAC 90/30 System and its software represents a unique union of complementary hardware and software. Together, they provide unparalleled performance to the UNIVAC OS/3 user. The data management facility illustrates these design

principles with its use of reentrant routines; these are shared routines used to eliminate code duplication and ensure the efficient use of main storage. When two or more programs are being executed concurrently, each program will have data management requirements, some of which may turn out to be a duplicate of the other. To eliminate potential main storage waste, data management routines which are in main storage and being used by an executing program will be shared by subsequent programs requiring the same routines. All routines that can contribute to the advantages are supplied in a reentrant form.

This program sharing concept has important advantages to the user that should be examined in more detail; consider the following hypothetical examples of multiprogramming in a system where 45K bytes are available for the supervisor and problem programs.

In the first example, two programs are executed concurrently. Each is linked to all the necessary data management routines. As a consequence, 39K bytes of main storage are required, and 6K bytes are unused.

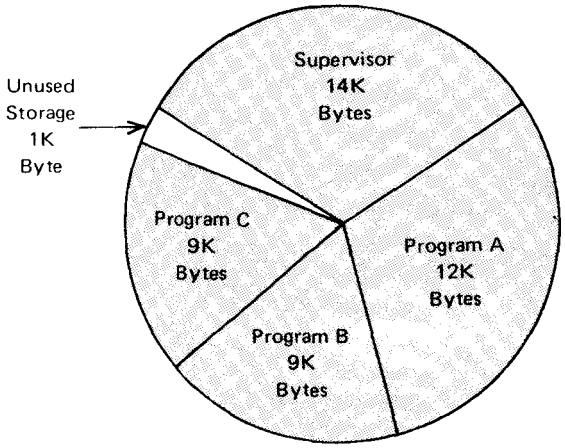
Nonreentrant Data Management



Function	Storage
Supervisor	9K
Program A	
User Code	12K
Tape data management	1K
ISAM	3K
	16K
Program B	
User Code	8K
Tape data management	1K
Printer data management	1K
ISAM	4K
	14K

In the second example, reentrant data management routines are linked to the supervisor, increasing its size, but freeing more main storage for user programs – enough to support a third concurrent program.

Reentrant Data Management



Function	Storage
Supervisor (including data management routines)	14K
Program A	
User code	12K
Program B	
User code	8K
Printer data management	1K
	9K
Program C	
User Code	8K
Card data management	1K
	9K

FILE VERSATILITY

The data management facility provides the user a full range of file organization methods to assist in establishing, organizing, maintaining, and referencing data files. These organization methods include all the standard access methods as well as the new UNIVAC OS/3 ASAM and nonindexed access methods.

COMPATIBILITY

Data management operations and procedures have a high degree of compatibility with the UNIVAC 9400 and 9700 Systems, but the emphasis of design has been on making conversions from UNIVAC 9200/9300 Series Systems, and IBM 360/20 and System/3 as painless as possible. The UNIVAC OS/3 Data Management has been designed so that data management programming changes are minimal.

summary

The transformation of raw data into useful information is the function of all data processing equipment – and the goal of all users. The UNIVAC 90/30 Data Processing System has been designed to accomplish this transformation of bits and bytes into facts and figures in the most efficient manner possible. This efficiency is made possible by the careful mating of the software and hardware components in the system, a design philosophy respected throughout the development of the UNIVAC 90/30. As one of these software components, the UNIVAC OS/3 Data Management contributes greatly to the essential task of transforming data into information whether the user writes his programs in assembly language (and controls the data management facility directly), or chooses one of the powerful UNIVAC OS/3 language processors (and uses data management automatically within his COBOL, FORTRAN, or RPG program).

Please note that a more detailed description of the features and capabilities of the UNIVAC OS/3 Data Management, complete with illustrations and examples drawn from actual practice is available in the *UNIVAC Operating System/3 Data Management User Guide.* A review of the programming used with data management, written for experienced personnel, is also available; see the *UNIVAC Operating System/3 Data Management Programmer Reference.*