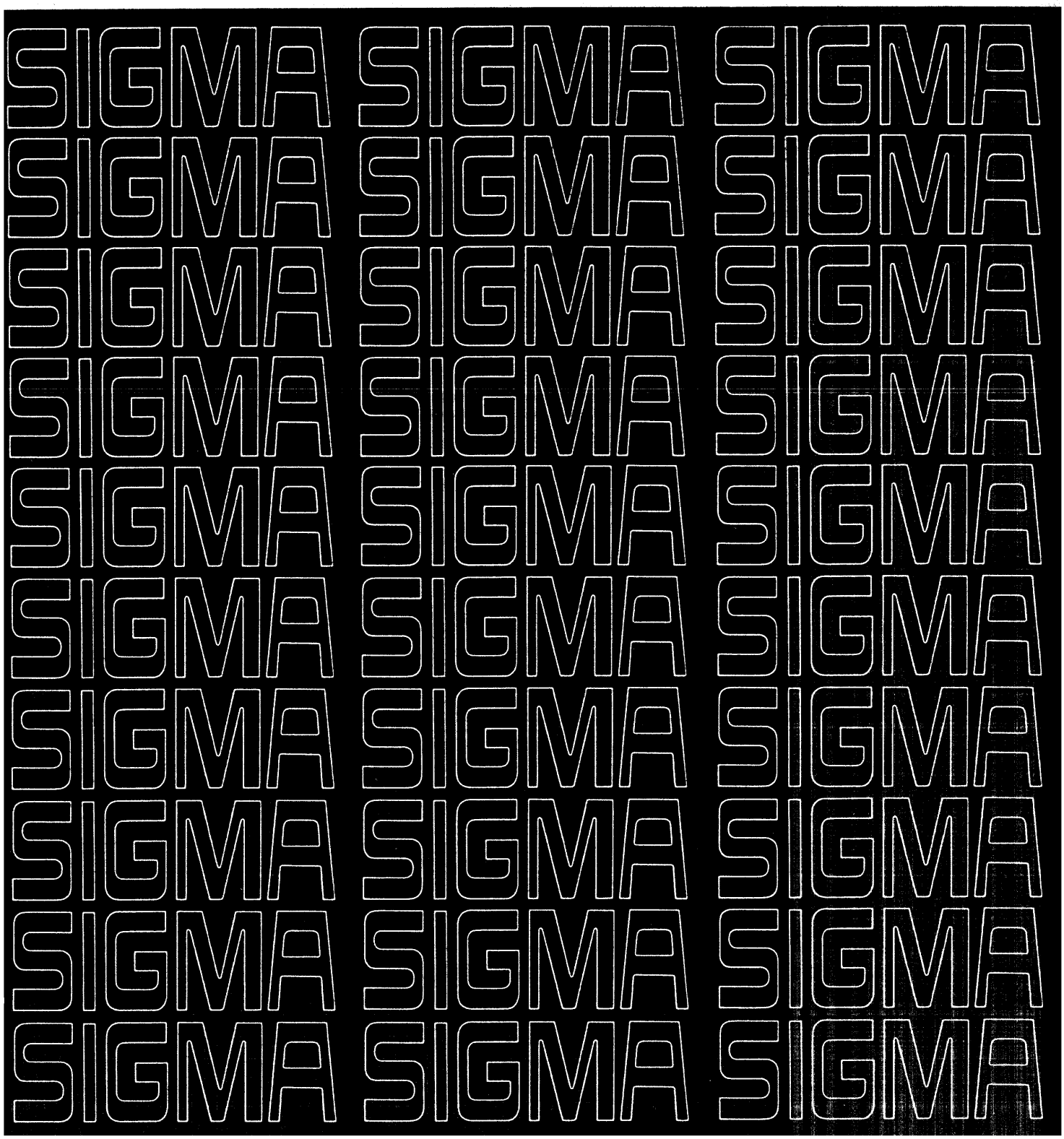
SDS

Scientific Data Systems
A XEROX COMPANY

SDS SIGMA MANAGE

Reference Manual



Price: $2.25

MANAGE
REFERENCE MANUAL

for

SDS SIGMA 5/7 COMPUTERS

PRELIMINARY EDITION

90 16 10A

July 1969

SCIENTIFIC DATA SYSTEMS A XEROX COMPANY/701 South Aviation Boulevard/El Segundo, California 90245

Printed in U.S.A.

RELATED PUBLICATIONS

Title	Publication Number
Sigma 5 Computer Reference Manual	90 09 59
Sigma 7 Computer Reference Manual	90 09 50
Sigma 5/7 Batch Processing Monitor Reference Manual	90 09 54
Sigma 5/7 Batch Processing Monitor Operations Manual	90 11 98
Sigma 5/7 Sort/Merge Reference Manual	90 11 99

CONTENTS

APPENDIXES

ILLUSTRATIONS

TABLES

1. INTRODUCTION

SDS Manage is a generalized file management system that permits decision makers with limited programming knowledge to make use of computerized data files.

Manage eliminates the need to write individual computer programs for performing many of the tasks associated with using a flexible and responsive management information system. It is made up of several subprograms which perform the functions of building the data base or making use of it. To use any of the subprograms, the user enters control codes and data from the appropriate Manage forms into the system, and the specified job will be performed.

In a typical installation, setting up the initial data files and generating standard reports will be handled by experienced business analysts or programmers. Some companies will prefer that all Manage processing be handled by computer department personnel only. However, Manage is designed so that a businessman or manager can use the system directly, with a minimum of data processing orientation. Reports to be printed (maximum of 99) are generated from only one pass through the data file, regardless of the complexity of the specified reports.

This chapter provides the user with general information on Manage program and file characteristics, and on system component programs. These component programs are explained in detail in Chapters 2, 3, 4, and 5. Appendix A presents some sample Manage deck setups, and Appendix B gives DCB names used by the Manage processors. Appendix C describes the use of registers for own-code linkage during the FILEUP phase of operation. Appendix D shows several sample Manage runs.

In the following chapters, certain conventions have been adopted for defining Manage commands. Capital letters indicate command words that are required in the literal form shown. Lower case letters are figurative representations of parameters. Command parameters enclosed by braces ({ }) indicate a required choice. Parameters enclosed by brackets ([ ]) are optional.

MANAGE PROGRAM CHARACTERISTICS

The programs that constitute Manage may be classed as "load and go" file-processing compilers. Each is a generalized program oriented toward a data file dictionary. Each data file dictionary describes the format of one specific data file. Using control parameters and applicable data file dictionaries, these programs perform the following functions:

- Create and maintain files in almost any format.
- Retrieve information, selectively or nonselectively, from such files.
- Report and/or summarize retrieved information according to the user's requirements.

File creation and maintenance entails setting up a new file or modifying the contents of an existing file. File maintenance functions consist of inserting, deleting, and changing records in a file.

Data retrieval includes the selection of records (or parts of records) from a single master file or two matching files. The selection criteria are specified by the user and allow him to examine the relationships between data fields, constants, and the results of arithmetic operations on data fields. Up to 99 criteria can be specified in any one request, linked by either of the logical operators AND and OR. Selection tests include "greater than", "less than", and "equal to" conditions, as well as their negatives. Data-retrieval queries involving a file can be processed in a single run, or as groups of up to 99 such queries involving the same file. It is also possible to select every record (or parts of every record) by the absence of selection criteria and the inclusion of an additional parameter.

Report generation involves editing of selected data fields into a prescribed format and sequence, automatic insertion and alignment of columnar headings, counting of items within groups, performing necessary punctuation of numeric data, and summarizing data in the report.

MANAGE FILE CHARACTERISTICS

The data file dictionary is the central control element in the Manage system. It consists of definitions (recorded in a RAD-based library) that precisely describe the characteristics of a particular data file. Because each data file dictionary relates to only one data file, a system contains as many dictionaries as there are files. All current data file dictionaries are contained in file DICT.

Control and formatting parameters need be defined only once; they are centralized in dictionary form in the RAD library. Note that this is an improvement on conventional program generators, which typically require file definition parameters to be submitted as input each time a report is prepared from a file.

Manage processes records that are organized by fields. Each field contains data in any one of the following representations.

1. Alphabetic: the standard SDS EBCDIC character set. The various Manage programs will insert, change, or compare fields containing up to 255 bytes of this type of data. Up to 999 bytes of data of this type may be defined as a text field for insertion and change purposes. However, the various Manage processors can use a maximum of 255 of these bytes for comparison and sorting purposes. No arithmetic will be performed on alphabetic fields by these processors.

2. Binary: a fixed-point, signed, binary integer value, from 1 to 4 bytes in length. Arithmetic and algebraic comparisons may be performed on data of this type.

3. Packed Decimal: a signed, decimal digit value, from 1 to 31 digits long, packed into a 1- to 16-byte field. Arithmetic and algebraic comparisons may be performed on data of this type.

Files to be handled by a Manage processor must be sequential files. These sequential files may consist of either fixed- or variable-length records. If fixed-length records are specified, Manage will process blocked records from user-formatted or foreign files. Variable-length records may be blocked only in Monitor-formatted files.

A variable-length record consists of a fixed-length root segment and a variable number of multiple-entry fields. Any number of different multiple-entry fields may be specified. Every multiple-entry field must specify a counter field in the root segment of each record. This counter field contains the actual number of multiple entries existing in that record, and is controlled automatically by Manage. If more than one multiple-entry field cites the same counter field, the multiples are said to be grouped. One of the fields in a group is designated as a group key. Deleting a group key deletes the entire group.

The number of multiple-entry fields actually appended to a variable-length record is controlled by the values found in the counter fields. The order of appearance of all multiple-entry fields is specified by the user.

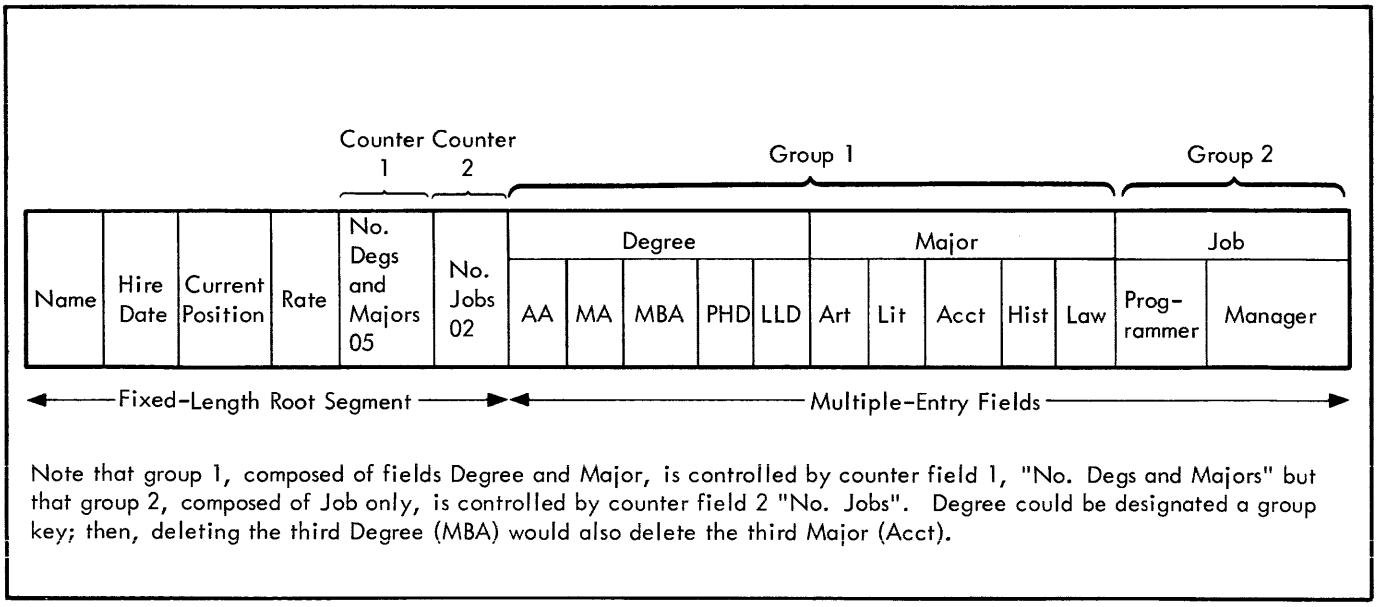
For example, consider a personnel master file in which each employee is represented by one logical record. The fixed portion of each record might contain identification and current status, while the multiple entries show level of education and job history. None or all of these types of fields could be present, but always in the same order of appearance, with Degree first, Major second, Job third, etc. The value in the counter specifies how many fields actually exist. Figure 1 shows such a variable-length record.

MANAGE SYSTEM COMPONENTS

Programs that constitute the Manage system interface with the data file dictionary that describes the format of the specific data file to be processed. Using parameters from control cards and applicable data file dictionaries, the programs perform file creation and maintenance, selective and nonselective data retrieval, and report generation. The programs in the Manage system are shown in Table 1.

Table 1. Manage Programs

Program Name	Purpose
Dictnary	Creates and maintains Manage file dictionaries and generates dictionary listing.
Fileup	Creates and maintains data files.
Retrieve	Performs selective and nonselective data retrieval and formatting.
Report	Generates reports automatically (input is from Retrieve).
Sort	Called as a subroutine to perform preliminary sorting for Dictnary, Fileup, and Retrieve data, and for Report listings when required.



Note that group 1, composed of fields Degree and Major, is controlled by counter field 1, "No. Degs and Majors" but that group 2, composed of Job only, is controlled by counter field 2 "No. Jobs". Degree could be designated a group key; then, deleting the third Degree (MBA) would also delete the third Major (Acct).

Figure 1. Sample Variable-Length Record

GENERAL PROCEDURES

The processes of file definition (via Dictnary), and file creation and maintenance (via Fileup) continue to be the responsibility of experienced computer analysts and programmers. However, the need for technical talent to make use of a Manage-oriented information system decreases markedly after the initial data base creation. Once the appropriate information has been captured in computer files, managers can make extensive use of it without any intervening computer programming effort. This reduces the delay between conception of a report and its actual delivery.

The system is extremely flexible, and is very useful for unanticipated exception reports. Additionally, the user may request output of a catalog containing either the dictionary of a particular data file, or the complete contents of file DICT (all current data file dictionaries). A catalog of an individual data file dictionary contains user-supplied information for the file and its fields, ordered by starting-byte position, and a list of field names and their starting-byte positions, ordered alphabetically.

The quick response time of a Manage information system is achieved through use of its general purpose retrieval program. This program selects requested information from a data file on any Sigma storage medium. The selection and the final disposition of the information is initiated by filling in the following set of request forms:

1. General Request Specifications Form
2. Retrieval Specifications Form
3. Retrieved Output Specifications Form

These simple forms are capable of describing highly complex combinations of retrieval and reporting. Each retrieval described by the forms is called a request and is given a request number. Up to 99 multiple requests can be satisfied at the same time, that is, by a single pass through the data file. The Retrieve control cards that are punched from the forms, whether for a single request or for multiple requests, need not be entered in any definite order.

The request specification forms are filled out, giving the characteristics of the retrieval and reporting operations. They are punched on cards and provided to Retrieve, which reads them and verifies them for correct syntax. The specifications for all the required reports are then placed in a report specification output file. The specified data base file is then opened and each record in it is examined. If needed, a matching "secondary" file record can also be included in the selection process. Under this procedure, only those primary records with a matching secondary will be examined for retrieval. In the event of multiple secondary records with a matching primary, only the first secondary will be examined for retrieval. All fields in both matching records are accessible for any standard usage. The retrieval criteria for each individual request are used to select records or portions of records from the inputs and to pass the "hits" into a retrieved output file.

The report-specification and retrieved-output files are passed to Report for the final processing steps. The specifications for listing each report are examined in the following cycle:

1. If the report is in a sequence different from that of the original data base, Sort is called to create a resequenced file.
2. The report is formatted as specified in the originating request.
3. The report is printed.

EQUIPMENT CONFIGURATION REQUIREMENTS

The minimum system requirement is a Sigma 5/7 with Batch Processing Monitor capability. It is estimated that most meaningful applications can be processed with 15,000 words of storage available for the Manage processors and working areas. Also, approximately 28,000 words of RAD storage will be dedicated to Manage program storage. Each user file dictionary entered into the system will require approximately five words of RAD space per defined field.

All Manage processors use byte string operations, thus requiring the byte string simulator on the Sigma 5.

If any packed-decimal fields are defined by the user, or if print-editing functions are requested, the decimal hardware or decimal simulator is required.

The retrieval process is programmed to generate a large temporary file (the retrieved records) on the RAD. The space required is therefore user-dependent and cannot be estimated. If magnetic tape units are available, the user may reduce his RAD requirements by assigning the retrieved records to tape.

The report process may call upon Sort before listing each request. This will require temporary RAD space to hold all data records retrieved for a given request. If magnetic tape units are available, the user may reduce his RAD requirements by assigning the Sort program's files to tape.

2. DICTNARY

Dictnary is the Manage dictionary-generating program that creates and updates the data file dictionaries, which describe the user's file characteristics. These user data file dictionaries are in turn processed by other Manage programs. (Any non-Manage program may read the dictionary, but alteration by other than the dictionary generator is not allowed.) The data dictionaries are kept in the system RAD library as the file named DICT. Two records are required for each user's file description. In addition to generating dictionaries, Dictnary outputs a catalog of file dictionaries and other information, subject to user specification.

Logically, the dictionary generator must be the first Manage function to be performed, but it is independent of other Manage programs and need not be run with the others. It is important that the correct time and date be input at system load-time to ensure proper dating when the dictionary generator and the other Manage programs are called.

Each data file dictionary contains the following information:

1. General user file structure.
2. Format of the records that constitute the user file.
3. Attributes of the data fields contained in a record.
4. Processor-dependent parameters such as punctuation and column headings for the Report program.

FILE CONSTRAINTS

Each file must have at least one sort key. This parameter is required by Fileup to ensure proper sequence of the data base.

All fields (other than sort key fields) may be redefined completely or in part. For example, an alphabetic area defined as "ADDRESS" might have two other fields called "STREET" and "CITY" which reference parts of the overall area. Redefinition is also permitted for fields containing packed decimal or binary values, but is not permitted for multiple-entry fields.

In addition, each data file with variable-length records is subject to the restrictions listed below:

1. Files containing blocked records must be Monitor-formatted (LABEL or FILE parameter in a DCB ASSIGN). Unblocked records may be either Monitor-formatted or user-formatted (DEVICE parameter in a DCB ASSIGN).
2. Multiple-entry fields, ordered consecutively must follow the fixed fields in a record.
3. A multiple-entry field must reference a separate counter field (present in every record) that contains the number of iterations of the multiple-entry field in that record.
4. A counter field may be referenced by more than one multiple-entry field, but each multiple-entry field may reference only one counter. Counter fields must always be defined as binary values.
5. Counter fields and key fields may not be defined as multiple fields.
6. Groups of multiple fields that reference the same counter must always have a group key field. In a group consisting of only one data field, it also is designated as a group key.

The maximum number of fields in any one data file's dictionary is 409. The maximum number of multiple-field groups per file is 15.

Record and field sizes are generally limited by the number of digits that can be handled in the size columns of the specification cards. However, the following outline presents practical limitations:

1. Binary fields are limited to 4 bytes.
2. Packed-decimal fields are limited to 31 digits, plus an algebraic sign.
3. Mixed-mode arithmetic operations or comparisons will not be performed by Retrieve or Report.
4. Alphabetic fields are limited to 255 characters for comparison and sorting purposes.
5. Text fields are limited to 999-character fixed fields. No arithmetic or editing can be done. (If a text field is a key field, comparison is limited to the first 255 characters.) The purpose of text fields is to print horizontal text in Report regardless of other column usage.
6. Depending on field size, other Manage programs may truncate (from the left) or round values when doing arithmetic operations.
7. If user labels consist of multiple records and/or an end-of-file gap, the file must be specified as user-formatted.

CALLING DICTNARY

Input for a Dictnary run consists of the following cards:

1. Dictnary processor call card.
2. File definition cards.
3. Field definition cards.

Sort orders the definition cards by file name, card type, fixed fields, multiple fields, and starting position or multiple-order number. File dictionaries are processed one by one and field by field. Errors will be flagged by

diagnostic messages accompanied by the incorrect card image. If errors are found, the new version will be rejected without affecting the existing dictionary. (If the catalog is restricted to successful updates and creations, the file's dictionary will not be cataloged. If the catalog is to be of all dictionaries, the old version, if any, will apply.)

The catalog will be output after all updates and creations have been processed. The first page of the catalog contains the names of files successfully updated or created. The exceptions are dictionaries that have been deleted and those with a suppress indicator on the file definition card. Deleted file dictionaries are noted in messages output before the catalog.

Formats of Dictnary cards are given below. An asterisk (*) after column numbers indicates that the field is optional.

The format of the Dictnary processor call card is fixed. As it signals BPM to link the dictionary generator, it must be the first card in the deck. Its form is shown below,

where

!DICTNARY is required as shown, starting in column 1.

b indicates a blank character in column 10.

{b A S} indicates the type of catalog to be printed. A blank character in column 11 specifies that the catalog is to consist of those file dictionaries successfully created or updated during this run. An A indicates that the catalog is to consist of all file dictionaries in existence. An S (suppress) means that no catalog is to be printed.

Both file and field definition cards may be input for more than one file at a time without being ordered.

FILE DEFINITION CARD

See Figure 2 for a diagram of the file definition card. The file definition record is not required when an existing dictionary is being changed.

Columns	Description
1	1: identifier for file definition records.
2-9	File name: a left-justified name composed of eight alphabetic (EBCDIC) characters. Embedded blanks are not allowed, though trailing blanks are acceptable. The allowed characters are A to Z, 0 to 9, @, $, #, and %.
10-13	Maximum logical record length: from 0001 through 9999 bytes, with leading zeros; field is blank if only a dictionary list is required (see column 18).
14-16*	Blocking factor: for user-formatted file, from 001 through 999; for Monitor-formatted file, zeros or blanks. (See column 18).
17*	User header label indicator: an H indicates that the header is a single record, F indicates that the header consists of one or more records ending with a file mark. (See column 18).
18*	Dictionary list option: an L means that this file's dictionary should be purged of any void entries and then listed. The L option is used when other specifications on this card are blank and no field definition cards for this file are input. An S indicates that the listing of this dictionary creation or update should be suppressed (unless the Dictnary processor card specifies that all dictionaries be listed).
19-80*	Comments: left-justified alphanumeric field to be listed in the dictionary catalog.

An existing dictionary may be deleted completely by submitting a file definition card with no record characteristics or dictionary list option (that is, columns 10 to 18 blank). The complete deletion of a file dictionary will be noted in the listing output before the catalog with the file name and a suitable message. Also, the latest contents of any dictionary may be listed by submitting a file definition card with an L in the dictionary list option (column 18). A scan will take place to purge void entries. (A void entry is created when an existing definition is changed from a fixed field to a multiple-entry field. This void entry exists until the next update or list of that dictionary, at which time it is deleted.)

FIELD DEFINITION CARD

See Figure 2 for a diagram of the field definition card.

Columns	Description
1	2: identifier for field definition records.
2-9	File name: a left-justified name composed of eight alphabetic (EBCDIC) characters. Embedded blanks are not allowed, though trailing blanks are acceptable. The allowed characters are: A to Z, 0 to 9, @, $, #, and %.
10-17	Field name: a left-justified name composed of eight alphabetic (EBCDIC) characters. Embedded blanks are not allowed, though trailing blanks are acceptable. The allowed characters are: A to Z, 0 to 9, @, $, #, and %.

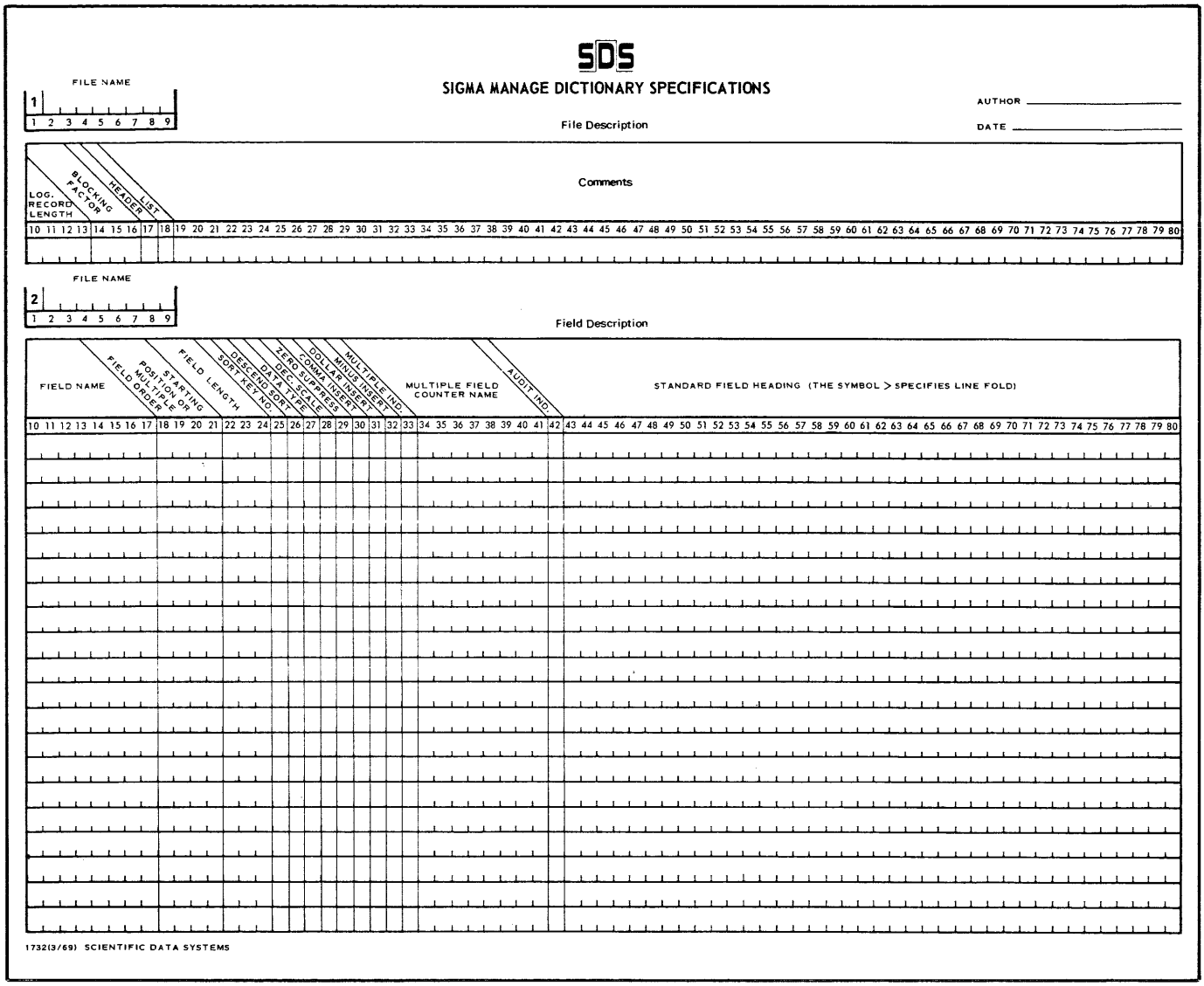


Figure 2. Manage Dictnary Specifications

Columns	Description
18-21	Starting-byte position or multiple-entry order: four-digit starting-byte location of this fixed field. The leftmost byte in a record is byte 0001, the second is 0002, etc. If the field is a multiple-entry field, this value indicates its ordering as to all other multiple-entry fields, that is, 0001 for the first multiple-entry field, 0002 for the second, etc. Blanks are used if this field is to be deleted.
22-24	Field length: from 001 through 999 bytes, with leading zeros. Blanks are used if this field is to be deleted.
25*	Key level: if the field is blank, it is not a key field. An M indicates that this is a multiple-entry group key. It associates all multiple entries tied to the same counter. A numeral from 1 through 9 indicates that this is a file sort key. The specified value is the sort level, where 1 specifies the primary-level sort, 2 is the secondary-level sort, etc. A text field comparison is limited to the first 255 characters. Note that at least one fixed field must be a sort key, and one multiple field in each counter field's group must be the group key.
26	Sort direction: used for file sort key fields only, where a blank indicates an ascending sort, and D indicates a descending sort.
27	Field type: indicated by one of the following codes.

Code	Type	Maximum Length
B	Binary	4 bytes
P	Packed decimal	16 bytes

Columns	Description

Code	Type	Maximum Length
A	Alphabetic (EBCDIC)	255 bytes
T	Text (EBCDIC)	999 bytes

Note that packed decimal fields always contain an odd number of digits equal to (byte length x 2) -1. If the field type is text, comparison is limited to the first 255 characters.

28* Decimal scaling (for packed decimal fields): specifies the number of positions to the right of the decimal point. The field may contain a blank (no decimal point will be printed), or a numeral from 1 through 9. For example, a field whose length is specified as seven digits and has a decimal point code of 3 takes the form nnnn.nnn upon being printed. Note that the decimal point is not actually carried in the data, but is inserted only at the time of printing.

29* Zero suppress: a Z entry inhibits the printing of leading zeros for the field's contents. If column 29 is left blank, leading zeros are printed unless the field also has a C (comma) specification in column 30.

30* Comma insertion: a C causes commas to be inserted in a numeric field prior to printing. Commas are placed to the left of every third digit counting from the units position. Thus, a numeric field whose contents are 1000000 will be printed as 1,000,000. If the contents were 0600000, the displayed quantity would appear as 600,000. Note that comma insertion forces suppression of leading zeros, whether specified or not (see zero suppress code, above). If column 30 is blank, commas are not inserted, nor are leading zeros suppressed (unless zero suppression is explicitly called for by use of the zero suppress code).

31* Dollar sign: a $ character causes a dollar sign to precede the most significant digit of a numeric field on printout. If column 31 is left blank, a dollar sign is not inserted.

32* Minus sign: a "-" causes a trailing minus sign to be printed when the field's content is negative. If column 32 is blank, the sign of the value will not be printed regardless of the field's content.

33* Multiple-entry indicator: an M indicates that this is a multiple-entry field.

34-41* Multiple field's counter name: name of the previously-defined root-segment field that is the counter for the number of occurrences of this field in any given record. The left-justified name is composed of eight alphabetic (EBCDIC) characters. Embedded blanks are not allowed, though trailing blanks are acceptable. The allowed characters are A to Z, 0 to 9, @, $, #, and %.

42* Audit: an A indicates that a change of data in this field due to a file update will cause the entire record to be dumped in an audit file.

43-80* Standard column heading: up to 38 EBCDIC characters of heading to be used when this field is printed. The > character (greater than) will indicate a new heading line for REPORT.

An already existing field definition may be deleted by submitting a field definition card with no characteristics other than file and field name (that is, columns 18 to 80 blank). A field's deletion will not be noted.

SAMPLE FILE DESCRIPTION

Definition of a sample personnel master file and the associated field names are listed below. The file contains fixed-length records (that is, no multiple-entry groups) and is maintained in employee-number sequence within each company division.

Field Description	Field Name	Comments
Corporation Division Number	DIVISION	
Employee Number	EMPNO	A five-digit number
Employee Name	NAME	
Street Address	L1	First line only
Employee Address	ADDRESS	The entire address
City and State	L2	Second line only
Zip Code	ZIP	
Social Security Number	SOSECNO	
Date of Birth	BIRTHDTE	Month, Day, Year
Sex	SEX	M or F
Marital Status	MARITAL	Code Meaning
		M married
		S single
		D divorced
		W widowed

Field Description	Field Name	Comments	
Citizenship Status	CITIZEN	Code	Meaning
		A	alien
		B	native born
		N	naturalized
Date of Hire	HIREDATE	Month, Day, Year	
Month of Hire	HIREMO		
Year of Hire	HIREYEAR		
Rate at Time of Hire	HIRERATE	$xx.xxx	
Number of Dependents	DEPEND		
Job Description	POSITION		
Current Pay Rate	PAYRATE	$xx.xxx	
Date Pay Rate Established	PAYDATE	Month, Day, Year	

Dictionary specifications for this example are shown in Figure 3. The resultant dictionary is shown in Appendix D, Figure 5.

DICTNARY MESSAGES

All specifications on definition cards will be examined. Errors will cause output of an appropriate message, preceded by a printout of the incorrect record. In the messages shown below, the parameter "a" represents the user's file name, and "b" or "c" represents the user's field name.

BAD SYS OR RAD I/O RETURN : nn

There has been an irrecoverable and unexpected return while processing file DICT, where nn represents the BPM return code.

DICTIONARY SPECIFICATION READ ERROR

There was an error by Sort in reading the specifications. Dictnary aborts.

FILE a DICTIONARY DELETION

This message records a successful dictionary deletion; it precedes the catalog printout.

SDS

SIGMA MANAGE DICTIONARY SPECIFICATIONS

AUTHOR KOPITO

DATE JULY 1969

1 FILE NAME: MASTER

File Description

Comments

LOG RECORD LENGTH | BLOCKING FACTOR | HEADER | LIST

0184 | AMALGAMATED MANUFACTURING CORP. - MASTER FILE

2 FILE NAME: MASTER

Field Description

FIELD NAME	FIELD ORDER	POSITION OF MULTIPLE	STARTING POSITION	FIELD LENGTH	SORT KEY NO.	DESCEND SORT	DATA TYPE	ZERO SUPPRESS	DEC. SCALE	DOLLAR INSERT	COMMA INSERT	MINUS INSERT	MULTIPLE IND.	MULTIPLE FIELD COUNTER NAME	AUDIT IND.	STANDARD FIELD HEADING (THE SYMBOL > SPECIFIES LINE FOLD)
DIVISION	00	03	00	11			B	Z								DIV.
EMPNO	00	07	00	22			B									EMPL> NO
NAME	00	09	02	1			A									EMPLOYEE NAME
ADDRESS	00	30	03	9			T									
L1	00	30	02	0			A									STREET
L2	00	50	01	9			A									CITY-STATE
ZIP	00	69	00	3			P									ZIP>CODE
SOSECNO	00	79	00	9			A									SOCIAL>SECURITY>NUMBER
BIRTHDTE	00	88	00	6			A									DATE OF> BIRTH >MO/DY/YR
SEX	00	94	00	1			A									S>E>X
MARITAL	00	95	00	1			A									M>R>T>L
CITIZEN	01	01	00	1			A									C>T>Z>N
HIREDATE	01	02	00	6			A									DATE OF> HIRE >MO/DY/YR
HIRERATE	01	09	00	3			P	3	Z	$						RATE> AT >HIRE
DEPEND	01	32	00	1			B	Z								NO>OF>DEP
POSITION	01	37	01	0			A									JOB>TITLE
PAYRATE	01	51	00	3			P	3	Z	C	$	-				RATE> OF > PAY
PAYDATE	01	54	00	6			A									DATE >RATE SET>MO/DY/YR
HIREYEAR	01	06	00	2			A									
HIREMO	01	02	00	2			A									

1732(3/69) SCIENTIFIC DATA SYSTEMS

Figure 3. Sample Personnel Master File Specifications

FILE a FIELD b BOUNDARY OUTSIDE RECORD

The specified field begins at zero or ends outside the boundaries of the record. Check the field start position and length against total record length.

FILE a FIELD b COUNTER NOT BINARY OR HAS NO GROUP KEY

Output of this message is due to one of the following causes:

1. A fixed field that was tagged as a counter field by a multiple field is not binary.
2. There is no multiple field designated as the group key associated with this counter field.

FILE a FIELD b ILLEGAL AUDIT INDICATOR

A value other than A or blank has been input.

FILE a FIELD b ILLEGAL CARD TYPE

The card type was neither a 1 nor a 2; 2 is assumed.

FILE a FIELD b ILLEGAL COMMA INSERT

A value other than C or blank has been input.

FILE a FIELD b ILLEGAL COUNTER NAME

Either a nonlegal alphabetic character combination has been found, or the given field does not exist in the current dictionary.

FILE a FIELD b ILLEGAL DECIMAL SCALING FACTOR

The value is nonnumeric.

FILE a FIELD b ILLEGAL DOLLAR SIGN INSERT

A value other than dollar sign or blank has been input.

FILE a FIELD b ILLEGAL FIELD LENGTH

Either a nonnumeric value, blanks, or all zeros have been found, or the length is too large for the type of field.

FILE a FIELD b ILLEGAL MULTIPLE ENTRY ORDERING

Either a nonnumeric value, blanks, or all zeros have been found, or there is a gap in the ordering. All further multiple-entry definitions will result in errors.

FILE a FIELD b ILLEGAL NAME

Output of this message is due to one of the following causes:

1. An illegal alphabetic character combination in file or field name has been input.
2. A duplicate-named field definition card has been input.
3. A group key field has been specified as a counter field.

FILE a FIELD b ILLEGAL NEGATIVE SIGN

A value other than minus or blank has been input.

FILE a FIELD b ILLEGAL REPLACEMENT OF COUNTER FIELD

A change from a fixed- to a multiple-type field has been defined, but other multiple fields require this field as their fixed-field counter.

FILE a FIELD b ILLEGAL SORT KEY DIRECTION

If this field is a sort key field, a value other than A or D has been found. If this field is not a sort key, a nonblank value has been found.

FILE a FIELD b ILLEGAL SORT KEY LEVEL

One of the following conditions has occurred:

1. A nonnumeric value or zero has been found for a fixed field, or the key level has already been assigned to another field.
2. A nonblank value other than M has been found for a multiple field.
3. Another multiple-entry field has been previously designated as the key to this group.

FILE a FIELD b ILLEGAL STARTING BYTE POSITION

A nonnumeric value, blanks, or all zeros have been input.

FILE a FIELD b ILLEGAL ZERO SUPPRESS INDICATOR

A value other than Z or blank has been input.

FILE a ILLEGAL BLOCKING FACTOR

A nonnumeric value or an embedded blank has been found in the user's file blocking factor. Note that an all-blank factor is legal, as it indicates a Monitor-formatted file.

FILE a ILLEGAL BLOCKING OF VARIABLE LENGTH RECORDS

Manage will not process user-blocked records with multiple-entry fields. The acceptable forms for variable-length records are Monitor-formatted file or user-formatted unblocked file.

FILE a ILLEGAL HEADER INDICATOR

The indicator was not an F, an H, or a blank.

FILE a ILLEGAL LOGICAL RECORD LENGTH

A nonnumeric value, a blank, or all zeros have been found when the card is neither a delete nor a list-only option card.

FILE a KEYS IMPROPERLY SEQUENCED

Either there is not at least one sort key, or the sort key level numbers are not contiguous.

FILE a MEMORY OVERFLOW

Output of this message is due to one of the following causes:

1. There are too many fields for the new dictionary. Overflow may be caused by void entries, or by an excess of definition cards being created or transferred from the old version.

2. Not enough working core was available to set up the dictionary. This may happen during the update or catalog phase.

3. After processing some entries, additional working core was not available to hold more standard column headers.

4. Work areas have become saturated by defining too many files on one pass.

Further analysis of this file's dictionary by Manage is inhibited.

FILE a NAME ILLEGAL

There is an illegal alphabetic character combination in the file name.

FILE a NO DICTIONARY OR TYPE 1 CARD

One of the following conditions has been found:

1. An attempt has been made to list a nonexistant dictionary.

2. Field definition cards have been input without a previous dictionary to update, or a type 1 file definition card is needed to create a new dictionary.

FILE a NONEXISTENT DICTIONARY

The file's dictionary is not in file DICT.

FILE : a TRANSFER OF FIELD : b REQUIRES IMPROPER OR DELETED COUNTER FIELD : c

Multiple-entry field b, in a previous version of the dictionary, requires field c as its counter. Field c has been deleted, changed to a nonbinary type, or made a multiple field in the current update.

MORE THAN 15 MULTIPLE FIELD GROUPS IN THIS FILE

There are too many multiple-entry field groups in the file.

3. FILEUP

Fileup is the MANAGE program that creates or maintains data files that are defined in file DICT. Two methods of file creation or updating are available to the user: the gang method, and the selection method.

1. The gang method accepts a single set of data and updates every record with it, regardless of previous record values.

2. The selection method accepts data that applies to each master record, and updates records according to key field selection and requested functions.

Specifications for an update may be kept permanently in the library under the user's account number. This procedure eliminates the need to submit the same parameters for each run, and shortens overhead for successive runs of the same files. The transaction file is not limited to card images. However, individual fields that are to be handled as signed numeric data must be in hardware-compatible form.

Transactions can be sorted by Fileup, if requested. Replacement, algebraic addition, and blank- or zero-fill may be specified on individual fields. Other operations must be done with user own-code routines. In addition, entire records may be selectively deleted or inserted using Fileup.

MASTER FILE DESIGN

The master file is defined in file DICT. It is the file against which all changes are applied. The old version of the master file is one of several inputs to Fileup. The new version is output incorporating all changes. The Manage dictionary generator (Dictnary) imposes the following constraints on design of the master file:

1. Each file must have at least one sequence key.

2. Each multiple group must have a group selector key field.

To ensure file sequence, Fileup will not act on the sequence key fields or their overlays. Therefore, any changes to these fields must be made by deleting the original version of the record and inserting a complete new record as a separate transaction (see record function codes).

Fileup will act on variable-length records which meet the design requirements of the dictionary generator (Dictnary). Each multiple group must have one of the fields within the group designated as the group selector key. This field must have unique values so as to identify individual occurrences of the group even though Fileup is not concerned with the sequence of all occurrences.

The counter field that controls the number of occurrences in a group is automatically maintained by Fileup and must not be cited in the update specifications.

Selection of an occurrence is made by identifying a group selection key value (not by its place or order of occurrence). Group selection keys can only be deleted, inserted, or replaced. The replacement option identifies the occurrence so that action on other fields within the group can take place. Usually, group selection keys will be associated with complete group occurrences rather than with individual values. Once a group selector key is established at a given level of occurrence, all multiple-entry fields within the group will continue to exist until the group selection key field is deleted. Conversely, if a group key is deleted, the entire group of fields at that level is also deleted. Any change involving a multiple field must be tied to a group selector key value as well as to the record sequence keys for complete indentification to the occurrence level.

TRANSACTION FILE DESIGN

The transaction file is defined by the specifications to Fileup with the file and field names serving as the link to the dictionary definition of the master file. The transaction file may contain interleaved records of more than one design. Though each record will have more than one field, the record will, hereafter, be called "a transaction." Transactions will contain the changes to be applied to the master file within the limits set under "Master File Design" above. Except for the following requirements, transaction design is up to the user, including overlaying and redefinition of field positions.

1. All transactions must be the same length.

2. All sequence key fields of the master file must be in every transaction.

3. If transactions are of more than one design, a transaction code field must be defined to identify the various formats.

4. If a combination of delete-a-record, insert-a-record, or change-a-record functions is required, a record function code field must be defined to identify the function of each transaction as a whole.

5. All transactions must have sequence key fields, transaction code field, and record function code field in the same locations.

6. Transaction codes and record function codes will be considered alphabetic (EBCDIC).

Transactions are not restricted to card images. Record function codes for records are shown below in their order of evaluation.

Code	Meaning
D	Delete a record if all keys match.
I	Insert record in proper sequence if keys do not match.
U	Update a record if all keys match.

If key conditions are not as expected, the transaction record is listed and no change is made to the master file. If no record function code is specified, the type of run will determine the default condition. A creation run will default to an insert function, while an update run will default to an update function.

A transaction must be input for each occurrence level of a multiple field. Sequence key values must be repeated and the corresponding group selector key field value must be present for occurrence identification.

AUDIT FILE DESIGN

Audit fields may be defined in the dictionary so that master file changes can be monitored. An audit file will be created if changes are made affecting an audit field. The file will contain the entire record from the old input version of the master file. Specific causes are given below.

1. An update to one or more audit fields in a record.
2. Deletion of a record due to the record function code.
3. Deletion of a record on a return from own-code (see below) to the delete function.

USER OWN-CODE

Fileup can be incorporated in the user's library as a number of special purpose programs with user own-code modules. The user assembles his program as independent relocatable object modules and then builds an overlay structure (with his new load module name) that merges his program with the Fileup root segment (FILEUP0). This results in a unique program for that particular file maintenance task. The user may then call out any desired version and specify links to his own-code at any or all of the following points:

1. Before opening either version of the master file.
2. Immediately before writing the output master file or volume header, and after reading the input master file or volume headers.
3. Immediately after reading a logical record from the input master file. (A delete record return is provided.)
4. Immediately after matching a master record to a transaction, with access to both records.
5. Immediately before writing a logical record on the output master file regardless of transaction changes. (A delete return is provided.)
6. Before closing the transaction, the audit, the input master, and the output master files.

Trailers will not be handled by Fileup; hence, they must be processed by user own-code.

Specifying user own-code accesses on the file update card will cause the link to be attempted. If the link is not successful, an immediate abort will occur.

External entry points for own-code to be defined in the user's program are given in explanations of Fileup specification cards, below.

A file update may reference:

1. Specifications
2. Transactions
3. DICT
4. Old master
5. New master
6. Audit

The specifications and DICT files will be read and interpreted first. Fileup will then reference the transactions, audit, and old and new master files. Therefore, a maximum of four DCBs will be active at one time; a user with linked own-code routines may use his own DCBs.

FIELD CONVERSION

Fileup will convert transaction data to the correct type for the master file. However, the user must be aware of certain size limitations. For example, the user is responsible for correctly scaled inputs (integer is assumed). Table 2 below shows transaction field size limitations and restrictions.

An alphabetic transaction field may not be all blank when a zero numeric value is intended. At least one right-most zero must be explicitly present. Note the use of the conditional replacement function code to bypass this restriction.

When an alphabetic transaction field contains negative values to be converted to packed decimal or binary form, it must have an overpunch (11-zone punch) over the least significant digit.

Table 2. Transaction Data Field Sizes

Transaction Field Data Type	Master Field Data Type		
	Alphabetic (EBCDIC) or text	Binary	Packed Decimal
Alphabetic (EBCDIC) or text	Equal.	Maximim transaction field size is 10 bytes.[t]	Maximum byte length is equal to (master byte length x 2) -1.
Binary	Maximum transaction field byte length is 4 bytes.[tt]	Less than or equal to master length.	Maximum transaction field byte length is 4 bytes.[tt]
Packed Decimal	Maximum byte length is equal to (master byte length/2) +1.[tt]	Maximum transaction field size is 6 bytes.[t]	Less than or equal to master length.

[t]An arithmetic overflow trap may occur during the actual update. Some values fall within the maximum transaction field size, but overflow the maximum hardware binary value ($2^{31} - 1$) upon conversion. The run will be aborted.

[tt]Arithmetic truncation of significant data may occur if the master field is not large enough to contain the converted value. The run will be aborted. Numeric transaction fields may be smaller than the master field. An automatic left-zero fill occurs.

CALLING FILEUP

Input for a Fileup run may consist of the following cards:

1. Fileup processor call card.
2. File update card.
3. Field update cards.

Fileup will call Sort to read (via the M:SI DCB) and sort the specifications by card type function, transaction code, and starting byte position. If no errors exist in the specifications, the actual update occurs.

If input/output errors are encountered during a run in either the old master file or the transaction file, the record will be listed on the logging device along with an error message telling which file was in error. Any sequence error in the master or transaction files or illegal hardware arithmetic operation will immediately terminate the job. On input/output errors, the user has the option of either ending the run, or bypassing the error and continuing the update. If continuation is elected, a sort of the transaction file will also allow continuation on input/output errors. Assignments must be made for both master files and the audit files. An assignment for the sorted transactions may be made but will default to the RAD. No file releases may be entered on the assignments, but may be done later via the File Manage program.

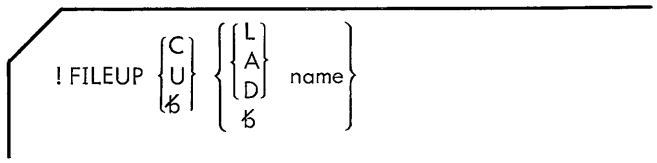
If requested, the transaction file will be sorted according to key fields, record function codes, and transaction code. Therefore, changes with the lowest record function code and transaction code for a given master record will be performed first. Gang transactions are sorted along with other transaction types. They will not be acted upon until their key values bring them in sequence with the master file. (Note that multiple fields cannot be ganged.) In addition, the user must ensure that only one record for each gang transaction is input, since a later record of the same transaction type will replace previously input values.

During the processing of a transaction, the fields will be processed in order of their starting byte in ascending order. More than one transaction may apply to a master field.

During creation of a new record, Fileup will preset alphabetic fields to blanks (hex. 40), binary fields to zero (hex. 00), and packed decimal fields to zero (hex. 00C). Note that this is for computational convenience only, and does not create a null state. If not updated by the user, such preset fields will be treated as valid data by Retrieve and Report. Undefined areas will be preset to binary zero.

Formats for Fileup run cards are given below. An asterisk (*) after column numbers indicates that the field is optional. Register use for own-code linkage is given in Appendix C.

The Fileup processor is called from the System Library with a Fileup call card. Its form is shown below.



where

! FILEUP is required as shown, starting in column 1.

{C, U, b} indicates the type of run. One of the following codes must be in column 11.

Code	Meaning
C	Create a new file.
U	Update an existing file.
b	Delete the library, where b represents a blank character.

{{L, A, D, b} name} is the library action code. One of the following codes must be in column 13.

Code	Meaning
L	Specifications are to be obtained from the library under the name in columns 14 to 17.
A	Specifications are to be obtained from the M:SI device and are to be added to the library under the name in columns 14 to 17.
D	Delete the library specifications named in columns 14 to 17. No new specifications should follow this processor call card. (When D is used, column 11 should be blank.)
name	contains the library name (left-justified) of the specifications in columns 14 to 17. This field should be blank if column 13 is blank.
b	Specifications are to be obtained from the M:SI device and are not to be saved.

If a change to library specifications is made, the entire set must be input via the M:SI device.

The specifications below define the format of the input transaction file described by the type 3 card. Each transaction record format is derived by Fileup by grouping together all the type 4 cards with the same transaction ID code. The sequence key fields in the transaction records are defined only once, and are assumed to apply to every type of transaction record. The transaction fields are given the same name that is used in the dictionary for the master file to permit association of new data with the old.

FILE UPDATE CARD

See Figure 4 for a diagram of the file update card.

Columns	Description
1	3: identifier for a file update card.
3-10	Master file name: name of the file being updated.
12-15	Transaction record length: logical record length in the transaction data file.
17-19*	Transaction blocking factor: blocking factor for user-formatted transaction file. A blank indicates a Monitor-formatted file.
21*	Transaction file header: for user-formatted files only. An H indicates that the header label is to be skipped. An F signals the presence of one or more labels followed by an EOF mark.
23*	Own-code on opening: a Y specifies own-code linkage to FUPOP before opening any files.
24*	Own-code on header: a Y specifies own-code linkage to FUPHD for validating an input user header, and for creating an output user header.
25*	Own-code on input: a Y specifies own-code linkage to FUPIM after reading each input master record.
26*	Own-code on transaction: a Y indicates own-code linkage to FUPMT after successfully matching a change transaction to an input master record.
27*	Own-code on output: a Y signals own-code linkage to FUPOM before writing each output master record.
28*	Own-code on closing: a Y indicates own-code linkage to FUPCL after all record processing is complete, but before closing any files.
30*	Sort: an S means sort the transaction file according to keys and record function code prior to update process.
32*	I/O error bypass: a B indicates bypass any transaction or input master file input/output errors and continue processing.
34-37*	Transaction ID starting byte: indicates the starting byte (from 0001 to 9999) in transactions of the field that determines the format of the transaction. This parameter is not needed if there is only one transaction format.

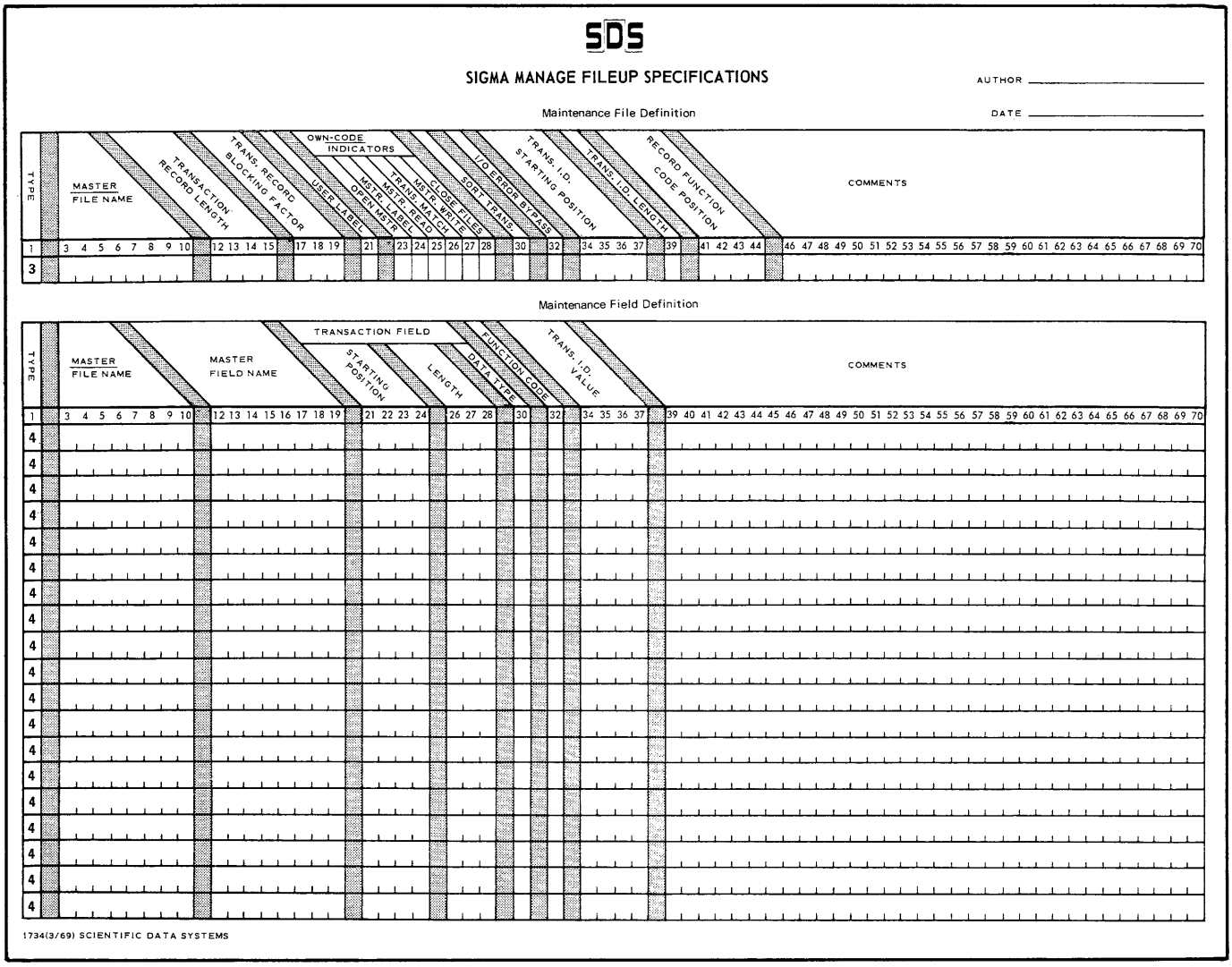


Figure 4. Manage Fileup Specifications

Columns	Description
39*	Transaction code length: byte length of transaction code (1 to 4). This parameter is not needed if there is only one transaction format.
41-44*	Record function code field position: byte position of single character record function code (0001 - 9999).

FIELD UPDATE CARD

See Figure 4 for a diagram of the field update card.

Columns	Description
1	4: identifier for a field update card.
3-10	Master file name: name of the file being updated.
12-19	Master field name: name of the master field associated with the transaction data.
21-24*	Transaction starting position: starting byte position (0001 to 9999) of transaction data associated with the master field. The field must be blank for the B or N function (see field function code below).
26-28*	Transaction field length: byte length (001-999) of transaction data. This field must be blank for the B or N function of the field function code (column 32 below).
30*	Data type: code for type of transaction, where A means alphabetic (EBCDIC), T means text, B represents binary, and P specifies packed decimal data. Note that Fileup automatically converts transaction data to the appropriate master form as specified in "Field Conversion" above. This field is blank for the B or N function of the field function code.

Columns	Description
32	Field function code: action code for specified transaction fields when applied to corresponding master field.

Code	Meaning
A	Algebraic add. Add the transaction field value to the master field. Illegal when master field is alphabetic or a multiple-entry group selector field.
B	Blank fill. Clear master field to blanks. No field in the transaction is required for this action. Illegal for nonalphabetic fields or multiple-entry group selector fields.
C	Conditional replacement. If the transaction field is not blank or zero (depending on data type), replace old master field value with transaction field value; otherwise, take no action.
D	Delete occurrence of the multiple-entry group identified by this group selection key. Only allowed for multiple-entry group key fields.
G	Gang Replacement. Replace master field value with transaction field value in every master file record from the one identified by the sequence keys until the end of the file or until the transaction is replaced by another. Not allowed with multiple-entry field.
I	Insert occurrence of multiple group identified by the multiple-entry group key. Only allowed for multiple-entry key fields.
N	Numeric zero fill. Clear master field to the correct type of zeros for the field type. No field in the transaction is required for this action. Illegal for multiple-entry group key fields.
R	Replacement. Unconditionally replace master field value with transaction field value.
S	Sequence key field. This field applies to all transaction records; therefore, transaction code is not applicable.

Columns	Description
34-37*	Transaction code: left-justified value of code that identifies type of record associated with this specification. Blank for sequence key fields.

CREATION OF A SAMPLE MASTER FILE

A system for creating and updating the Amalgamated Corporation's personnel master file is described below. It assumes that a complete record is generated for each new employee by keypunching two transactions (coded as 10 and 11) from a hiring form. Changes in job title and pay rate result in a third type of transaction (coded 22) which changes an existing master record. The transaction records are shown in Figure 5, and the required Fileup definitions in Figure 6. The record function code (column 80) would normally be an I in transactions coded 10 and 11, since these represent a new hire and should not match a previous master record. Similarly, transactions coded 22 would have a function code of U in column 80, since they represent changes to existing records.

The Fileup specifications show that the field in columns 70 to 74 of the transactions coded 11 is to be placed in both the hire rate and current pay rate fields in the master. Similarly, the field in columns 64 to 69 of the transaction is both the hire date and date of current pay rate. However, the submission of a transaction coded 22 at a later date will change only the current pay rate, date of current rate, and job title.

The Fileup specifications show either field function codes of S, which identifies the transaction file key fields, or C, which specifies conditional replacement of nonblank transaction fields. This permits partial transaction records to be generated. Note that sequence key fields are defined only once in the Fileup specifications, but apply to all transactions. If any item of information is unknown for a new hire, it may be left blank in the transaction record and will be ignored by Fileup. At some later date, the missing information may be submitted on a transaction with a record function code of U in column 80. This will cause Fileup to seek a match between the transaction and the previously established master record. At this point, Fileup will again ignore any blank transaction fields (which by now are already in the file) and will pick up only the items being filled in.

FILEUP MESSAGES

The following messages are output during Fileup. A printout of the incorrect record may accompany the message.

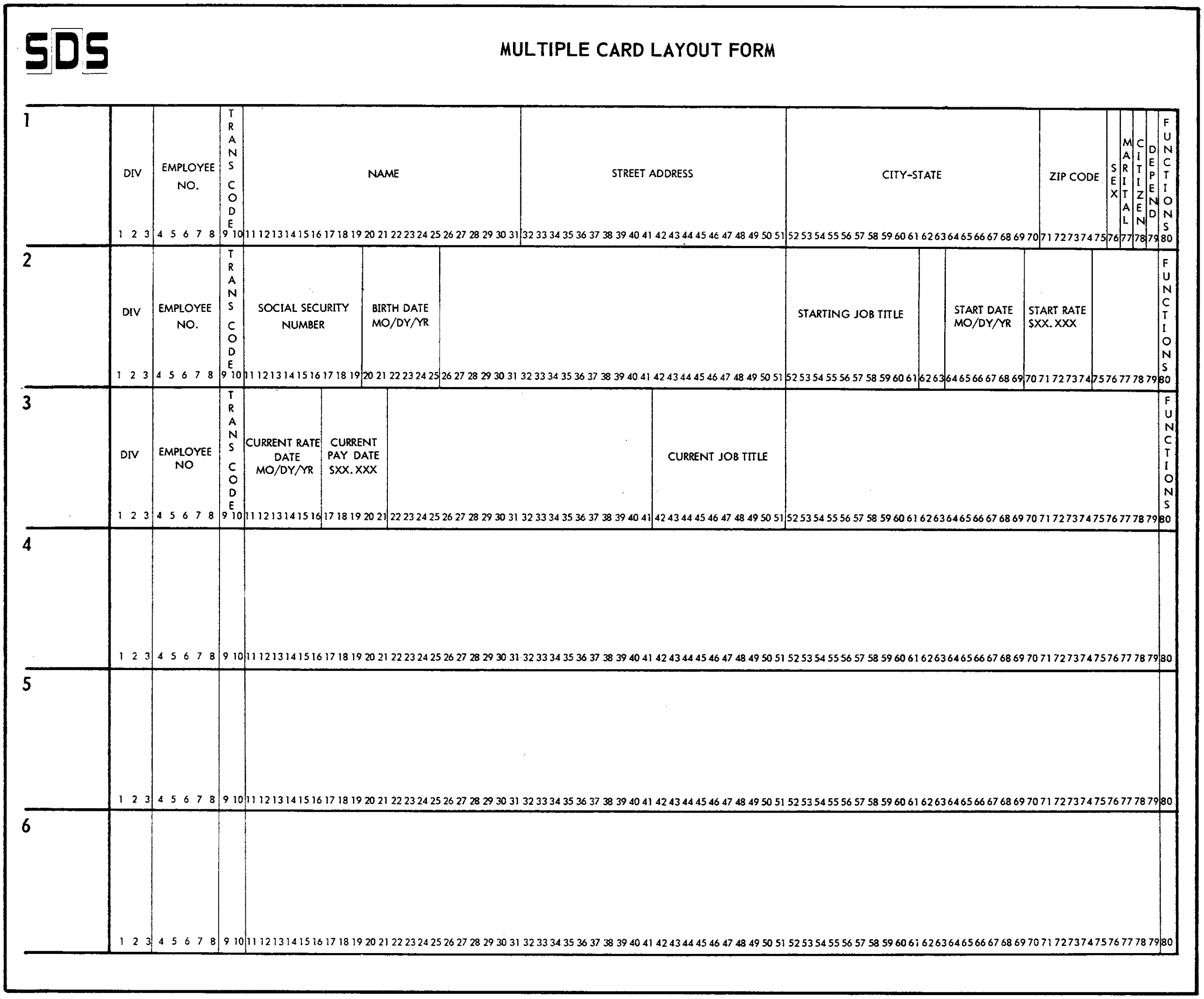


Figure 5. Sample Transaction Records

ALL KEYS NOT SPECIFIED

Either not all of the master file sort keys have been identified in the transaction record, or one or more have been rejected due to other errors.

ARITHMETIC TRAP IN THE FOLLOWING TRANSACTION RECORD

An arithmetic overflow or an illegal decimal digit trap has occurred during a field conversion operation. A mandatory abort occurs.

FILEUP SPECIFICATION READ ERROR

An error has occurred while sorting the specification input through the M:SI DCB. A mandatory abort occurs.

ILLEGAL BOUNDARY – FIELD xx

The specified transaction field, as defined by its starting position and length, is outside the bounds of the record.

ILLEGAL CARD TYPE

The card type code is not 3 or 4, or multiple type 3 cards were found.

ILLEGAL DATA TYPE – FIELD xx

The data type code must be A, B, P, or T. If the field function code specifies a blank or numeric zero fill (B or N), it must be blank.

ILLEGAL FUNCTION–FIELD xx

An error has been found in the field function code, which is subject to the following restrictions.

1. If the field is a master sort key, the field function code must be S.
2. If the transaction field is defined as text, or if the master field is defined as alphabetic or text, the field function code may not be A (addition).
3. If the master field is defined as binary or packed decimal, the field function code may not be B (blank fill).
4. If the master field is a multiple-entry field, the field function code may not be G (gang replacement).
5. If the master field is a multiple-entry field and:
   a. a group selector key, the field function code must be I, D, C, or R.
   b. not a group selector key, the field function code must not be I or D.

ILLEGAL GROUP KEY MATCH

One of the following conditions has been found.

1. An insert has been requested for a multiple-entry field but the group key to be inserted already exists in the master record.
2. A delete, replace, or update group is requested but the group key does not exist in the master record.

ILLEGAL HEADER INDICATOR

Either the transaction file header indicator is not H, F, or blank, or is F for a Monitor-formatted file.

ILLEGAL I/O ERROR CODE

The input/output error bypass indicator is not B or blank.

ILLEGAL ITEMS SPECIFIED – FIELD xx

The transaction field starting byte, length, and data type must be blank when the record function code is B or N (blank or numeric zero fill).

SDS

SIGMA MANAGE FILEUP SPECIFICATIONS

AUTHOR KOPITO

DATE JULY, 1969

Maintenance File Definition

TYPE	MASTER FILE NAME	TRANSACTION RECORD LENGTH	TRANS. RECORD BLOCKING FACTOR	OWN-CODE INDICATORS: USER LABEL	MSTR. LABEL	MSTR. REQ.	TRANS. MATCH	MSTR. MATCH	CLOSE FILES	SORT TRANS.	I/O ERROR BYPASS	TRANS. I.D. STARTING POSITION	TRANS I.D. LENGTH	RECORD FUNCTION CODE POSITION	COMMENTS
1	3 4 5 6 7 8 9 10	12 13 14 15	17 18 19	21	* 23	24	25	26	27	28	30	32	34 35 36 37	39	41 42 43 44	46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70
3	MASTER	0080	001										0009	2	0080	UPDATE AMALGAMATED MASTER

Maintenance Field Definition

TYPE	MASTER FILE NAME	MASTER FIELD NAME	TRANSACTION FIELD: STARTING POSITION	LENGTH	DATA TYPE	FUNCTION CODE	TRANS. I.D. VALUE	COMMENTS
1	3 4 5 6 7 8 9 10	12 13 14 15 16 17 18 19	21 22 23 24	26 27 28	30	32	34 35 36 37	39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70
4	MASTER	DIVISION	0001	003	A	B		MAJOR KEY - IN ALL TRANSACTIONS
4		EMPNO	0004	005	A	S		MINOR KEY - IN ALL TRANSACTIONS
4		NAME	0011	021	A	C	10	
4		L1	0032	020	A	C	10	NOTE - TRANSACTION FIELDS ARE
4		L2	0052	019	A	C	10	PUT INTO MASTER IF THEY ARE
4		ZIP	0071	005	A	C	10	NOT BLANK. BINARY AND PACKED
4		SEX	0076	001	A	C	10	DECIMAL VALUES ARE
4		MARITAL	0077	001	A	C	10	AUTOMATICALLY CONVERTED TO
4		CITIZEN	0078	001	A	C	10	PROPER FORM FROM ALPHA.
4		DEPEND	0079	001	A	C	10	
4		SOSECNO	0011	009	A	C	11	START OF NEXT TRANSACTION
4		BIRTHDTE	0020	006	A	C	11	
4		POSITION	0052	010	A	C	11	
4		HIREDATE	0064	006	A	C	11	REDEF OF TRANS FIELD UPDATES 2
4		PAYDATE	0064	006	A	C	11	MASTER FIELDS SIMULTANEOUSLY
4		HIRERATE	0070	005	A	C	11	
4		PAYRATE	0070	005	A	C	11	
4		PAYDATE	0011	006	A	C	22	START OF NEXT TRANSACTION
4		PAYRATE	0017	005	A	C	22	
4		POSITION	0042	010	A	C	22	

1734(3/69) SCIENTIFIC DATA SYSTEMS

Figure 6. Sample Fileup Specifications

ILLEGAL LIBRARY CODE

The library action code in column 13 of the Fileup processor call card is subject to the following conventions.

1. Must be L or A if library name is specified.
2. Must be D if run type is blank and library name is specified.
3. Must be blank if library name is blank.

ILLEGAL LIBRARY NAME

The library name in columns 14 to 17 of the Fileup processor call card must be blank if the library action code is blank.

ILLEGAL MASTER LENGTH – FIELD xx

The master field is greater than 31 bytes and therefore cannot be completely filled by converting the numeric transaction field to EBCDIC.

ILLEGAL NAME – FIELD xx

Either the master file name in a type 4 card is different from the name in the type 3 card, or the field name cannot be found in the dictionary of the master file.

ILLEGAL OWN CODE INDICATOR – CLOSE

ILLEGAL OWN CODE INDICATOR – HEADER

ILLEGAL OWN CODE INDICATOR – INPUT MASTER

ILLEGAL OWN CODE INDICATOR – OPEN

ILLEGAL OWN CODE INDICATOR – OUTPUT MASTER

ILLEGAL OWN CODE INDICATOR – TRANSACTION

The specified own-code indicator is not Y or blank. If the dictionary indicates a master file header, own-code to process the header is always required. If no header is specified in the dictionary, own-code for headers is not allowed.

ILLEGAL RECORD FUNCTION CODE

A record function code field is not I, U, or D.

ILLEGAL RECORD FUNCTION CODE MATCH

One of the following conditions has been found:

1. There is no key match between the V or D-type transaction record and a master record during an update run.
2. There is a match between the I-type transaction record and a master record during an update run.
3. V and D function codes were encountered in a creation run without a preceding I-type record.

ILLEGAL RECORD FUNCTION CODE POSITION

The record function code position must be either all numeric (nonzero) or all blank, and must lie within the bounds of the transaction record.

ILLEGAL SORT INDICATOR

The transaction file sort indicator is neither S nor blank.

ILLEGAL SPECIFICATION – FIELD xx

Output of this message is caused by one of the following two conditions.

1. A transaction data field overlaps a master sequence key field or a multiple-entry counter field.
2. A master key field or multiple-entry group selector key has been specified more than once on the same transaction record.
3. A multiple-entry counter field has been specified on a transaction.

ILLEGAL START POSITION – FIELD xx

The transaction field starting position must be either all numeric (nonzero), or blank if the field function code specifies a blank or numeric zero fill (B or N).

ILLEGAL TRANSACTION BLOCKING FACTOR

The transaction blocking factor must be either all numeric (nonzero) or blank if the transaction file is Monitor-formatted.

ILLEGAL TRANSACTION CODE

The value in the transaction code field does not correspond to any value supplied in the transaction specifications to Fileup.

ILLEGAL TRANSACTION CODE POSITION

Output of this message is caused by one of the following conditions:

1. The starting byte or length of the transaction identification code is not all blank or all numeric.
2. The transaction identification code length is greater than 4 bytes.
3. The transaction identification code field does not fall within the boundaries of the transaction record.

ILLEGAL TRANSACTION CODE SPECIFIED – FIELD xx

Output of this message is caused by one of the following conditions:

1. The specified transaction code is longer than the code length shown in column 39 of the type 3 card.
2. The transaction code must be blank if no code position is shown in columns 34 to 37 of the type 3 card.
3. The transaction code must be blank if the field function card is S (sequence key field).

ILLEGAL TRANSACTION LENGTH–FIELD xx

Output of this message is caused by one of the following conditions:

1. The transaction field length is not all numeric (nonzero), or blank if the field function code specifies a blank or numeric zero fill (B or N).
2. The transaction field length exceeds the maximum permitted for the type of data, where unpacked decimal fields may not exceed 31 bytes, packed decimal fields may not exceed 16 bytes, and binary fields may not exceed 4 bytes.
3. The transaction field length violates the rules of conversion to master field data.

ILLEGAL TRANSACTION LOGICAL RECORD LENGTH

The transaction record length either contains a nonnumeric character or is zero.

ILLEGAL RUN TYPE CODE

Either the run type code in column 11 of the Fileup processor call card must be U or C, or it must be blank if a specification library entry is to be deleted. If nonblank, a library deletion will not be performed.

INITIALIZATION COMPLETE–READY TO PROCESS FILES

This message signals the completion of an error-free initialization.

IRRECOVERABLE I/O ERROR xx

An irrecoverable input/output error has occurred, where xx represents the BPM error code. The job will be continued or aborted according to Fileup option.

LABEL END MARK REQUIRES DEVICE ASSIGNMENT FOR INPUT MASTER

LABEL END MARK REQUIRES DEVICE ASSIGNMENT FOR OUTPUT MASTER

LABEL END MARK REQUIRES DEVICE ASSIGNMENT FOR TRANSACTION

A header, consisting of one or more records ending with a file mark, is indicated for the named file, but it is not assigned to a device. Processing the file mark requires a device assignment.

MANAGE FILE UPDATE LIBRARY: bb FILE: cc

This message precedes the listing of specifications read through the M:SI DCB, where bb is the library name of the Fileup processor call card, and cc is the master file name from the type 3 card. The message names the master file to be processed, and the name that is associated with the new specification library entry, if any.

MASTER FILE SEQUENCE ERROR

An out-of-sequence condition has been detected in the master file, necessitating a mandatory abort.

MULTIPLE FIELD PRESENT WITHOUT ITS GROUP KEY THIS TRANSACTION

One or more members of a multiple-entry group have been defined for the indicated transaction record, but the group selector key is missing. Any transaction record containing a multiple-entry group field must also contain a group selector key.

NONEXISTENT DICTIONARY

Manage was unable to find the required master file dictionary under the specified name and current account number.

NONEXISTENT LIBRARY

Manage was unable to find the specification library entry under the name indicated on the Fileup processor call card and the current account number.

NO SPECIFICATIONS ALLOWED FOR GIVEN LIBRARY ACTION

Transaction specifications are not expected if the Fileup processor call card indicated a library deletion. The specifications will be validated but not saved, and the run will be aborted.

NO SPECIFICATIONS INPUT

Specifications were not available through either the M:SI DCB or the library.

OBSOLETE LIBRARY

The specification library entry has been made obsolete by the generation of a new master file dictionary. The

complete set of transaction specifications must be revalidated and added to the specification library.

OWN CODE LINKAGE UNSATISFIED – CLOSE

OWN CODE LINKAGE UNSATISFIED – HEADER

OWN CODE LINKAGE UNSATISFIED – INPUT MASTER

OWN CODE LINKAGE UNSATISFIED – OPEN

OWN CODE LINKAGE UNSATISFIED – OUTPUT MASTER

OWN CODE LINKAGE UNSATISFIED – TRANSACTION

The own-code linkage named is not satisfied, but is requested in the Fileup specifications.

TRANSACTION CAUSES MASTER RECORD TO EXCEED MAXIMUM SIZE

A multiple entry group insertion transaction has been found which causes the master record to exceed the maximum logical record length specified in the Dictionary. The transaction is listed and rejected. A Dictionary update run is required to indicate a higher maximum length.

TRANSACTION FILE SEQUENCE ERROR

An out-of-sequence condition has been detected in the transaction file, necessitating a mandatory abort.

TRANSACTION FILE SORT ERROR

The input transaction file could not be sorted successfully.

The messages shown below give the specified record counts after the successful completion of a Fileup run.

MASTER RECORDS INPUT xxxxxx

MASTER RECORDS ADDED xxxxxx

MASTER RECORDS DELETED xxxxxx

MASTER RECORDS OUTPUT xxxxxx

AUDIT FILE SIZE xxxxxx

TRANSACTION RECORDS IN xxxxxx

4. RETRIEVE

The Retrieve program extracts data from a data base file according to user-specified search criteria. Retrieve analyzes the user's specifications to generate the code necessary to satisfy the request. Requests are expected to be batched. A maximum of 99 requests will be processed with a single reading of a data base file. Retrieve performs the following operations:

- Reads the retrieval/report specifications from the standard input device (M:SI).
- Transmits the report specifications to the Report program.
- Interprets the retrieval search and extraction specifications.
- Reads file DICT, which describes the file from which the requested information is to be extracted.
- Generates a set of routines that satisfy the request.
- Formats the intermediate output file records and transmits these formats to the Report program.
- Executes the retrieval routines, which extract the requested information from the selected records of the request file.
- Writes the extracted data into an intermediate output file for further processing by the Report program or by a user program.
- Saves request specifications in the Manage request library, if desired.
- Obtains a standard request from the request library, if specified.

The Manage user controls the retrieval function by specifying the following:

1. Name of the file which contains the required information.
2. Criteria to be used in determining those records in the request file from which information is to be extracted.
3. Information to be extracted from the selected records.

Additionally, the user may specify whether he wishes the extracted information to be formatted and listed as hard copy, or whether he requires an intermediate file for further processing by a non-Manage program.

The user may specify that data from two files is required to satisfy his request. This procedure is allowed when the two files have the same key (at least to the level of the file with the fewest key fields), and permits positive association of related records.

File matching is subject to the following constraints:

1. Both files must have the same key structure to the level of the file with the fewest key fields, as in,

Primary File		Secondary File	
Key	Field Name	Key	Field Name
1	Div	1	Division
2	Location	2	Location
3	Shop Number	3	Shop Number
4	Equipment Class		

2. Key fields at the same level must be of the same data type, that is, both binary, alphabetic, or packed decimal.
3. The sequence of both files must be the same, that is, both ascending or both descending.
4. Key fields at the same level are not required to have the same field name or length. If the lengths are different, the following comparison procedure will be followed:
   a. Alphabetic fields will be matched from left to right, using the length of the shorter of the two fields.
   b. The shorter of two numeric fields will be left zero filled and matched to the length of the longer field.

Record selection criteria are presented to the Retrieve program as search specifications. A record is selected when the criteria are satisfied. Specifications may range from a "bypass search", which will yield information from every record in the file, to the specification of a unique key value criterion, which is satisfied by only one record in the entire file.

Output specifications name the field(s) whose contents are required by the user. It is possible to request the contents of one field in a single record in the file, the contents of every field in every file record, or anything between these two extremes.

SELECTION CRITERIA

The user describes his record selection criteria as a list of two-term relationships separated by one of the logical operators AND and OR. A maximum of 20 consecutive AND criteria will be accepted in Retrieve in any given request. Note that an intervening OR criteria restarts this count. A criterion is specified as a relationship between a field in the file, and one of the following:

1. A different field
2. A constant
3. The result of an evaluated arithmetic expression.

The relational operators that may be used are given below.

Operator	Meaning
EQ	Equal to
LS	Less than
GR	Greater than
NE	Not equal to
GE	Greater than or equal to
LE	Less than or equal to

Numeric fields that are being compared must be of the same data type, that is, both binary or both packed decimal.

ARITHMETIC EXPRESSIONS

An arithmetic expression consists of two terms connected by an add, a subtract, a multiply, or a divide operator. The terms may be

1. Two separate fields
2. Evaluated results from two previously specified expressions
3. A field and an expression result
4. A field and a constant
5. An expression result and a constant.

Note that if two fields are specified, they must be of the same data type, that is, both binary or both packed decimal. Expressions are evaluated and saved to be used as search criteria or in subsequent expressions and may be included in the Retrieve output data. The size of a saved expression is always set at 15 digits plus an algebraic sign if it is packed decimal, or 64 bits if it is binary. Some sample arithmetic expressions are given below.

First Term	Arithmetic Operator	Second Term	Arithmetic Result Name
Field 1	+	Field 2	xxx
Field 3	-	xxx	yyy
Field 4	*	+ 15.7	zzz
yyy	/	+37	nnn

Although an expression consists of an arithmetic operation involving only two terms, it is possible to build up expressions of almost any desired complexity by successive iterations of this process. For example, to evaluate

$$X = A + \frac{B+2C}{D} - E$$

set up the problem as follows.

First Term	Arithmetic Operator	Second Term	Arithmetic Result Name
C	*	+2	T_1
B	+	T_1	T_2
T_2	/	D	T_1
T_1	-	E	T_2
A	+	T_2	X

where X, T_1, and T_2 are internal storage fields to be defined at run-time by the Retrieve processor.

Arithmetic expression results are made available for all subsequent comparisons and may be included in the retrieved output data.

Retention of fractional significance (number of decimal places) in arithmetic-expression results is as follows:

1. Addition and subtraction: the result is carried to the greatest number of decimal places in either term.
2. Multiplication: the result is rounded to the greatest number of decimal places in either term.
3. Division: the result is truncated to the greatest number of decimal places in either term.

Since arithmetic expressions may be randomly interspersed with the criteria in a request, it is not always apparent which expressions will be evaluated under a given set of conditions. Retrieve always begins with the first line of a request and processes the following lines in consecutive order. However, the request contains implied branch points that are governed by the contents of the record being examined. There are two conditions that cause portions of the request to be ignored. They are:

1. A comparison that is true, that is, one that satisfies the conditions imposed by the user, and that has OR as a logical connector. In this case, the record will be selected, and all subsequent request lines will be ignored. In the example of Figure 7, lines 01 and 02 are always evaluated, but if line 03 is true, all further lines will be ignored for that data record.
2. A comparison that is false, that is, one that does not satisfy the user's conditions, and that has AND as a logical connector. In this case, the next line to be evaluated or compared will be the line following the next OR connector, if there is another OR in the request. If there are no more OR connectors before the END connector, the record will be rejected. In the example of Figure 8, if line 01 is false, then lines 02 and 03 are inconclusive and are skipped. Line 04 will be compared but may be in error since NEW % may not have been evaluated.

As long as neither of these conditions obtains, evaluation of arithmetic expressions and examination of comparison criteria will continue in sequence until the END connector is reached.

SDS

SIGMA MANAGE RETRIEVAL SPECIFICATIONS

REQUEST NO. C

AUTHOR KOPITO

DATE JULY, 1969

LINE NO.	FIRST TERM: FIELD OR ARITHMETIC RESULT NAME	OPER: COMP OR ARITH OPER	OPERAND: FIELD OR ARITH. RESULT NAME/LITERAL	OPERAND: LITERAL CONSTANT VALUES: ARITHMETIC RESULT NAME	LOG. CON.: AND OR EXP END
0,1	NEWHIRE	+	+.0,0,0,	NEW0,0,0,	EXP
0,2	ONBOARD	+	+.0,0,0,	TOT0,0,0,	EXP
0,3	OPENREQ	EQ	+0.		OR
0,4	NEW0,0,0,	/	TOT0,0,0	NEW%	EXP
0,5	DIVISION	EQ	+19.,+20.,+25.		END

Figure 7. Sample Comparison with OR Connector

SDS

SIGMA MANAGE RETRIEVAL SPECIFICATIONS

REQUEST NO. C

AUTHOR KOPITO

DATE JULY, 1969

LINE NO.	FIRST TERM: FIELD OR ARITHMETIC RESULT NAME	OPER: COMP OR ARITH OPER	OPERAND: FIELD OR ARITH. RESULT NAME/LITERAL	OPERAND: LITERAL CONSTANT VALUES: ARITHMETIC RESULT NAME	LOG. CON.: AND OR EXP END
0,1	OPENREQ	EQ	+0.		AND
0,2	NEW0,0,0	/	TOT0,0,0	NEW%	EXP
0,3	DIVISION	EQ	+19.,+20.,+25.		OR
0,4	NEW%	GR	+.25,0		END

Figure 8. Sample Comparison with AND Connector

DATA FORMATS

Some of the significant characteristics of the data base file from which data can be retrieved are given below.

1. Variable-length records are permitted. A variable-length record differs from a fixed-length record in that a variable-length record will contain one or more multiple-entry fields. These fields contain a list of different values which the field has assumed or might assume at different times or under different conditions.

2. Data base fields may be binary integer, packed decimal, or alphabetic (EBCDIC).

3. A binary field must be from 1 to 4 bytes long and must lie within byte boundaries.

4. Variable-length fields are not permitted. A given field is the same size in every record of the file. However, in the case of a multiple-entry field, the fixed-length value may occur a variable number of times.

5. A counter field in the root segment must be specified for every multiple-entry field. This binary field contains the number of occurrences of the multiple-entry field in any given record. One field may serve as a counter for several multiple-entry fields.

CALLING RETRIEVE

The Retrieve program uses the Batch Processing Monitor for file services and for overlay loading, and Dictnary to determine the format of the request file. Retrieve passes those parameters that describe the report format to Report, as well as the intermediate file data and intermediate file format information.

Input for a Retrieve run consists of the following cards:

1. Retrieve processor call card.
2. General request specification card (type A).
3. Title and heading card (type B).
4. Retrieval specification card (type C).
5. Retrieved output card (type D).

Up to 99 requests (sets of A, B, C, and D cards) may follow the Retrieve processor call card.

After the processor call card, general request specifications are input on card types A and B. Following these may be a list of retrieval specifications (card type C). These specifications will define record selection criteria and arithmetic expressions, if a selective search is required (that is, search mode in column 30 of card type A record is S). Retrieval specifications typically contain:

1. One of the following conditions:
   a. The name of a field, which is a record selection criterion, and a constant or list of constants against which the criterion field value is to be compared.
   b. The name of a criterion field and the name of another field in the record, the contents of which are to be compared against the criterion value.
   c. The two terms of an arithmetic expression, which can be a combination of field names, constants, or the results of previously evaluated expressions.
   d. The name of a previously defined arithmetic expression result, and any other criterion field or constant.
2. One of the following relational operators for a criterion expression:

Operator	Meaning
EQ	Equal to
LS	Less than
GR	Greater than
NE	Not equal to
GE	Greater than or equal to
LE	Less than or equal to

3. One of the following algebraic operators for an arithmetic expression:

Operator	Meaning
+	Add
-	Subtract
*	Multiply
/	Divide

4. One of the following:
   a. A logical operator that connects two criterion statements (AND, OR).
   b. An expression identifier (EXP).
   c. An end identifier (END).

The next input form is the retrieved output card (type D). Although this form contains a great deal of information, the Retrieve program is concerned only with the names of the fields that are to be extracted from the selected records, and the order in which these fields are listed by Report. The order of listing the extracted field names determines the format of the intermediate file record. Each request in a given batch generates its own specific output format. In a batch of 99 requests, the maximum record size that can be listed is 1648 bytes (412 words). The maximum increases by four bytes for each request under 99 in a batch. These specifications are brought in during Retrieve and validated there, so as to maintain a single entry point of parameters into the system.

Every request will undergo a thorough validity check. Diagnostics will be returned to the requestor to indicate such conditions as:

1. Required parameters missing or in the wrong place.
2. A field named in the request not defined for the request field.
3. A field mode inconsistent with its use.

If errors are found in a request, the request is aborted after identifying as many of the errors as possible. If the dictionary for the request file is unreadable, the entire run is aborted.

Formats of these cards are given below. In the following card definitions, an asterisk (*) after column numbers indicates that the field is optional.

The format of the Retrieve processor call card is given below.

```
!RETRIEVE
```

where !RETRIEVE is required as shown, starting in column 1.

GENERAL REQUEST SPECIFICATION CARD

See Figure 9 for a diagram of the general request specification card (type A).

Columns	Description
1-4	Request number: any four characters identifying all the specification records that pertain to a given request.
6	A: identifier for a general request specification card.
8	Library request: an L means that the Manage request library should be searched for specifications previously saved under this request number. The library specifications will constitute the request to be processed. The only changes accepted from the current request specifications will be reporting system parameters (columns 36 to 54).

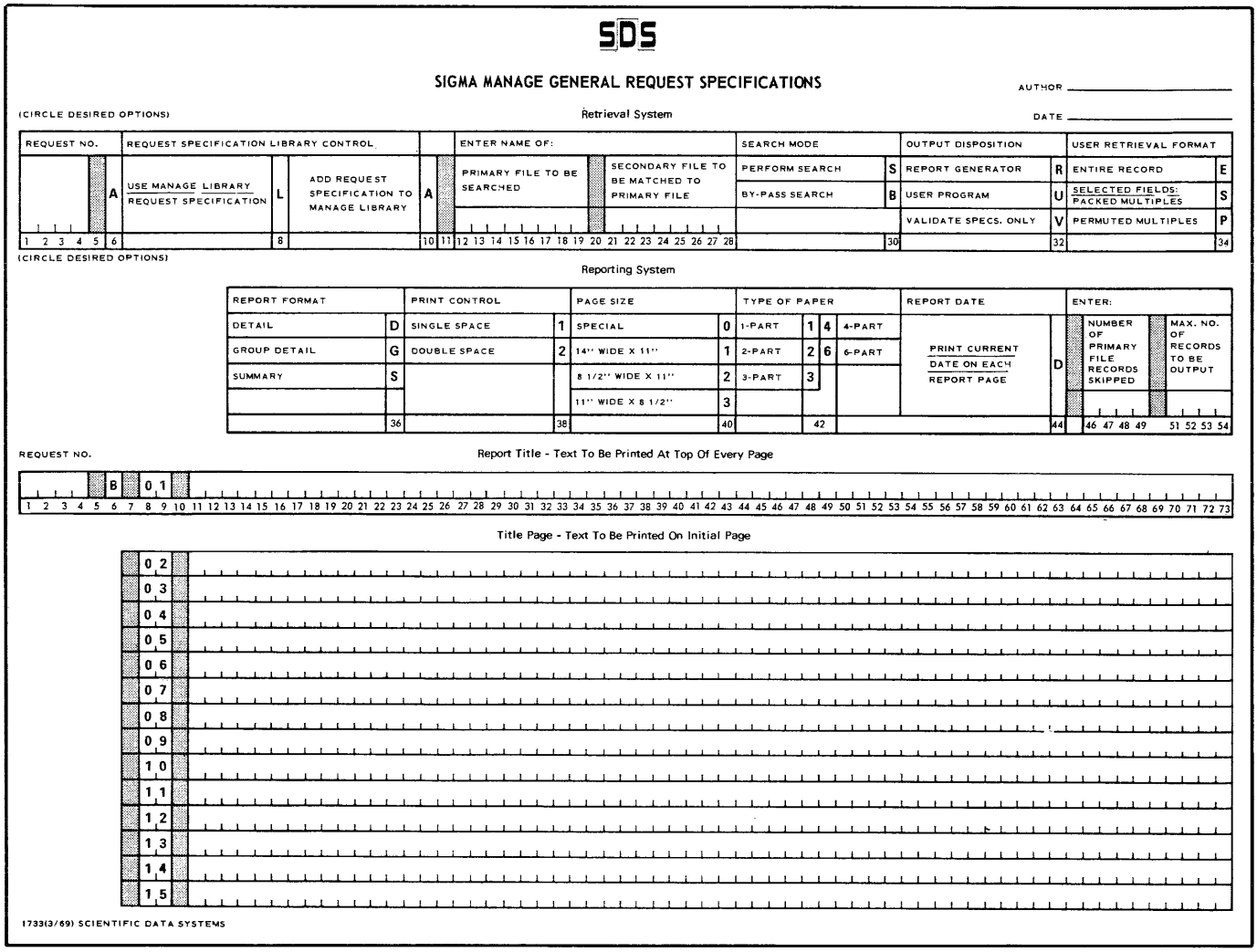


Figure 9. Manage General Request Specifications

Columns	Description
8 (cont.)	The saved request will be checked to ensure that a change to file DICT has not occurred since the card was originally submitted. If a DICT change has occurred, the library copy is obsolete and must be completely resubmitted. If this parameter is left blank, the primary file name (columns 12 to 19) and secondary file name (columns 21 to 28) parameters must be submitted.
10*	Add to request library: an A means that this request should be added to the Manage request library under this request number. If a previous request was saved under this number, it will be replaced. This parameter is ignored if the user intends to validate specifications only.
12-19	Primary file name: an eight-character name of the data base file to be searched. If column 8 is blank, this parameter is required.
21-28	Secondary file name: an eight-character name of a second file to be matched to the primary file. If specified, retrieval will take place only when primary records have a matching secondary. The matched records may be considered as one logical entity, with freedom of selection of any field in either record. Primary field names take precedence over duplicated secondary field names. Note that all requests in a batch either must specify the same secondary file-matching parameter, or must all be blank.
30	Search mode: an S means to selectively search for records meeting the criteria shown in the retrieval specifications (below). B means to bypass the search criteria. Every record will be retrieved according to the output specification (type D). No retrieval specifications may be included. If column 8 is blank, this parameter is required.
32	Disposition of output: an R means that the retrieved data is to be sent to the Report program. U specifies that retrieved records are to be written in a separate file for processing by a user program. Only one request per batch in Retrieve may specify this option. V means validate the request specifications, but bypass the actual execution. If column 8 is blank, this parameter is required.

Columns	Description
34*	User program retrieval format: one of the codes shown below.

Code	Meaning
E	Output the entire record in its original form to a user file for own-code manipulation.
P	Permute multiple-entry groups. One record is created in the output file for every occurrence in each group. Furthermore, each occurrence is created in an output record with all other occurrences in all groups to be retrieved.
S	Output selected fields in compact form. (This is the option used for files with no multiple-entry groups.) The number of output records will be equal to the highest number of occurrences in any multiple-entry group, or a minimum of one record if no groups are involved. When all the occurrences in a shorter group have been output, a null value will be used until the longest group has been exhausted.

Columns	Description
36*	Report format: one of the codes shown below.

Code	Meaning
D	Print detail records (that is, all records retrieved).
G	Print grouped detail lines (that is, the key fields for the first in the group). Break control fields will print only when their values change.
S	Print summary total lines only.

Columns	Description
38*	Print control: a 1 means that the report is to be single spaced, with double-spaced totals. A code of 2 means that the report should be double-spaced, and that the totals should be triple-spaced.
40*	Page size: one of the codes shown below.

Code	Page Size	Print Positions	Lines Per Page
1	14 inches wide, 11 inches long	132	52
2	8-1/2 inches wide, 11 inches long	85	52
3	11 inches wide, 8-1/2 inches long	108	38
0	Special form, to be mounted at list time. Print positions and total lines are entered from the operator's console.		

Reports are listed in order by page size, paper type, and request number.

Columns	Description
42*	Type of paper: a code of 1, 2, 3, 4, or 6 specifies that the report be printed with other requests using a 1-, 2-, 3-, 4-, or 6-part form. No code is required for a special form (see page size above). Note that those reports that require the highest number of copies of a given page size are printed first.
44*	Report date: a D means that the current date (in the form mon dy 'yr) is to be inserted in a heading line at the top of each page.
46-49*	Number of records to skip: a four-digit number of primary file records to be skipped before this request becomes active. Each request in a Retrieve batch may specify a different value.
51-54*	Maximum number of output records: a four-digit maximum number of records to be written in the output file for this request.

TITLE AND HEADING CARD

The title and heading card (type B) is shown in Figure 9. This card is optional.

Columns	Description
1-4	Request number: must be the same four-character identification that appeared in the general request specification card (type A) above.
6	B: identifier for a title and heading card.
8-9	Line number: a code of 01 identifies a single line of text (maximum of 63 characters) to be centered at the top of every page except the preface. A code of 02-99 identifies and sequences up to 98 lines of text to be printed on separate pages prefacing the request listing. The request form is preprinted with 02 to 15. If additional prefacing text is required, numbering should continue consecutively (16, 17, etc.).
11-73	Text: text lines to be used as a page heading or a report preface.

RETRIEVAL SPECIFICATION CARD

The retrieval specification card (type C) is shown in Figure 10.

Columns	Description
1-4	Request number: must be the same four-character identification that appeared in the general request specification card (type A) above.
6	C: identifier for a retrieval specification card.
8-9	Line number: a code of 01-99 sequences lines of retrieval criteria and arithmetic expressions. The retrieval search operation and expression evaluation are carried out in line-number sequence. The form is preprinted with 01 to 25. Should a single search specification require more than 25

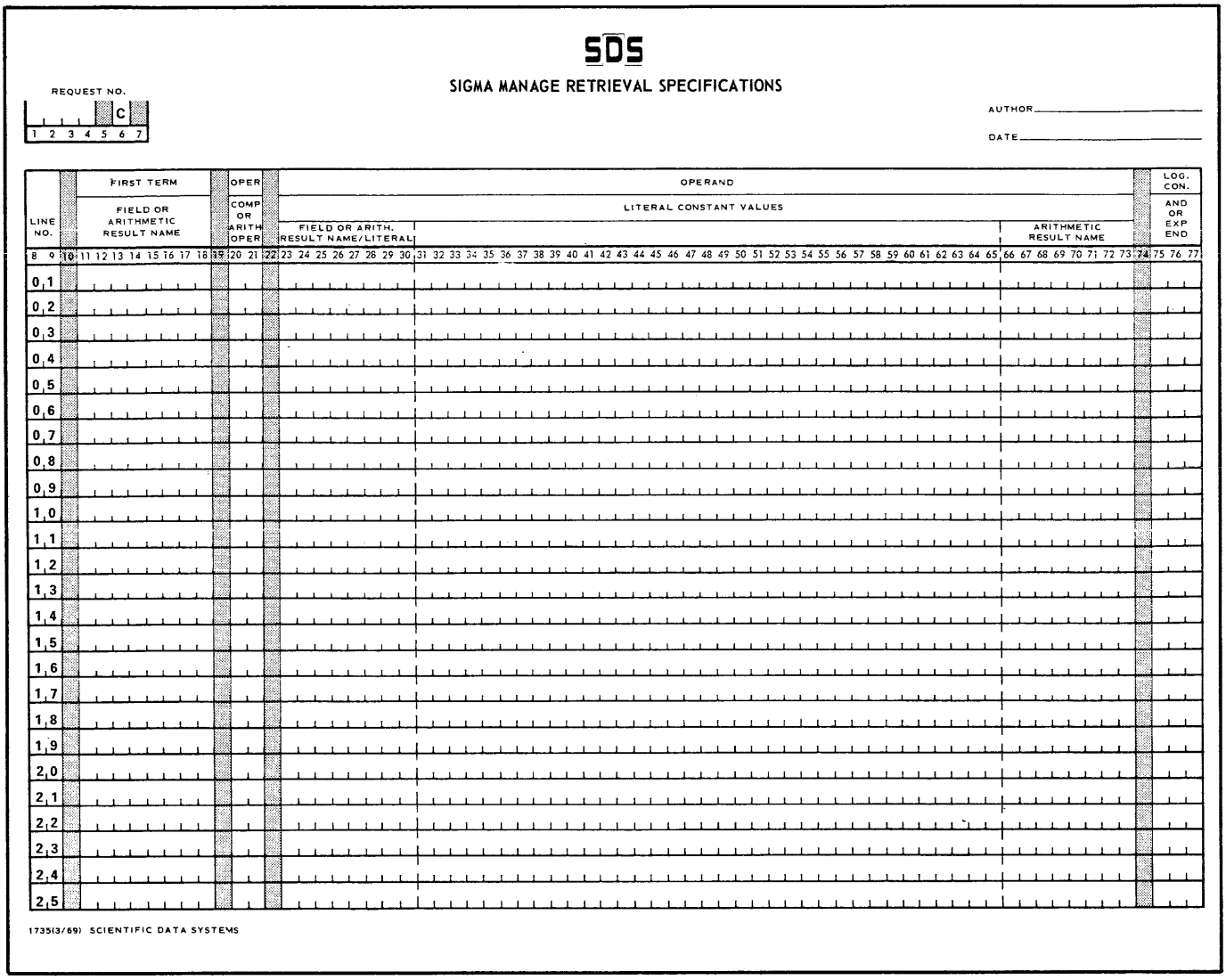


Figure 10. Manage Retrieval Specifications

Columns	Description
8-9 (cont.)	lines, numbering should continue consecutively (26, 27, etc.).
11-18	First term: an eight-character name of the data base field (either primary or secondary) to be acted upon, or the name assigned to the result of a previously defined arithmetic operation result.
20-21	Operator: the logical or arithmetic function to be applied to the value defined by the first term name.
23-30	Field or expression result name: see the description of first-term name above.
or	
23-73	Literal constant: one or more literal values. An arithmetic expression may specify only one numeric literal. A retrieval criterion may specify multiple literals separated by commas. Multiple literals must all be of the same type and number of digits, with the same decimal scaling (if applicable).

Columns	Description
23-73 (cont.)	If the first term parameter is a packed decimal field, a numeric literal consists of a plus or minus sign, a maximum of 15 digits, and a decimal point. For example, +385. and -16.7 are literal constants. If the first term is a binary field, a numeric literal consists of an algebraic sign and a maximum of 10 digits (maximum value is $2^{31}-1$). (Note that under these conditions, a numeric literal may not contain a decimal point.) If the first term parameter is alphabetic, an alphabetic literal will consist of an equal sign followed by up to 49 characters and a trailing comma, as in =ABC,. Literals must be alphabetic if the first term parameter is alphabetic.
66-73	Arithmetic expression result name: an eight-character name to be assigned to the result of an arithmetic expression. The resultant value takes on the computational characteristics of the first term parameter, and may be treated as a data base field on all subsequent specification

Columns	Description
66-73 (cont.)	lines. The size of the result is always 15 digits plus algebraic sign if decimal, or 64 bits if binary.
75-77	Logical connector: one of the connectors given below.

Connector	Function
AND	Connects a given specification line (n) and the following line (n+1) with a logical AND operation. Both n and n+1 must be "true" to satisfy the retrieval criteria.
OR	Connects line (n) and line (n+1) with a logical inclusive OR. That is, the criteria are satisfied if either specification is true, or if both are true.
EXP	Identifies an arithmetic expression specification.
END	Identifies the last retrieval specification card in a request.

Note that the field from columns 23 to 73 is known as the operand field.

RETRIEVED OUTPUT CARD

The retrieved output specification card (type D) is shown in Figure 11.

Columns	Description
1-4	Request number: must be the same four-character identification that appeared in the general request specification card (type A) above.
6	D: identifier for the retrieved output specification card.
8-9	Line number: a code of 01 to 99 that controls the format of the Manage output. Both the retrieved records and the listed report will maintain data fields in the sequence determined by the line number. A field that is entered on line 01 will be the leftmost field on the report, and so on.
11-18	Field name, numeric literal, or expression line numbers: if the parameter is a field name, Retrieve

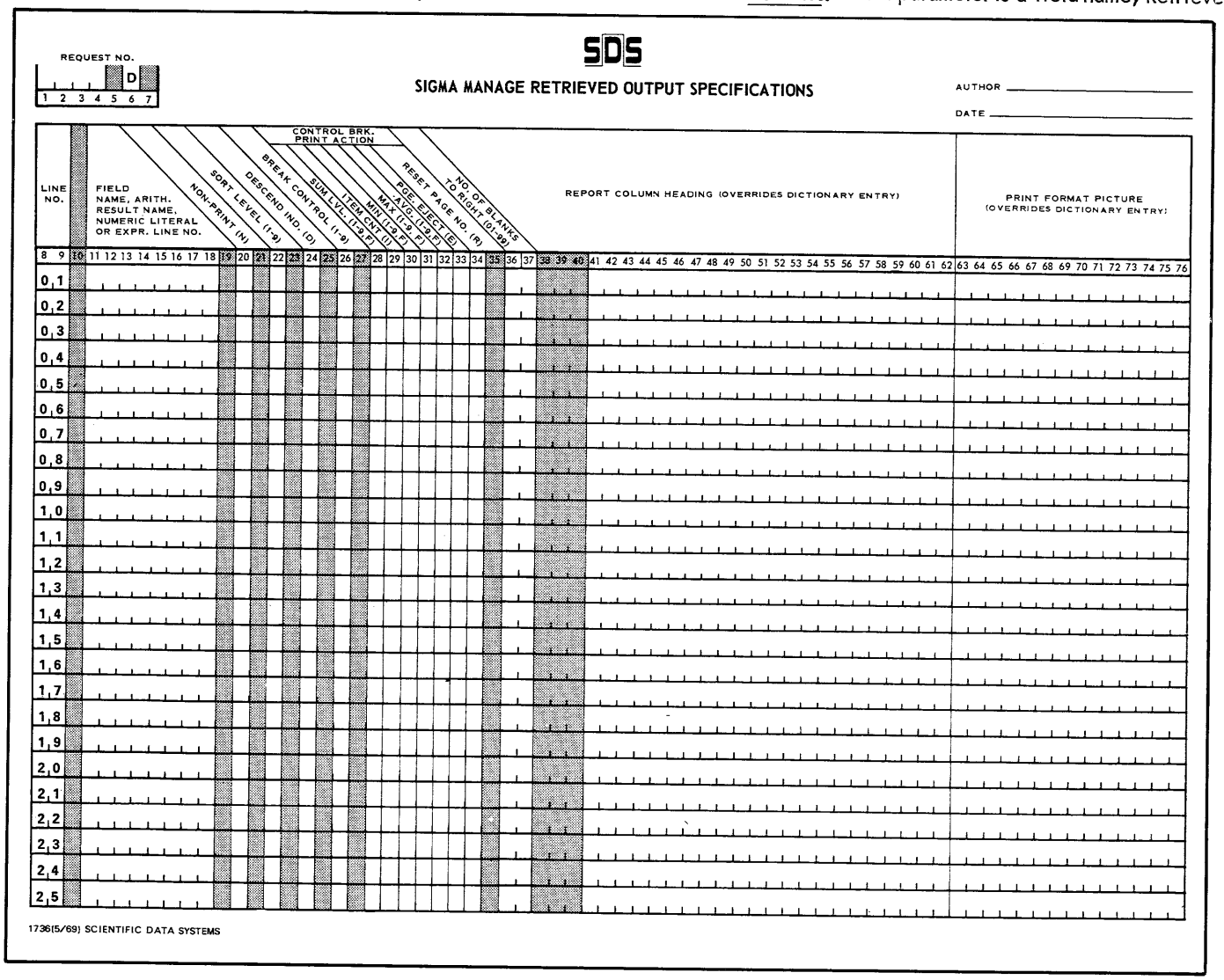


Figure 11. Manage Retrieved Output Specifications

Columns	Description
11-18 (cont.)	will extract the appropriate value and place it in the output record. It will be listed by Report unless specifically suppressed. Any following arithmetic expressions may reference the output field by its line number. A numeric literal consists of an algebraic sign followed by up to seven digits, or by up to six digits and an embedded decimal point. A numeric literal does not create an output field and will not be listed, but any following arithmetic expressions may use it at list time by referencing its line number. An arithmetic expression consists of a two-digit line number, an arithmetic operator (+, -, *, /), and a second line number, with no embedded blanks. The two line numbers must reference a previously-defined output field or numeric literal.
20	Nonprint indicator: an N suppresses the printing of the output field or arithmetic expression defined on this line.
22	Sort level: a number from 1 to 9, or blank. If nonblank, it identifies this field as a report sequence field and specifies the level of sort. The highest level is 1 and the lowest is 9. If the designated report sequence fields match the data base keys, or if no sort level is found, the report is to be listed in the original file sequence and the Sort program will be bypassed. Note that the designation of sort fields is independent of the designation of control break fields.
24	Descending sort indicator: a D means that the report requence field should be sorted in descending order.
26	Total break control: a number from 1 to 9, or blank. The named field is to initiate the designated level of totals whenever it changes in value. The highest-level total break control value is 1. For example, a 3 means that the field is to be used as the control field for 3rd level totals or group counts. When the information in the field changes, 3rd level totals or group counts will be printed out for all total fields indicated as summing level 3 (see below), and also on all lower-level total fields. (Higher-level control breaks always force lower totals to be printed.) The specified level of total break control may be assigned to only one field in the output specification. The value field must be a data base field or a retrieval arithmetic result field. If no total breaks are specified, the report will be a simple list with final totals only. The total byte length of all fields used for total breaks may not exceed 255 bytes. Note that the designation of break control fields is independent of the designation of sort fields. Regardless of any sorting, a change in value of a break control field from one retrieved record to another causes the appropriate level (and all lower levels) to become "active". All sums, item counts, minimums, maximums, and averages tied to each active level are then printed.
28	Summing level: a number from 1 to 9, F, or blank. The named field or expression result is totaled and associated with nth level total breaks (F = Final Total). Up to 15 summed fields may be displayed. For example, when a 4th level total break is taken, a total will be printed out for the field named, if summing level 4 is indicated in column 28. Several fields can be designated as summing level 4; totals for all of them will print out when a 4th level break is taken. Multiple fields to be totaled may be associated with different break levels. For example, fields 9, 10, and 13 could be associated with 2nd level breaks, while fields 8 and 12 could be associated with 3rd level breaks. Fields 9, 10, and 13 would be indicated as summing level 2, and fields 8 and 12 as summing level 3. (The 2nd level break would also force the totals on fields 8 and 12; higher levels force lower level totals.) However, 9, 10, and 13 would be printed only on level 2 or higher.
29	Item count: an I indicates that the field specified by field name is one for which the user wants an item count. If an I is present, there must be an entry in column 26. When a total break is taken, a count of the number of items in the group will be printed. A blank in the column indicates that no group count is to be taken. Regardless of group count or any other control feature, an overall item count is produced automatically as a final total on every report.
30	Minimum value: a number from 1 to 9, F, or blank. The minimum value in the named field or expression result is retained and printed with the specified level of total break. Up to nine minimum values may be displayed.
31	Maximum value: a number from 1 to 9, F, or blank. The maximum value in the named field or expression result is retained and printed with the specified level of total break. Up to nine maximum values may be displayed.
32	Average value: a number from 1 to 9, F, or blank. The average value of the named field or expression result is calculated and printed with the specified level of total break. Up to nine average values may be displayed.
33	New page eject: an E, in conjunction with a total break control entry in column 26, causes the printer page to advance when a control break (change in value) occurs. The advance occurs after printing of the totals or group counts. If column 33 is left blank, page ejection is controlled by line count in the Report program.

Columns	Description
34	Page number reset: an R, in conjunction with a new page eject specification (column 33 above), causes page numbering to restart at 1.
36-37	Number of blanks to right: a number from 00 to 99. The specified number of print positions (to a maximum of one complete print line) will be skipped before the next print item is displayed. If blank, one space will occur before the next item is displayed.
41-62	Column heading: up to 22 EBCDIC characters to be used as a column heading for this report (overrides the standard dictionary heading). The character > (greater than) will indicate a new heading line.
63-76	Print format picture: controls the preparation of data for printing. This specification is required for all arithmetic expression results that are listed. It is optional for fields named in the dictionary and, if specified, overrides dictionary formatting control. The picture consists of a series of code characters that specify the following information: 1. The number and kind of characters that will appear in the print area assigned to this item. 2. Decimal scaling, zero suppression, comma insertion, negative sign insertion, and dollar sign insertion. Code characters that are used in a picture depend upon the nature of the data field being formatted as outlined below. Alphabetic fields 1. An A specifies that the corresponding print position is to be occupied by a character from the data field. 2. Any other specified character is to be inserted in the corresponding print position. For example, assume a data field JUN0168, and a picture of AAAbAAbAA. The display would be JUN 01 68. Numeric fields (packed decimal and binary) 1. Z specifies zero suppression. 9 specifies that the corresponding print position is to be occupied by a number from the data field. B specifies blank insertion. 0 specifies zero insertion. . specifies decimal point insertion and termination of zero suppression. , specifies comma insertion. - specifies trailing minus if negative sum. $ specifies dollar sign insertion to the left of the high order position of the field. 2. The rules of print editing via picture control are the same as those used in SDS COBOL (other than dollar sign insertion).

SAMPLE DATA RETRIEVAL AND REPORT

The personnel director of Amalgamated Corporation must supply the State Department of Employment with a listing of all noncitizen female employees. The report is to be ordered by name, with the most recent employee listed first. The report must show name, address, marital status, social security number, and starting pay rate. An averate of the starting pay rate for each year must also be calculated. Figure 12 shows the required general specifications, while Figures 13 and 14 show retrieval criteria and report specifications, respectively. The resultant report is shown in Appendix D, Figure 15.

RETRIEVE MESSAGES

The following messages are output during Retrieve. A printout of the incorrect record precedes the message.

ABNORMAL OPEN ERROR

The request pile could not be opened. The run is aborted.

ARITHMETIC OPERATOR USED WITH ALPHA FIELD

No arithmetic is permitted on alphabetic (EBCDIC) fields. The card image is printed and the remainder of the card ignored. The request will not be executed.

CARD TYPE ERROR

The specification card type is not A, B, C, or D. The card image will be printed with the message and the request will be rejected after validation.

DATA BASE RECORD TOO LARGE FOR AVAILABLE MEMORY

The dynamic memory allocation algorithm was not able to assign sufficient space to the data base record input buffer to hold one full data base record. A minimum of 1024 words will be assigned to the input buffer under the least dynamic memory availability. The run will be aborted.

DESCENDING SORT LEVEL MISSING

The retrieved output specifications call for a report to be sorted in descending order on a field without specifying the sort level of that field. The request will be rejected after validation.

SDS

SIGMA MANAGE GENERAL REQUEST SPECIFICATIONS

AUTHOR KOPITO

DATE JULY, 1969

(CIRCLE DESIRED OPTIONS)

Retrieval System

REQUEST NO.	REQUEST SPECIFICATION LIBRARY CONTROL			ENTER NAME OF:		SEARCH MODE	OUTPUT DISPOSITION	USER RETRIEVAL FORMAT
SDEM	A USE MANAGE LIBRARY REQUEST SPECIFICATION	L ADD REQUEST SPECIFICATION TO MANAGE LIBRARY	A	PRIMARY FILE TO BE SEARCHED: MASTER	SECONDARY FILE TO BE MATCHED TO PRIMARY FILE	PERFORM SEARCH (S) / BY-PASS SEARCH B	REPORT GENERATOR (R) / USER PROGRAM U / VALIDATE SPECS. ONLY V	ENTIRE RECORD E / SELECTED FIELDS: PACKED MULTIPLES (S) / PERMUTED MULTIPLES P

(CIRCLE DESIRED OPTIONS)

Reporting System

REPORT FORMAT	PRINT CONTROL	PAGE SIZE	TYPE OF PAPER	REPORT DATE	ENTER:
DETAIL (D)	SINGLE SPACE (1)	SPECIAL 0	1-PART (1) 4-PART	PRINT CURRENT DATE ON EACH REPORT PAGE (D)	NUMBER OF PRIMARY FILE RECORDS SKIPPED
GROUP DETAIL G	DOUBLE SPACE 2	14" WIDE X 11" 1	2-PART 2 6-PART		MAX. NO. OF RECORDS TO BE OUTPUT
SUMMARY S		8 1/2" WIDE X 11" 2	3-PART 3		
		11" WIDE X 8 1/2" (3)			

REQUEST NO.

Report Title - Text To Be Printed At Top Of Every Page

SDEM	B	01	NONCITIZEN FEMALE EMPLOYEES

Title Page - Text To Be Printed On Initial Page

02	REPORT TO STATE DEPARTMENT OF EMPLOYMENT SHOWING ALL NONCITIZEN
03	FEMALE EMPLOYEES.
04	
05	REPORT IS SEQUENCED BY:
06	HIRE YEAR (DESCENDING)
07	HIRE MONTH (DESCENDING)
08	NAME
09	
10	THE REPORT SHOWS VARIOUS ITEMS OF DATA FOR EACH EMPLOYEE,
11	INCLUDING STARTING RATE OF PAY.
12	
13	THE AVERAGE STARTING RATE OF PAY FOR EACH YEAR IS ALSO SHOWN.
14	
15	

1733(3/69) SCIENTIFIC DATA SYSTEMS

Figure 12. Sample General Request Specifications

SDS

SIGMA MANAGE RETRIEVAL SPECIFICATIONS

REQUEST NO. SDEM C

AUTHOR KOPITO

DATE JULY, 1969

LINE NO.	FIRST TERM: FIELD OR ARITHMETIC RESULT NAME	OPER: COMP OR ARITH OPER	OPERAND: FIELD OR ARITH. RESULT NAME/LITERAL	OPERAND: LITERAL CONSTANT VALUES / ARITHMETIC RESULT NAME	LOG. CON.: AND OR EXP END
01	SEX	EQ	=F,		AND
02	CITIZEN	EQ	=A,		END

Figure 13. Sample Retrieval Specifications

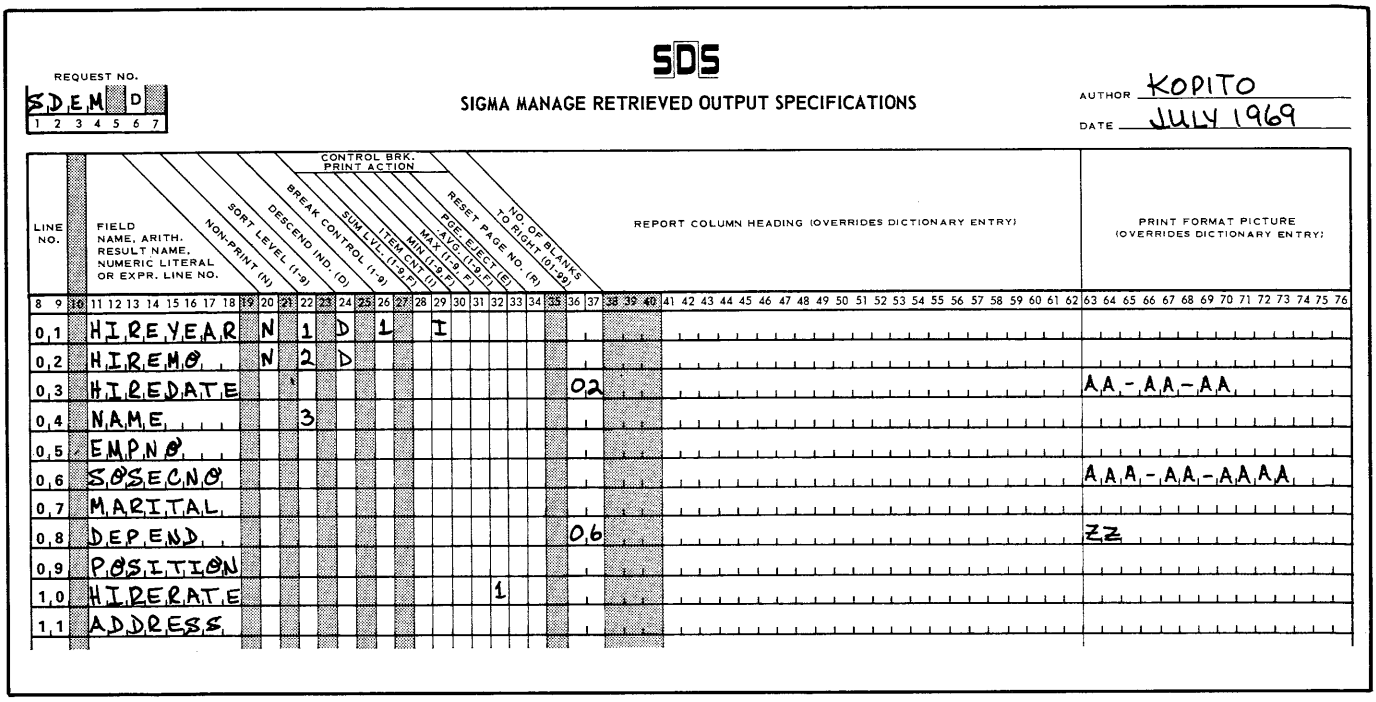


Figure 14. Sample Report Specifications

DICTIONARY READ FAILURE

The dictionary for the request file is unreadable. The run is aborted.

DUPLICATE LINE NUMBERS

The retrieved output specification line numbers are not unique. The card image on which the duplicate line numbers was detected will be printed and the request will be rejected after validation.

DUPLICATE SORT OR CONTROL BREAK LEVEL

The sort or control break level is duplicated on another report specification.

EXIT TABLE/DOUBLEWORD OVERFLOW. CUT OFF AT xxxx

The combined exit table (EXITAB) and doubleword table (DBLWD) space requirements exceed the memory allocation for these two tables. Code generation will be terminated and all requests prior to the one that caused the overflow will be executed.

EXPRESSION CONTAINS 2 LITERALS

Both line numbers in the arithmetic expression refer to literals. The expression is rejected.

EXPRESSION TERM NOT NUMERIC FIELD OR LITERAL

An expression line number refers to a retrieved output specification line that does not specify either a numeric data base field, a numeric arithmetic result field, or a numeric literal. The request will be rejected after validation.

EXTRANEOUS DATA IN EXPRESSION FIELD

An expression in a retrieved output specification may contain only two digits (0-9), an arithmetic operator (+, -, *, /), and two more digits. This message is printed along with the card image if the remainder of the field (columns 16 to 18) is not blank. The request will be rejected after validation.

FIELD NAMED NOT DICT NAME OR ARITH RESULT FIELD

One of the fields on the card was not listed in file DICT as a component field of the request file; nor has it been previously listed by the requestor as a label to be applied to the result of the evaluation of an arithmetic expression. The request will be rejected after the remaining specifications are processed.

FILE SPECIFIED NOT BATCHED REQUEST FILE

A request requires a primary or secondary data base file other than the file or files specified by the first request in the batch. It should be noted that the first request is not necessarily the one that is at the front of the input deck but rather the request that is sequenced first by the Sort program. The general request specification card is printed and the request is rejected with no further processing.

ILLEGAL SORT OR CONTROL BREAK KEY

The report specification calls for a sort or control break on a literal or arithmetic expression.

INSUFFICIENT MEMORY AVAILABLE FOR RETRIEVE

Retrieval will not be attempted without at least 15 pages of dynamic memory. This message indicates that 15 pages are not available. The run will be aborted.

INVALID DESCENDING SORT SPECIFICATION

The descending sort parameter is neither D nor blank.

INVALID DIGIT IN BLANK FILL COUNT

The blank fill count field on the retrieved output specification card was not blank and it contained a nonnumeric character. The rest of the request will be validated but it will not be executed.

INVALID DIGIT IN EXPRESSION LINE NUMBER

A character other than 0-9 was encountered in an expression on a retrieved output specification card. The card image will be printed and the remainder of the request will be validated. The request is rejected.

INVALID DIGIT IN LINE NUMBER

At least one of the two digits in the retrieved output specification line number field (columns 8,9) is not a numeric (0-9). The remainder of the request will be validated but the request will not be executed.

INVALID DIGIT IN LITERAL

An illegal decimal digit has been detected in a numeric literal. The card image is printed and the rest of the card is ignored. The request will be rejected after the rest of the request cards are processed.

INVALID DIGIT IN NUMBER OF RECORDS TO OUTPUT

The maximum number of retrieved output records specified by the user on the general request specification form (columns 51 to 54) contains a nonnumeric character, if it is not blank. The remainder of the request will be validated but it will not be executed.

INVALID DIGIT IN NUMBER OF RECORDS TO SKIP

An illegal character (not 0-9 or all blanks) was found in the general request specification card field stipulating the number of initial primary file records to be skipped or ignored (columns 46 to 49). The remainder of the request will be validated, but it will not be executed.

INVALID EXPRESSION LINE NUMBER

One or both line numbers do not exist, or they contain a nonnumeric character.

INVALID FILE DICTIONARY NAME

No dictionary could be found for the named request file. Either the file name was misspelled or the dictionary has been lost. The entire run is aborted.

INVALID ITEM COUNT

The item count parameter on the above report specification line is neither I nor blank.

INVALID LOGICAL CONNECTOR

A logical comparison is followed by a connector other than AND, OR, or END. The request will be rejected.

INVALID NON-PRINT SPECIFICATION

The nonprint parameter is neither N nor blank.

INVALID NUMERIC LITERAL

A literal in the above report specification is incorrect. Probable causes are:

1. An embedded blank or all blanks.
2. Multiple decimal points.
3. A nonnumeric character.

INVALID OPERATOR

The operator on a retrieval specification card is not one of the following: EQ, NE, GR, LS, GE, LE, +, -, *, /. The card image is printed and the rest of the card ignored. The request will be rejected.

INVALID PAGE COUNT SPECIFICATION

The page count parameter is neither R nor blank.

INVALID PAGE EJECT SPECIFICATION

The page eject parameter is neither E nor blank.

INVALID SIGN IN NUMERIC LITERAL

A numerical literal on a retrieval specification card has a sign character other than + or -. The request will be rejected after validation.

INVALID SORT OR CONTROL BREAK LEVEL

The sort or control break level parameter is neither 1 to 9 nor blank.

INVALID SPEC IN COL. xx

The general request specification is in error in the column specified. The request will be rejected after validation.

INVALID SUM-MIN-MAX-AVG LEVEL

One or more of the above parameters is not a number from 1 to 9 or F.

I/O ERROR xx TRYING TO READ OR WRITE REQUEST xxxx

The Monitor has returned an error type xx upon being asked to read request xxxx from the request library, or to write the request into the library. If the error occurred in trying to read the request from the library, the request cannot be processed. If the error occurred in attempting to write the request into the library, the request will be processed but it cannot subsequently be called from the library.

I/O ERROR xx TRYING TO WRITE RPT SPECS

The Monitor was unable to write a record into the report specification file (RPTSPC) for the reason given (xx). This message will be followed by a JOB ABORTED message, and the run will be terminated.

ITEM COUNTING RESTRICTED TO CONTROL BREAK FIELDS ONLY

An item count was specified for a field that is not a control break key.

LIBRARY REQ ID xxxx REJECTED. DICTIONARY OBSOLETE

The request file DICT and, presumably, the request file itself have been changed since the request was put into the library. It is necessary to resubmit the request deck after making any changes necessary to accommodate the current file structure.

LITERAL SIZE INCONSISTENCY

This message indicates that all literal constants in a multiple literal string do not contain the same number of characters, that a binary constant contains more than 10 digits, or that a packed decimal constant contains more than 15 digits. The card image is printed and the remainder of the card is ignored. The request will not be executed.

LITERAL SPECIFIED AS LEFT TERM

A literal or numeric constant is not allowed as the first or left term of a criterion or arithmetic expression on a retrieval specification (type C) card. The card image will be printed out and the request rejected. The rest of the card will not be examined.

LITERALS EXTEND BEYOND COL.73

The literal is too large. Retrieve will continue validating the remaining request specifications, but the request will not be executed.

MISSING CONTROL BREAK LEVEL

Levels must be ascending, from 1, with no gaps.

MISSING CONTROL BREAK REQUIRED FOR AVG FUNCTION

MISSING CONTROL BREAK REQUIRED FOR MAX FUNCTION

MISSING CONTROL BREAK REQUIRED FOR MIN FUNCTION

MISSING CONTROL BREAK REQUIRED FOR SUM FUNCTION

The designated function was specified to appear at a given control break level, but no control break key has been assigned at that level.

MISSING GENERAL REQUEST SPECIFICATION

The general request specification card (type A) was omitted for this request. The remainder of the specifications for this request will be skipped.

MISSING SORT LEVEL

Levels must be ascending, from 1, with no gaps.

MORE THAN 9 AVG VALUES SPECIFIED

The indicated report limitation has been exceeded.

MORE THAN 20 CONSECUTIVE AND'ED CRITERIA

Retrieve allows 20 successive AND criteria without an intervening OR. If this limit is exceeded, the remaining specifications will be validated, but the request will not be executed.

MORE THAN 9 MAX VALUES SPECIFIED

The indicated report limitation has been exceeded.

MORE THAN 9 MIN VALUES SPECIFIED

The indicated report limitation has been exceeded.

MORE THAN ONE DECIMAL POINT IN DECIMAL LITERAL

The remainder of the request will be validated, but it will not be executed.

MORE THAN ONE GENERAL REQUEST SPEC FOR REQUEST

Only one type A card is allowed per request. The second card will be printed and skipped and the remainder of the request will be rejected after validation.

MORE THAN ONE HEADER SPECIFICATION

Only one page header card (type B, number 01) is allowed for each report. If more than one header card is submitted, the request specifications will be validated, but the request will not be executed.

MORE THAN 15 VERTICAL SUMS SPECIFIED

The indicated report limitation has been exceeded.

MULTIPLE LITERALS IN ARITHMETIC EXPRESSION

An arithmetic operation was specified between a previously-defined field and a list of compound numeric constants. The second term in an expression may contain only one numeric literal. The request will be rejected.

NO RETRIEVAL SPECIFICATION END CARD

No retrieval specification card with END in columns 75 to 77 was found in a request. The retrieved output specifications, if any, will be validated but the request will not be executed.

NO VALID GENERAL REQUEST SPEC IN BATCH

Retrieve will not begin processing specifications until the first general request specification card (type A) is read. This message indicates that there are no general request specifications cards in the request input. The run is aborted.

NO VALID REQUESTS IN THIS RUN

Every request in the batch contains discrepancies. The job is terminated.

NUMERIC LITERAL SCALING ERROR

Multiple packed decimal literals do not all have the decimal point in the same position. The request will be rejected after validation.

OUTPUT SPECS SPECIFIED WITH ENTIRE RECORD OPTION

Retrieved output specifications were included in a request for which "entire record" was specified on the general request specifications. The entire record will be retrieved.

PRIMARY AND SECONDARY FILE KEYS INCOMPATIBLE RUN ABORTED

This message indicates one of three possible conditions:

1. The files to be matched have no common key field.
2. The mode of a key field (EBCDIC, binary, or packed decimal) is not the same in the two files to be matched.
3. The two files are not sorted in the same direction on a common key field. That is, one file is sorted in ascending order and the other in descending order.

If any one of these conditions obtains, the files cannot be matched. The run will be aborted.

REQUEST ID xxxx ABORTED. ARITHMETIC TRAP

When an arithmetic trap is encountered during the execution of any request, this message will be output and the request will be terminated. Any previously-retrieved data, and possibly the current record, will be saved and processed normally either by Report or as a partial user file.

REQUEST ID xxxx AND ALL FOLLOWING NOT DONE. EXECUTION PROGRAM OVERFLOW

Memory allocation for the execution program is not adequate to generate the code necessary to execute all the requests. The requests that do fit are executed.

REQUEST ID xxxx REJECTED. MULTIPLE ENTRY ARITH OFLOW

The 511-word multiple-entry arithmetic-table (MEATBL) capacity has been exceeded. The request will be rejected without further processing.

REQUEST ID xxxx REJECTED. NO REPORT SPECS.

The general request specification card identified the output for this request as a report. No report specifications (retrieved output specifications) were submitted with this request identification. The request will not be executed.

REQUEST ID xxxx REJECTED - NO SEARCH OR RETRIEVE SPECIFICATIONS

Search or retrieve specifications were omitted.

REQUEST ID xxxx REJECTED. PREVIOUS USER FILE REQUEST

Only one request for a nonreport file is permitted in a batch of requests. The first one encountered will be accepted. Any others will be rejected.

REQUEST ID xxxx REJECTED - SPECIFICATION ERROR(S)

This message is printed at the end of an incorrect request. It is preceded by one or more error messages describing the specific errors that caused the rejection.

REQUEST SPEC READ FAILURE

The input/output handler for the specification cards has returned an abnormal or error condition. The rest of the request cards are skipped.

REQUEST SPECIFICATIONS FOR xxxx OVERFLOW AVAILABLE MEMORY. REQUEST REJECTED.

Request xxxx overflowed the available space for the Retrieve working table. This request will be rejected and an attempt will be made to fit the next request into the available space. After all the requests that fit are processed, the file retrieval will be initiated. All the requests listed in this message must be rerun.

REQUEST SPECS UNEXPECTEDLY TERMINATED

The end of the batched input was encountered while processing retrieval specifications, but before the END card was found. This request will not be executed.

RESULT FIELD MODE INCONSISTENT WITH MODE OF TERMS

The user has assigned the same storage location to the result of more than one arithmetic expression evaluation, and all the affected arithmetic expressions are not of the same mode. When the storage location is first assigned, it is given the mode of the terms of the arithmetic expressions. Expressions that use this same result field must agree with it in mode. Any disagreement will cause the request to be rejected after any remaining specifications are validated.

RESULT FIELD NAME SAME AS DATA BASE FIELD NAME

The name assigned to the storage location of the results of evaluating an arithmetic expression (as specified by the user in columns 66 to 73 of the retrieval specifications) is the same as the name of a field in the request data base file. The request will be rejected after validating remaining specifications.

RIGHT AND LEFT TERM MODES DO NOT AGREE

Mixed-mode comparison or mixed-mode arithmetic is not permitted. Both terms in a comparison must be either EBCDIC, packed decimal, or binary. If the right-hand term is either a single or multiple literal constant, the literal format must be appropriate to the left term. That is, if the left term is EBCDIC, each literal constant must begin with an equal sign and end with a comma. If the left term is binary, each literal constant must begin with a plus or minus sign, and may have no embedded blanks or decimal points. If the left term is packed decimal, each literal constant must be preceded by a plus or minus sign and must contain one decimal point and no embedded blanks. The card image will be printed, the rest of the card ignored, and the request rejected.

SPECIFICATIONS FOR LIBRARY REQUEST

Retrieval and/or retrieved output specifications were submitted for a library request. The specifications will be validated, but the request will not be executed.

SPECS FOLLOW 'END' CARD OR BYPASS SEARCH OPTION

Retrieval specifications were included in a request in which "bypass search" was specified. The specifications are ignored.

SUM-MIN-MAX-AVG OF LITERAL OR ALPHA FIELD

None of the above functions may be specified for a literal or alphabetic field.

TOO MANY DIGITS IN NUMERIC LITERAL

A packed decimal literal contains more than 15 digits or a binary literal has more than 10 digits. The request will be rejected after validation.

TOO MANY RESULT FIELDS IN xxxx. REQUEST REJECTED.

An entry is added to file DICT for each arithmetic expression result in a given request. These additional entries are cleared out between requests. If the additional entries overflow the 2048 words allotted to DICT, the request will be aborted.

TOTAL SIZE OF BREAK FIELDS EXCEEDS 255 CHAR

The field above is specified as a sort or break control key. Its length, plus the length of all fields previously designated as keys, is greater than the allowed maximum of 255 characters. The request will be rejected.

UNREADABLE INPUT RECORD

A request file record could not be read. The run is aborted.

5. REPORT

The Report program prepares printed Reports from data extracted by the Retrieve program. Report performs the following functions:

- Reads the Report specification file for descriptions of requests to be listed.
- Reads the Report specification file for format specifications of the records applicable to requests.
- Sets up the required call for Sort if the report is in a sequence different from that of the original data base. Sets up sorting specifications, input/output file characteristics, and identifiers of the records to be selectively sorted. Links to the Sort.
- Builds a set of tables to control the formatting of the desired report.
- Reads the input file (from either Retrieve or Sort) and prepares the required report.
- Determines if all requests have been processed. If yes, ends the job. If any reports remain, returns to the first step in the processing cycle.

The Report program makes use of the following inputs:

1. Retrieved output data files, consisting of:
   a. Report specifications (by request number), and retrieved record format specification (by request number).
   b. Retrieved data records (variable formats, by request number).
2. Column headings and editing information from the dictionary.

REPORT SPECIFICATIONS

Specifications governing the format of the desired report for any given request are passed to Report in a separate file on the RAD. These specifications are of three general types:

1. Report format table: a single variable-length record packed with all the general and line descriptions submitted by the user. This information is used to set up the columnar arrangement of the listing, arithmetic operations, sequencing, and totaling.
2. Report text string: a single variable-length record containing the report heading text and page heading text.
3. Extracted data format table: a single variable-length record which defines the format of the data records extracted for this request.

REPORT CONVENTIONS

The following constraints and conventions apply to the Report program.

HORIZONTAL TOTALING

Horizontal totaling is accomplished by using the arithmetic expression capability. After two fields are initially defined, their line numbers are specified in an additive expression. If more than two fields are to be totaled, the intermediate expression results may be print-suppressed and rolled through as many horizontal additions as required. For example, the specifications below result in the printing of fields 1, 2, and 3, and the horizontal sum of all three (see line 05).

Line No.	Field Name, Arithmetic Result Name, Numeric Literal, or Expression Line Number	Non Print	Print Format Picture
01	Field 1		
02	Field 2		
03	Field 3		
04	01 + 02	N	
05	03 + 04		$9,999.99-

Note that the horizontal sum may be vertically totaled, averaged, or analyzed for minimum/maximum values.

TEXT FIELDS

Output fields defined as text fields (data type T) are assumed to be alphabetic. They may not be used in arithmetic expressions, and may not exceed 255 characters in length if used as a report key. The following procedures apply when printing a text field.

1. Report skips to a new line.
2. The text is indented one inch from left and right margins.
3. Automatic line folding occurs (without hyphenation) if the field exceeds one line.
4. Trailing blank lines are suppressed.
5. The field name is listed to the left of the first line.
6. No column headings are used.

ALPHABETIC FIELDS

Alphabetic fields whose total length cannot be printed on the line being constructed will cause a line feed. Up to one full line will then be printed, with no blank suppression. If the field length still exceeds one line, it will be truncated.

BINARY FIELDS AND EXPRESSIONS

The print width of a binary value is equal to one-third the number of bits, plus one additional character. The maximum value that can be converted and printed is $2^{31}-1$. Print punctuation for data fields is obtained from file DICT. A print picture may be specified for arithmetic expressions, or to override field specifications in file DICT.

FRACTIONAL SIGNIFICANCE

Retention of fractional significance (number of decimal places) in arithmetic-expression results is as follows:

1. Addition and subtraction: the result is carried to the greatest number of places in either term.

2. Multiplication: the result is rounded to the greatest number of decimal places.

3. Division: the result is truncated to the greatest number of places in either term.

CALLING REPORT

The Report program uses the Batch Processing Monitor for file services and overlay loading. Report makes use of output files, created by Retrieve, that contain report/data definitions and extracted data records. If the user elects to use standard headings and printing formats (via the default print layout), Report accesses them from file DICT, which was created by the dictionary generator. Sort is used when the listing sequence differs from the original data base sequence.

Input for a Report run consists of the Report processor call card explained below. The description of the retrieved output card is given in Chapter 4.

The Report processor call card is of the form

```
! REPORT
```

where ! REPORT is required as shown starting in column 1.

After being called, the Report program opens the specification file and reads those specifications pertaining to the first request. The report sequence is compared against the original data base sequence. If they are the same, sorting is not required, and the actual list phase will be entered directly. If the sequence of the retrieved data must be changed, Sort will be called. The Sort program will read in the entire retrieved data file, select the records pertaining to the request being processed, sequence them as required, and then enter the listing phase. If errors are found in a request, it will be aborted and the next listing will be attempted. At the end of each list, the specification file will be examined for remaining requests, and the cycle will be repeated as needed.

REPORT MESSAGES

The following messages are output during Report.

CANNOT FIND REPORT INPUT FILE

The Report program was unable to open the report input file. The file may not exist.

CANNOT READ REPORT SPECIFICATION FILE

An error or abnormal condition has occurred while reading the Report specification file. The file may not exist.

END MANAGE REPORTS-RESTORE STANDARD PRINT FORM

All user-specified special reports have been printed.

HEADINGS FOR MORE THAN 2 DETAIL LINES ARE OMITTED

Not enough headings are present for the total number of detail lines. This message is shown on the first page of the affected report listing. Each retrieved record will be completely listed with headings for the first two lines only.

MANAGE: LOAD x PART FORM – xxx WIDE – xx LINES/PAGE

Operator intervention is required to set up the specified type of paper. (Message appears at operator's console.)

MANAGE: LOAD SPECIAL FORM FOR REQUEST xxxx

Operator intervention is required to set up the special form. (Message appears at operator's console.)

MANAGE REQUEST xxxx FORM SPECIFICATIONS REQUIRED – ENTER 2 DIGIT NUMBER OF LINES

The two-digit line-number parameter is required. (Message appears at operator's console.)

MANAGE REQUEST xxxx FORM SPECIFICATIONS REQUIRED – ENTER 3 DIGIT NUMBER OF PRINT POSITIONS

A three-digit print-position-number parameter is required. (Message appears at operator's console.)

MEMORY OVERFLOW

Either there is insufficient memory to read in the report specification file from Retrieve, or there is no common page for Sort communication.

MISSING FIELD NAME IN FORMAT RCD

A field name has not been transferred properly from Retrieve to Report.

MISSING REPORT CONTROL ENTRY–REQ xxxx

A report specification record key has not been transferred properly from Retrieve to Report.

MISSING REPORT FORMAT RCD – REQ xxxx

A report format record has not been transferred properly from the Retrieve program to Report.

NUMBER OF PRINT LINES PER RCD OVER 255

The request is aborted because the number of print lines exceeds the allowed maximum.

READ ERR ON STANDARD HEADING DICTIONARY

The Report program was unable to read successfully the dictionary record containing the standard column headings.

REQ TERMINATED – ARITH TRAP PROCESSING PRINT ENTRY xx

An illegal arithmetic procedure was detected in the indicated line. The request is aborted and the next request is attempted.

UNABLE TO COMPLETE REPORT – INADEQUATE MEMORY

Available memory space is insufficient for complete processing of the report.

UNREADABLE RCD ENCOUNTERED–REQUEST ABORTED

An unreadable data record has been found in the Report input file. The next request in the batch will be attempted.

APPENDIX A. SAMPLE MANAGE DECK SETUPS

Manage operating examples are given below. The first four describe deck structures required to enter the four Manage processors into the System Library. The remaining five examples describe required deck setups for using the processors. Note that these illustrations do not show any limit cards since they are installation-dependent. The deck setups will also vary for different combinations of RAD, tape, and device assignments. Appendix B contains DCB information necessary when complex input/output assignments are required.

During execution, the Manage processors send various messages to the operator's console, such as tape mount and dismount commands. Required responses are described in the Batch Processing Monitor manual. The only nonstandard operator responses that may be required occur in the Report process, as outlined below.

1. Reports are grouped by paper type and number of copies. At the start of each group, the operator is informed of the form required in the printer, and printing is suspended until the operator keys in the appropriate BPM continue command.

2. Each report requiring a special form causes a suspension of processing until the operator keys in the number of print positions and line depth pertaining to that form. The console key-in procedure is prompted by Report after identification of the request being processed.

The standard Dictnary processor is constructed from the three temporary element files produced by the deck shown in Example A-1.

The deck in Example A-2 produces three element files from which are constructed either the standard Fileup processor, or the Fileup processor with user own-code. This module is callable with an OWNUP card.

The standard Retrieve processor is constructed from the four temporary element files produced by the deck shown in Example A-3.

The standard Report processor is produced by the deck shown in Example A-4.

The deck in Example A-5 illustrates a run that either updates a previous dictionary or creates a new one. The resulting file will contain all dictionaries defined under the current account number.

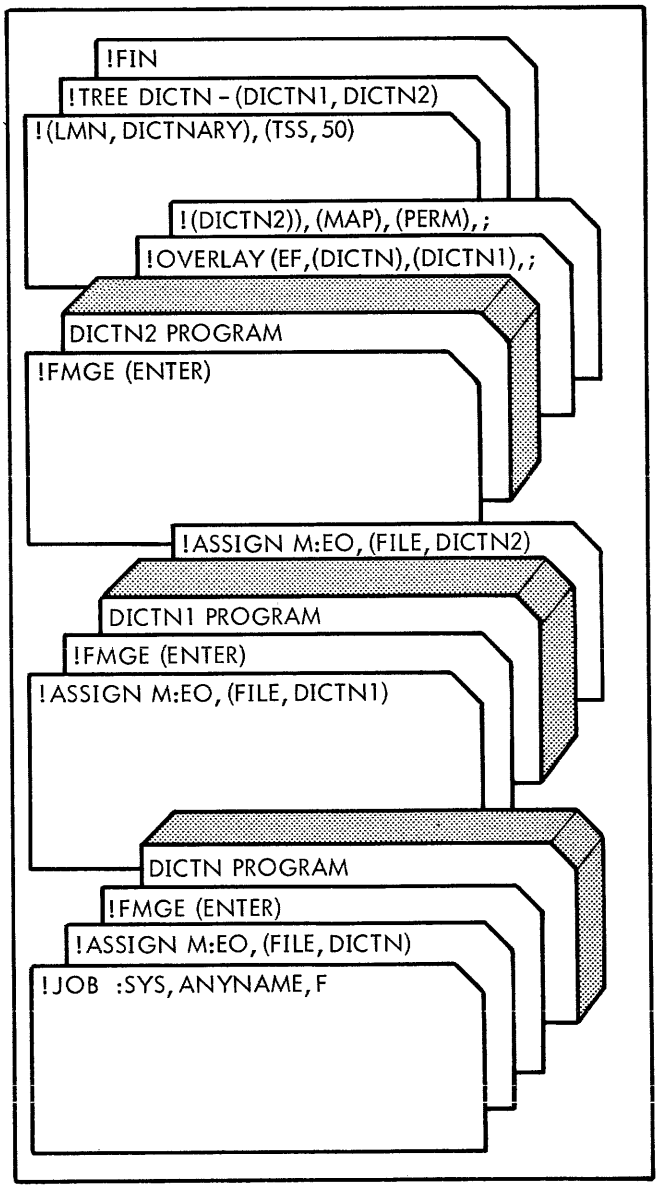
The deck in Example A-6 illustrates the updating of an existing RAD file. The Fileup specifications and the change transactions are on cards. The sorted transaction file UPTRAN is deleted at the end of the update process.

The deck in Example A-7 illustrates the updating of an existing tape file. The Fileup specifications are taken from a library named SPEC, and the change transactions are to be read from tape. The transactions are to be sorted and placed on another tape before the update begins. An audit file is also to be created on tape.

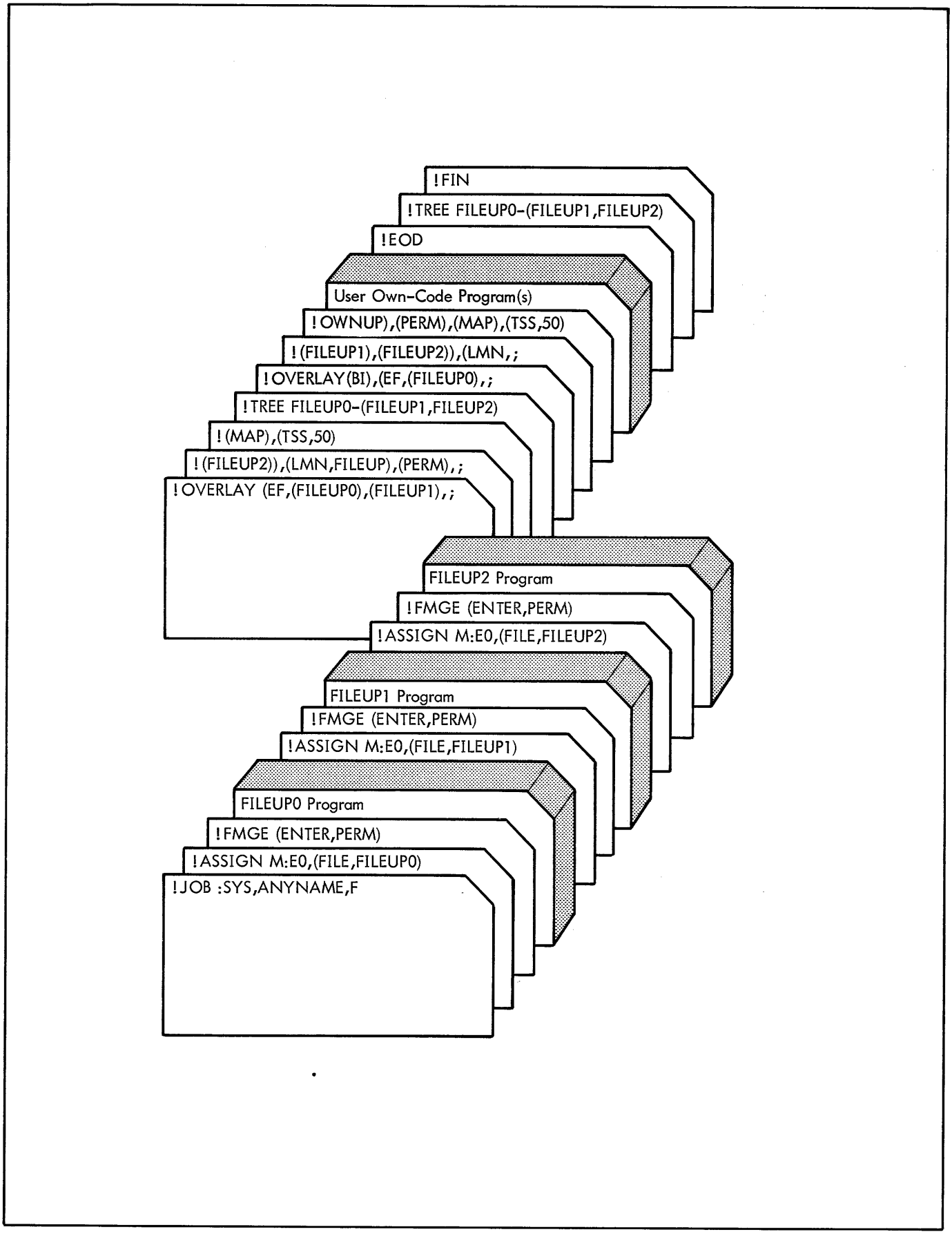
The deck in Example A-8 illustrates a retrieval and report generation run containing one or more requests. The data base is a RAD file, and all intermediate outputs are also defaulted to the RAD.

The deck in Example A-9 illustrates a retrieval run in which all requests require a match between a primary tape file and a secondary RAD file. A user file is to be created on tape, and requests to be listed are on another tape. During the Report process, any required sorting is to be done using 9-track tapes.

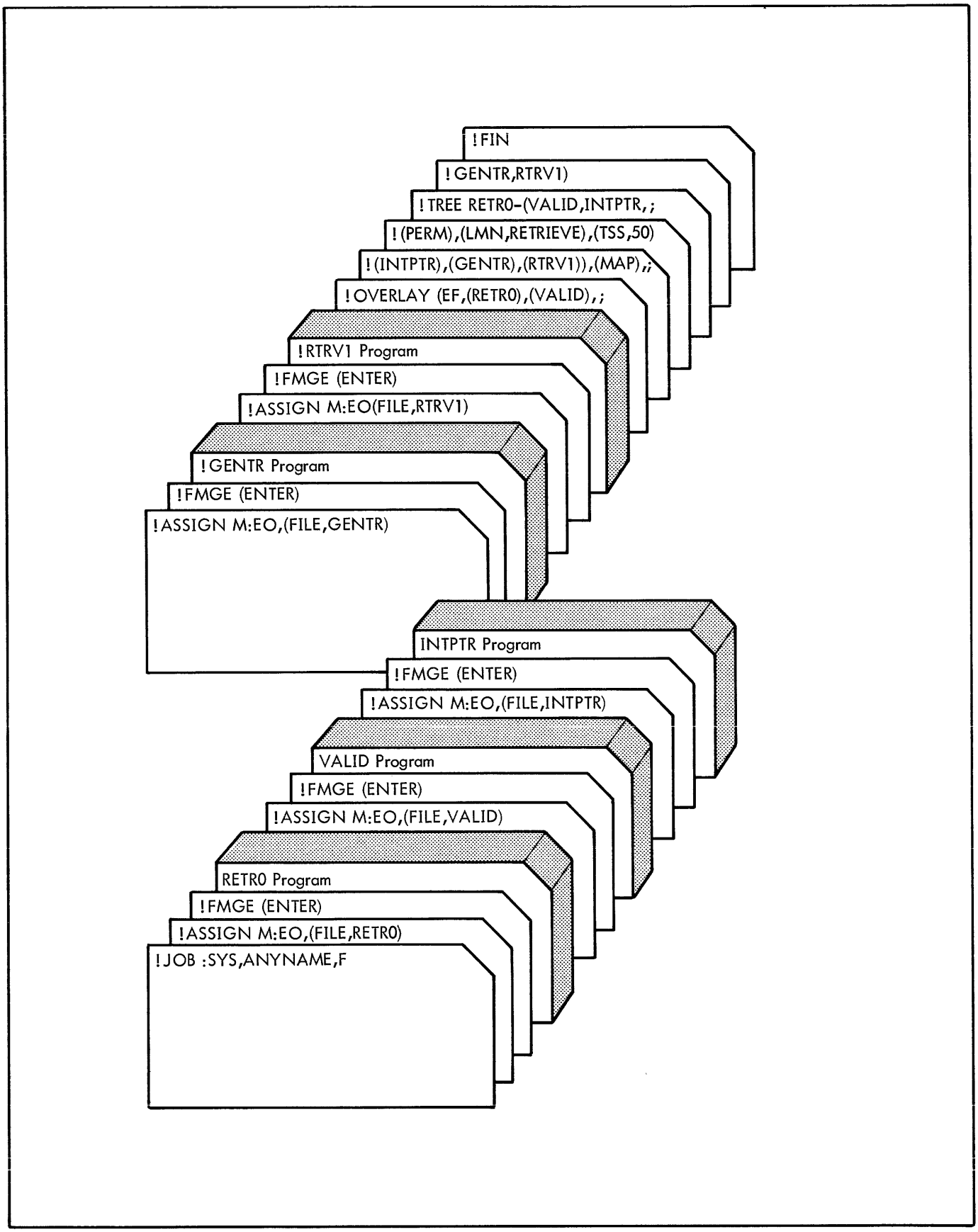
Example A-1. Forming the Dictnary Processor



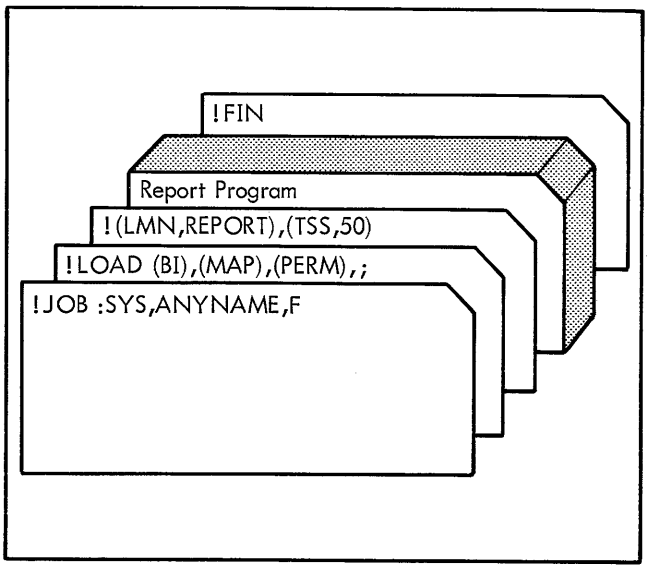
Example A-2. Forming the Fileup Processor



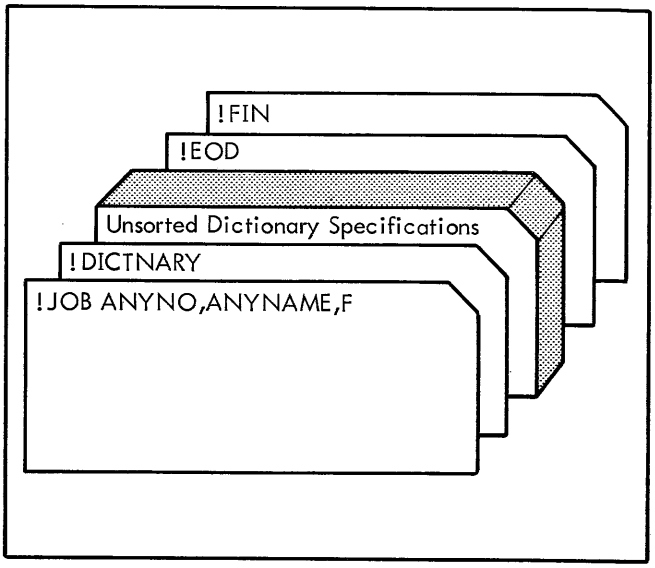
Example A-3. Forming the Retrieve Processor



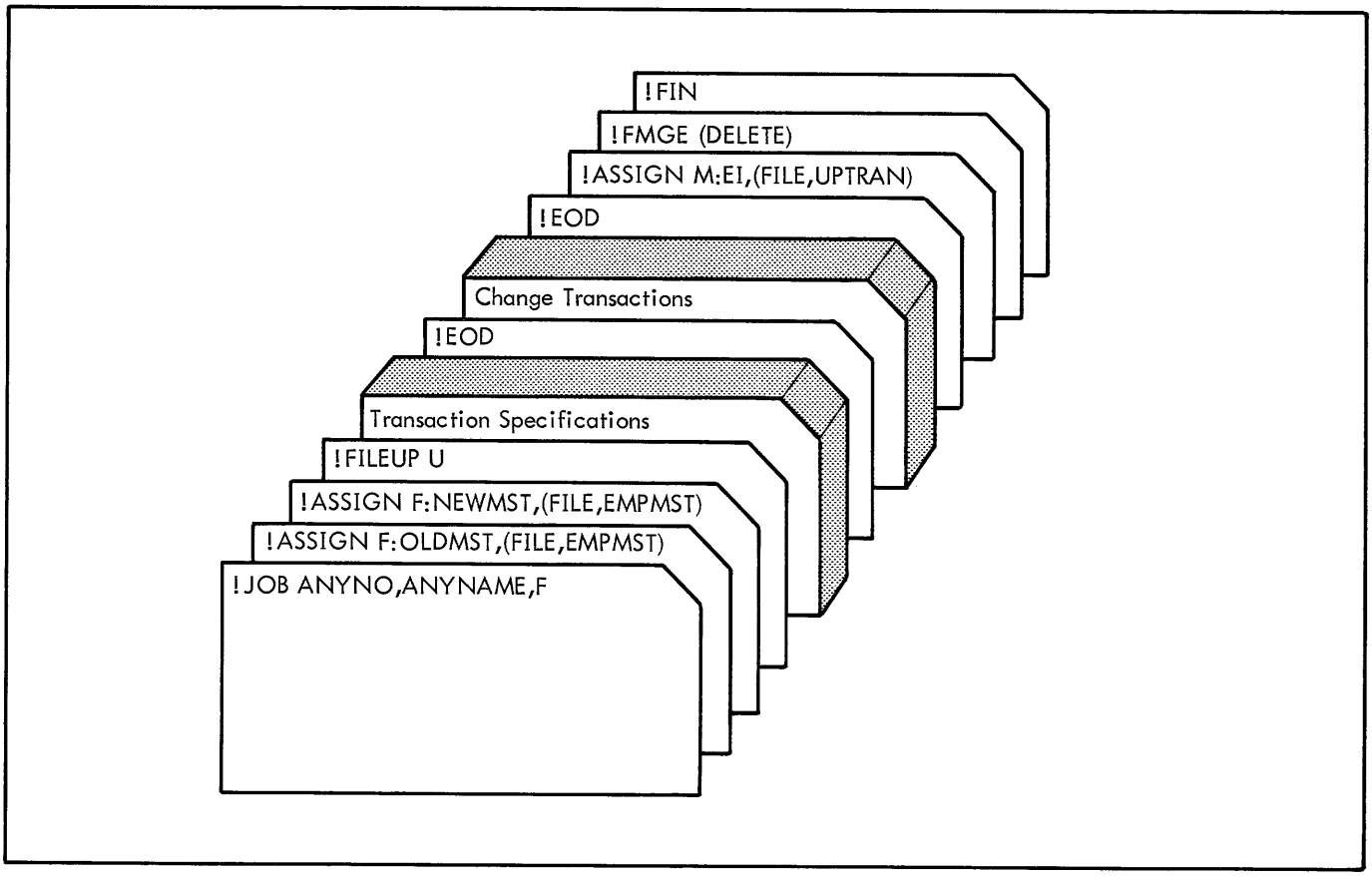
Example A-4. Forming the Report Processor



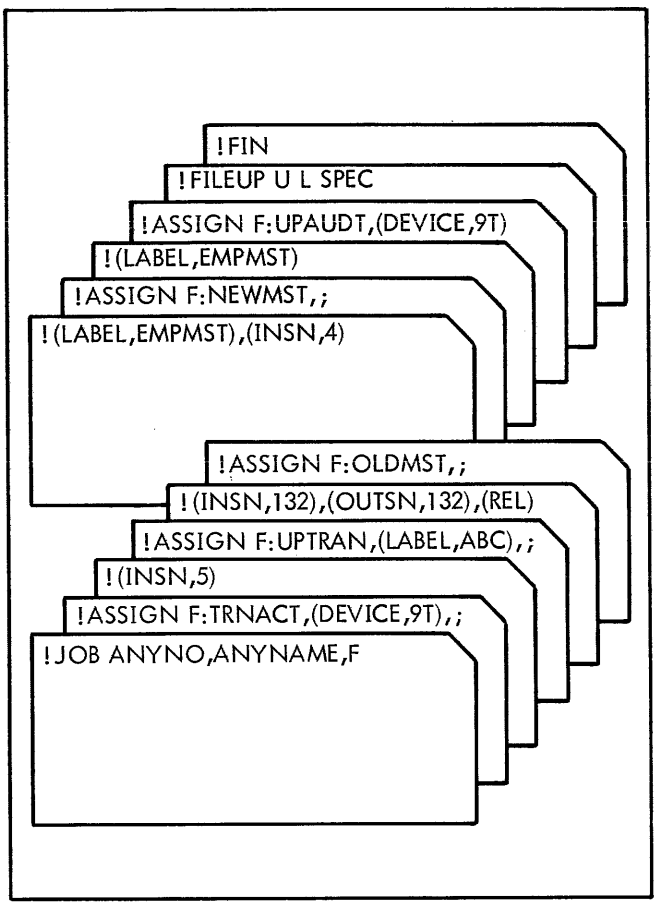
Example A-5. Dictnary Creation or Modification



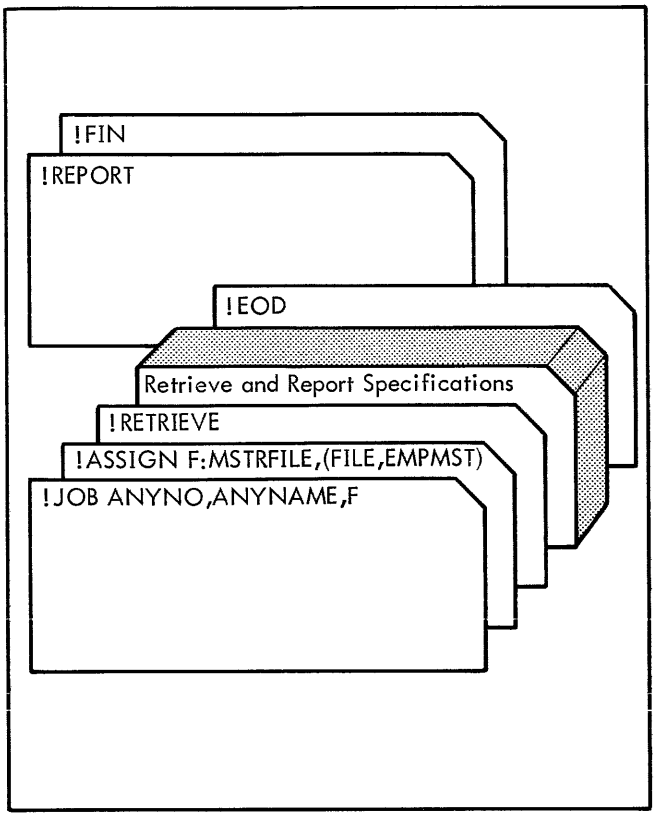
Example A-6. Updating a RAD File



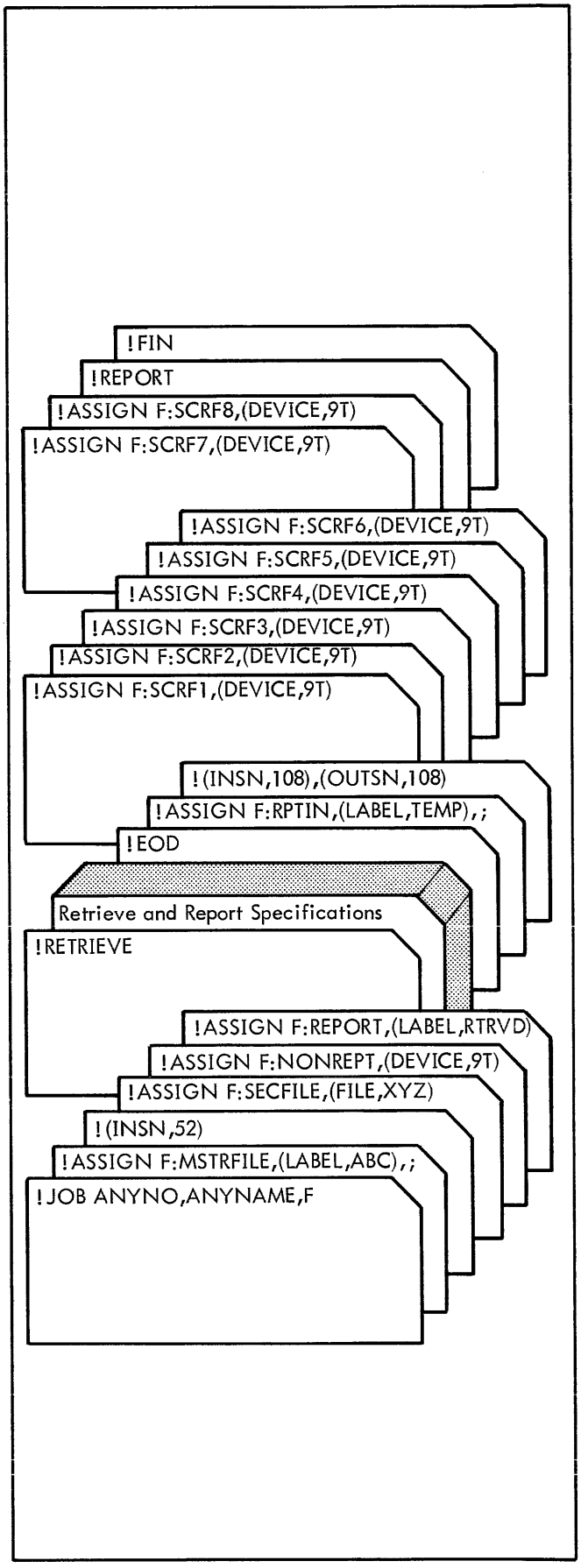
Example A-7. Updating a Tape File



Example A-8. Retrieval and Report Generation



Example A-9. Matched Retrieval and Report Generation



APPENDIX B. DCB NAMES FOR MANAGE PROCESSORS

Table 3 below gives the DCB names used by the Manage processors. Note that DCB defaults in Manage are arranged so that if all jobs are run under the same account number and if RAD space is available to handle the intermediate files, ASSIGNs are required for the following DCB names only.

DCB Name	Description
F:OLDMST	The input master file in Fileup
F:NEWMST	The output master file in Fileup
F:UPAUDT	The optional audit file in Fileup
F:MSTRFILE	The input data base file in Retrieve
F:SECFILE	The optional secondary input data base in Retrieve
F:NONREPT	The optional user output file in Retrieve

In such a situation, all parameter specifications and change transactions will be read through the SI device (usually cards).

In addition to the DCBs shown, all processors write error messages and diagnostics through the M:LO DCB. All DCBs accept account and password information via ASSIGN records.

Table 3. DCB Names for Manage Processors

DCB Name	Function	ASSIGN Required	Comments
		Dictnary	
M:SI	Input specification records.	No	Defaults to system SI device.
F:DCTSPC	Sorted input specification records.	No	Defaults to RAD. If assigned to tape, INSN and OUTSN are required.
F:DICT	Central dictionary input/ output.	No	To RAD only.
		Fileup	
M:SI	Input specification records.	No	Defaults to system SI device.
F:DICT	Central dictionary input.	No	To RAD only.
F:TRANS	Specification library input/ output.	No	To RAD only.
F:UPSPCS	Sorted input specification records.	No	Defaults to RAD. If assigned to tape, INSN and OUTSN are required.
F:TRNACT	Input change transactions.	No	Defaults to system SI device.
F:UPTRAN	Sorted input change transactions (optional).	No	Defaults to RAD. If assigned to tape, INSN and OUTSN are required. To release the sorted transaction file, use FMGE after a successful run.
F:OLDMST	Input master file.	Yes	May be to RAD, tape, or device.
F:NEWMST	Output master file.	Yes	May be to RAD, tape, or device.
F:UPAUDT	Output audit file (optional).	Yes	May be to RAD, tape, or device.
		Retrieve	
M:SI	Input specification records.	No	Defaults to system SI device.
F:DICT	Central dictionary input.	No	RAD only.
F:MSTRFILE	Input data base.	Yes	May be to RAD, tape, or device.

Table 3. DCB Names for Manage Processors (cont.)

DCB Name	Function	ASSIGN Required	Comments
		Retrieve (cont.)	
F:SECFILE	Input secondary data base (matching option).	Yes	May be to RAD, tape, or device.
F:NONREPT	Output user file (optional).	Yes	May be to RAD, tape, or device.
F:SI	Sorted input specification records.	No	Defaults to RAD. If assigned to tape, INSN and OUTSN are required.
F:REQLIB	Retrieve specification library input/output.	No	To RAD only.
F:RPTSPC	Output report specification file.	No	To RAD only.
F:REPORT	Retrieved data output file.	No	Defaults to RAD. May be assigned to tape.
		Report	
F:DICT	Central dictionary input.	No	To RAD only.
F:RPTSPC	Input report specification file.	No	To RAD only.
F:RTRVD	Retrieved data input file.	No	F:REPORT DCB from Retrieve is carried over automatically.
F:RPTIN	Sorted report data input file.	No	Defaults to RAD. If assigned to tape, INSN and OUTSN are required.

APPENDIX C. OWN-CODE LINKAGE FOR FILEUP

Registers 5 to 7 are used for communication between Fileup and the various user modules. In Table 4, below, these registers are described at the time of entry into the user's code and at the exit back to Fileup. Address values are at word resolution unless otherwise specified. An x indicates that the value in the designated retister is not significant. A method for generating a Fileup processor with user own-code is shown in Appendix A.

Table 4. Register Use for Own-Code Linkage

Entry Point		Register 5	Register 6	Register 7
FUPOP	Entry	x	Return address.	x
	Exit	x	x	x
FUPHD	Entry	0 means that the input master label has been read.†	Return address.	Byte address of input label buffer.††
	Exit	x	x	x
	Entry	1 means the output master label is ready to be written.†	Return address.	x
	Exit	x	x	Byte address of user's output label buffer.††
FUPIM	Entry	Return address to delete this input master record.	Return address to use this input master record.	Byte address of input master record.
	Exit	x	x	x
FUPMT	Entry	Byte address of matched change transaction record.	Return address.	Byte address of matched input master record.
	Exit	x	x	x
FUPOM	Entry	Return address to delete this output master record.	Return address to write this output master record.	Byte address of output master record.
	Exit	x	x	x
FUPCL	Entry	x	Return address.	x
	Exit	x	x	x

†The entry to FUPHD for output label processing is made before the entry for input label processing.

††Does not apply to master files with one or more header records ending with a file mark (F in user header label indicator of Dictnary file definition card). The user is responsible for reading and writing all such header records and file mark. Fileup will enter FUPHD after opening the input or output file. However, all required reading and writing of header records and file mark must be done in the user own-code. Fileup will resume processing under the assumption that the input or output file is positioned for reading or writing master data records.

INPUT AND OUTPUT LABEL FORMATS

The following conventions are observed in the processing of input and output label buffers:

1. The first byte of the buffer contains the length of the label which follows.
2. The second byte of the buffer contains the start of the label information.
3. Since the buffer is assumed to start on a word boundary, the label length byte occupies the first byte of a word.
4. Label length should not exceed either 255 bytes, or the length of a data block, else truncation will occur.

APPENDIX D. SAMPLE MANAGE RUNS

The sample dictionary printout shown below results from the file description given in Chapter 2.

```
MANAGE DATA FILE DICTIØNARY                                  TIME & DATE: 02:14 JUN 11,'69  PAGE:     1

****  FILE: MASTER    ****  LØGICAL RECØRD CHARACTER LENGTH: 1284,  BLØCKING FACTØR:   0
                                  MØNITØR FØRMATTED,  UNLABLED
                      CØMMENTS: AMALGAMATED MANUFACTURING CØRP.- MASTER FILE

LEFT  FIELD                            CØNTRØL   KEY CNTL FIELD  DECIMAL   ZERØ          PRINT   TRAIL  LEFT
CHAR   CHAR     FIELD      FIELD       CØUNTER   ØRDER    AUDIT   PØINT    SUPP  CØMMA   DØLLAR  MINUS  CHAR
PØSN  LENGTH     NAME      TYPE         FIELD         DIR FLAG   PØSITIØN  RESS  INSERT   SIGN   SIGN   PØSN
....   ...     ........    .....       ........    .   .  .....     .        .      .       .       .    ....
                                                             STANDARD REPØRT CØLUMN HEADINGS
                                                          ......................................

   3     1     DIVISIØN    BINRY                   1   A                     Z                             3
                                                          DIV

   7     2     EMPNØ       BINRY                   2   A                                                   7
                                                          EMPL> NØ

   9    21     NAME        ALPHA                                                                           9
                                                          EMPLØYEE NAME

  30    39     ADDRESS     TEXT                                                                           30

  30    20     L1          ALPHA                                                                          30
                                                          STREET

  50    19     L2          ALPHA                                                                          50
                                                          CITY-STATE

  69     3     ZIP         PACKD                                                                          69
                                                          ZIP>CØDE

  79     9     SØSECNØ     ALPHA                                                                          79
                                                           SØCIAL>SECURITY>NUMBER

  88     6     BIRTHDTE    ALPHA                                                                          88
                                                          DATE ØF> BIRTH >MØ/DY/YR

  94     1     SEX         ALPHA                                                                          94
                                                          S>E>X

  95     1     MARITAL     ALPHA                                                                          95
                                                          M>R>T>L

 101     1     CITIZEN     ALPHA                                                                         101
                                                          C>T>Z>N

 102     6     HIREDATE    ALPHA                                                                         102
                                                          DATE ØF>  HIRE  >MØ/DY/YR

 102     2     HIREMØ      ALPHA                                                                         102
 106     2     HIREYEAR    ALPHA                                                                         106
 109     3     HIRERATE    PACKD                                 3       Z              $                109
                                                          RATE> AT >HIRE

 132     1     DEPEND      BINRY                                         Z                               132
                                                          NØ>ØF>DEP

 137    10     PØSITIØN    ALPHA                                                                         137
                                                           JØB>TITLE

 151     3     PAYRATE     PACKD                                 3       Z      C       $       -        151
                                                          RATE> ØF > PAY

 154     6     PAYDATE     ALPHA                                                                         154
                                                            DATE >RATE SET>MØ/DY/YR
```

Figure D-1. Sample Dictionary

```
                              ••••••••••••••••••••
                              • SDS SIGMA MANAGE •
                              ••••••••••••••••••••

                              MANAGE REQUEST SDEM




                 REPORT TO STATE DEPARTMENT OF EMPLOYMENT SHOWING ALL NONCITIZEN
                 FEMALE EMPLOYEES.
                 REPORT IS SEQUENCED BY:
                  HIRE YEAR  (DESCENDING)
                  HIRE MONTH (DESCENDING)
                  NAME
                 THE REPORT SHOWS VARIOUS ITEMS OF DATA FOR EACH EMPLOYEE,
                 INCLUDING STARTING RATE OF PAY.
                 THE AVERAGE STARTING RATE OF PAY FOR EACH YEAR IS ALSO SHOWN.

JUN 11,'69                          NONCITIZEN FEMALE EMPLOYEES                                    1
••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••
DATE OF       EMPLOYEE NAME       EMPL    SOCIAL     M NO           JOB      RATE
  HIRE                             NO    SECURITY    R OF          TITLE      AT
MO/DY/YR                                  NUMBER     T DEP                   HIRE
                                                     L
••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••

10-12-69  HOLAHAN  MARY           015508 570-39-6608 M  2       WIRER       $ 1.800
ADDRESS   1479 BEVERLY DR        LONG BEACH    CAL.

08-10-69  DIEKMAN  EVONNE         026986 580-47-6832 M  8       ASSEMBLER   $ 1.850

08-10-69  DIXON  MARILYN          013974 560-22-3186 M  5       ASSEMBLER   $ 1.855
ADDRESS   4210 CART AVE          LYNWOOD       CAL.

08-10-69  DRISCOLL  SARAH         026262 554-62-5384 M  2       ASSEMBLER   $ 1.875
ADDRESS   1704 13TH ST           OCEAN PARK    CAL.
HIREYEAR AVERAGE

                                                                            $ 1.845

HIREYEAR ITEM COUNT 000004

10-12-68  HOLT  CAROL             027796 536-39-6167 M  2       WIRER       $ 1.755
ADDRESS   312 MORAGA AVE         LONG BEACH    CAL.

10-10-68  HORNEMAN  ROBERTA A     025339 570-39-1486 M  2       CLERK       $ 1.800
ADDRESS   4812 IMLAY AVE         VENICE        CAL.

10-10-68  HOWARD  MARTHA          027909 575-72-1489 M  4       CLERK       $ 1.825
ADDRESS   1514 VOIT ST           MARVISTA      CAL.

07-17-68  GALINDO  RUTH           012472 394-32-2812 M  1       SECRETARY   $ 2.750
ADDRESS   376 ALMA ST            HERMOSA BEACH CAL.

03-03-68  LESTER  SUSAN           017254 535-19-5447 W  1       WIRER       $ 2.100
ADDRESS   3122 ARIZONA ST        OCEAN PARK    CAL.

03-03-68  LOGAN  JOAN             029542 522-20-7169 W  1       WIRER       $ 2.110
ADDRESS   212 COLORADO AVE       VENICE        CAL.
HIREYEAR AVERAGE

                                                                            $ 2.057

HIREYEAR ITEM COUNT 000006

09-07-67  HUTH  HORTENSE H        028242 570-39-1142 D  2       ASSEMBLER   $ 2.000
ADDRESS   696 CLARK ST           SANTA MONICA  CAL.

07-17-67  GABLE  GRETA F          026062 525-62-7036 M  4       SECRETARY   $ 2.550
ADDRESS   3763 MOTOR AVE         SANTA MONICA  CAL.

07-17-67  GAGE  KAREN             018514 537-40-2956 M  3       SECRETARY   $ 2.550
ADDRESS   2205 MATRIX DR         SANTA MONICA  CAL.
```

Figure D-2. Sample Retrieved Data Report

```
JUN 11,'69                                NØNCITIZEN FEMALE EMPLØYEES                                          2
.........................................................................................................................
DATE ØF        EMPLØYEE NAME        EMPL     SØCIAL     M NØ         JØB      RATE
  HIRE                               NØ     SECURITY    R ØF        TITLE      AT
MØ/DY/YR                                    NUMBER      T DEP                 HIRE
                                                        L
.........................................................................................................................


07-17-67  GALLER  CARMEN          003391 532-48-7034 M  1       SECRETARY  $ 2.400
ADDRESS   428 VISTA AVE          SANTA MØNICA  CAL.

07-17-67  GALLØWAY  LØRI          027186 428-62-3078 M  5       SECRETARY  $ 2.375
ADDRESS   52286 MART LANE        VENICE         CAL.

07-17-67  STØUT  MICHELE          015954 548-62-7031 M  3       SECRETARY  $ 2.625
ADDRESS   422 KELTØN AVE         LØS ANGELES   CAL.
HIREYEAR AVERAGE

                                                                           $ 2.417

HIREYEAR ITEM CØUNT 000006


12-06-66  CRUSE  MAYBELLINE       026061 580-33-1414 M  3       ASSEMBLER  $ 2.000
ADDRESS   7177 SUNSET BLVD       N. HØLLYWØØD  CAL.

05-23-66  DEDRICK  SYLVIA R       013822 580-47-6683 M  2       EXPEDITER  $ 2.900
ADDRESS   3756 STEWART DR        SYLMAR         CAL.

05-23-66  DEMPSEY  ELLEN          030008 650-47-6688 M  2       EXPEDITER  $ 3.010
ADDRESS   1271 WALTER DR         LØS ANGELES   CAL.

05-23-66  EVANS  JEAN             011215 621-47-6681 M  2       EXPEDITER  $ 2.875
ADDRESS   556 16TH ST APT D      LØS ANGELES   CAL.
HIREYEAR AVERAGE

                                                                           $ 2.696

HIREYEAR ITEM CØUNT 000004

04-08-65  DEAN  CLAIR             013811 580-47-8827 N  2       ASSEMBLER  $ 2.100
ADDRESS   1548 HARVARD AVE       SANTA MØNICA  CAL.
HIREYEAR AVERAGE

                                                                           $ 2.100

HIREYEAR ITEM CØUNT 000001

-FINAL-  AVERAGE

                                                                           $ 2.243



JUN 11,'69                                NØNCITIZEN FEMALE EMPLØYEES
NUMBER ØF ITEMS RETRIEVED 000021
NUMBER ØF ITEMS READ IN   000021
```

Figure D-2. Sample Retrieved Data Report (cont.)

WAGE INCREASE PROBLEMS

The Amalgamated Corporation is engaged in wage negotiations. The management wishes to evaluate the annual wage costs of two alternative proposals covering assemblers, wirers, and group leaders. The alternatives are

1. A 6.75% rate increase for all three classifications.
2. A $.13 cents per hour increase for assemblers; a $.16 cents per hour increase for wirers; and a $.25 cents per hour increase for group leaders.

The Director of Industrial Relations has analyzed overtime patterns for these classifications and has arrived at the following formula:

1. Estimated annual gross wages for assemblers = (rate x 2080) + (rate x 1.5 x 102.5)
2. Estimated annual gross wages for wirers = (rate x 2080) + (rate x 1.5 x 156.0)
3. Estimated annual gross wages for group leaders = (rate x 2080) + (rate x 240) + $90.00

The Manage specifications shown in Figures D-3, D-4, and D-5 will retrieve the relevant employees and show the impact of the alternative wage increases on each department. Extensive use of arithmetic expressions is shown. In the retrieval specifications, these expressions calculate annual straight time and overtime earnings for each employee and the anticipated earnings if different increases were applied to each job type. The report specifications summarize the results of the expressions evaluated in Retrieve and also compute the effect of the 6.75% uniform increase. The resulting report is shown in Figure D-6.

SDS

SIGMA MANAGE GENERAL REQUEST SPECIFICATIONS

AUTHOR KOPITO

DATE JULY 1969

(CIRCLE DESIRED OPTIONS) Retrieval System

REQUEST NO.	REQUEST SPECIFICATION LIBRARY CONTROL			ENTER NAME OF:		SEARCH MODE		OUTPUT DISPOSITION		USER RETRIEVAL FORMAT	
WAGE	A USE MANAGE LIBRARY REQUEST SPECIFICATION	L ADD REQUEST SPECIFICATION TO MANAGE LIBRARY	A	PRIMARY FILE TO BE SEARCHED: MASTER	SECONDARY FILE TO BE MATCHED TO PRIMARY FILE	PERFORM SEARCH	S	REPORT GENERATOR	R	ENTIRE RECORD	E
						BY-PASS SEARCH	B	USER PROGRAM	U	SELECTED FIELDS: PACKED MULTIPLES	S
								VALIDATE SPECS. ONLY	V	PERMUTED MULTIPLES	P

(CIRCLE DESIRED OPTIONS) Reporting System

REPORT FORMAT		PRINT CONTROL		PAGE SIZE		TYPE OF PAPER			REPORT DATE	ENTER:	
DETAIL	D	SINGLE SPACE	1	SPECIAL	0	1-PART	1	4-PART	PRINT CURRENT DATE ON EACH REPORT PAGE: D	NUMBER OF PRIMARY FILE RECORDS SKIPPED	MAX. NO. OF RECORDS TO BE OUTPUT
GROUP DETAIL	G	DOUBLE SPACE	2	14" WIDE X 11"	1	2-PART	2	6-PART			
SUMMARY	S			8 1/2" WIDE X 11"	2	3-PART	3				
				11" WIDE X 8 1/2"	3						

REQUEST NO. Report Title - Text To Be Printed At Top Of Every Page

WAGE	B	01	CONFIDENTIAL- ALTERNATIVE WAGE ADJUSTMENTS -CONFIDENTIAL

Title Page - Text To Be Printed On Initial Page

02	THIS REPORT EVALUATES WAGE INCREASES PROPOSED FOR THE
03	AMALGAMATED MANUFACTURING CORPORATION.
04	
05	RETRIEVAL EXPRESSION COMPUTES ANNUAL STRAIGHT AND OVERTIME
06	EARNINGS FOR CURRENT RATES AND THEN FOR INCREASED RATES BY
07	JOB TITLE.
08	
09	REPORT SUMMARIZES BY TITLE WITHIN DIVISION.
10	EFFECT OF 6.75% INCREASE IS CALCULATED.
11	HORIZONTAL SUMMING IS EMPLOYED.
12	VERTICAL SUMMING, AND AVERAGE ANNUAL WAGES ARE DISPLAYED
13	ZERO FIELDS ARE CREATED TO PROVIDE A COLUMNAR EFFECT.
14	
15	

1733(3/69) SCIENTIFIC DATA SYSTEMS

Figure D-3. Wage Problem General Request Specifications

SDS

SIGMA MANAGE RETRIEVAL SPECIFICATIONS

REQUEST NO. WAGE C

AUTHOR KOPITO

DATE JULY 1969

LINE NO.	FIRST TERM: FIELD OR ARITHMETIC RESULT NAME	OPER: COMP OR ARITH OPER	OPERAND: FIELD OR ARITH. RESULT NAME/LITERAL	OPERAND: LITERAL CONSTANT VALUES	OPERAND: ARITHMETIC RESULT NAME	LOG. CON.: AND OR EXP END
0,1	PAYRATE	*	+2080.		CURANNUL	EXP
0,2	PAYRATE	*	+1.5		OTRATE	EXP
0,3	OTRATE	*	+102.5		OTGROSS	EXP
0,4	PAYRATE	+	+.13		NEWRATE	EXP
0,5	NEWRATE	*	+2080.		NEWANNUL	EXP
0,6	NEWRATE	*	+1.5		NEWOTRAT	EXP
0,7	NEWOTRAT	*	+102.5		NEWOTGRO	EXP
0,8	POSITION	EQ	=ASSEMBLER,			OR
0,9	OTRATE	*	+156.0		OTGROSS	EXP
1,0	PAYRATE	+	+.16		NEWRATE	EXP
1,1	NEWRATE	*	+2080.		NEWANNUL	EXP
1,2	NEWRATE	*	+1.5		NEWOTRAT	EXP
1,3	NEWOTRAT	*	+156.0			OR
1,4	POSITION	EQ	=WIRER,		TEMP	EXP
1,5	PAYRATE	*	+240.		OTGROSS	EXP
1,6	TEMP	+	+90.00		NEWRATE	EXP
1,7	PAYRATE	+	+.25		NEWANNUL	EXP
1,8	NEWRATE	*	+2080.		TEMP	EXP
1,9	NEWRATE	*	+240.		NEWOTGRO	EXP
2,0	TEMP	+	+90.00			END
2,1	POSITION	EQ	=GRP LEADER,			

Figure D-4. Wage Problem Retrieval Specification

SDS

SIGMA MANAGE RETRIEVED OUTPUT SPECIFICATIONS

REQUEST NO. WAGE D

AUTHOR KOPITO

DATE JULY 1969

LINE NO.	FIELD NAME, ARITH. RESULT NAME, NUMERIC LITERAL OR EXPR. LINE NO.	CONTROL BRK. PRINT ACTION: NON-PRINT (N)	SORT LEVEL (1-9)	DESCEND IND. (D)	BREAK CONTROL (1-9)	SUM LVL. (1-9 F)	ITEM CNT (1)	MIN (1-9 F)	MAX (1-9 F)	AVG. (1-9 F)	PGE EJECT (E)	RESET PAGE NO. (R)	NO. OF BLANKS TO RIGHT (01-99)	REPORT COLUMN HEADING (OVERRIDES DICTIONARY ENTRY)	PRINT FORMAT PICTURE (OVERRIDES DICTIONARY ENTRY)
0,1	DIVISION		1			1								>DIV	99-9
0,2	POSITION		2			2	I							> JOB>TITLE	
0,3	+0														
0,4	01*03													CUR	Z
0,5	CURANNUL						2							>STRAIGHT	ZZZ,ZZZ.99
0,6	OTGROSS						2							>O.T.	ZZ,ZZZ.99
0,7	05+06						2		2					>TOTAL	$ZZZ,ZZZ.99
0,8	01*03													+6.75%	Z
0,9	+1.0675														
1,0	05*09						2							>STRAIGHT	ZZZ,ZZZ.99
1,1	06*09						2							>O.T.	ZZ,ZZZ.99
1,2	10+11						2		2					>TOTAL	$ZZZ,ZZZ.99
1,3	01*03													ALT.2	Z
1,4	NEWANNUL						2							>STRAIGHT	ZZZ,ZZZ.99
1,5	NEWOTGRO						2							>O.T.	ZZ,ZZZ.99
1,6	14+15						2		2					>TOTAL	$ZZZ,ZZZ.99

Figure D-5. Wage Problem Report Specifications

```
                                   ••••••••••••••••••••
                                   • SDS SIGMA MANAGE •
                                   ••••••••••••••••••••

                                   MANAGE REQUEST WAGE


                        THIS REPORT EVALUATES WAGE INCREASES PROPOSED FOR THE
                           AMALGAMATED MANUFACTURING CORPORATION
                                   -
                        RETRIEVAL EXPRESSION COMPUTES ANNUAL STRAIGHT AND OVERTIME
                        EARNINGS FOR CURRENT RATES AND THEN FOR INCREASED RATES BY
                        JOB TITLE.
                                   -
                        REPORT SUMMARIZES BY TITLE WITHIN DIVISION.
                        EFFECT OF 6.75% INCREASE IS CALCULATED.
                        HORIZONTAL SUMMING IS EMPLOYED.
                        VERTICAL SUMMING, AND AVERAGE ANNUAL WAGES ARE DISPLAYED.
                        ZERO FIELDS ARE CREATED TO PROVIDE A COLUMNAR EFFECT.


JUN 14,'69                 CONFIDENTIAL- ALTERNATIVE WAGE ADJUSTMENTS -CONFIDENTIAL                    1
••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••
             CUR                                 +6.75%                                  ALT.2
 DIV    JOB        STRAIGHT    O.T.      TOTAL           STRAIGHT    O.T.      TOTAL           STRAIGHT    O.T.      TOTAL
      TITLE
••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••••

00-0 ASSEMBLER

POSITION TOTAL
                  258,377.60 19,098.83 $277,476.43     275,818.09 20,388.00 $296,206.09     273,520.00 20,218.15 $293,738.15
POSITION AVERAGE
                                       $  4,954.94                          $  5,289.39                          $  5,245.32
POSITION ITEM COUNT 000056

     GRP LEADER

POSITION TOTAL
                   41,392.00  5,496.00 $ 46,888.00      44,185.96  5,866.98 $ 50,052.94      45,552.00  5,976.00 $ 51,528.00
POSITION AVERAGE
                                       $  5,861.00                          $  6,256.62                          $  6,441.00
POSITION ITEM COUNT 000008

     WIRER

POSITION TOTAL
                  185,972.80 20,921.94 $206,894.74     198,525.96 22,334.17 $220,860.14     199,617.60 22,456.98 $222,074.58
POSITION AVERAGE
                                       $  5,046.21                          $  5,386.83                          $  5,416.45
POSITION ITEM COUNT 000041

DIVISION TOTAL
                  485,742.40 45,516.77 $531,259.17     518,530.01 48,589.15 $567,119.16     518,689.60 48,651.13 $567,340.73
DIVISION AVERAGE
                                       $  5,059.61                          $  5,401.13                          $  5,403.25
-FINAL-  TOTAL
                  485,742.40 45,516.77 $531,259.17     518,530.01 48,589.15 $567,119.16     518,689.60 48,651.13 $567,340.73


-FINAL-  AVERAGE
                                       $  5,059.61                          $  5,401.13                          $  5,403.25



JUN 14,'69                 CONFIDENTIAL- ALTERNATIVE WAGE ADJUSTMENTS -CONFIDENTIAL
NUMBER OF ITEMS RETRIEVED 000105
NUMBER OF ITEMS READ IN   000105
```

Figure D-6. Wage Problem Report

INDEX

Note: For each entry in this index, the most significant description appears first.

Note: For each entry in this index, the most significant description appears first.

Note: For each entry in this index, the most significant description appears first.

T

U

V

Z