

# IMAGE/1000 Data Base Management System

## Quick Reference Guide



HEWLETT  
PACKARD

---

HEWLETT-PACKARD COMPANY  
Data Systems Division  
11000 Wolfe Road  
Cupertino, California 95014

Library Index No.  
2DMGMT.345.92069-90005

MANUAL PART NO. 92069-90005  
Printed in U.S.A. March 1980

## PRINTING HISTORY

New editions are complete revisions of the manual. Update packages contain replacement pages or write-in instructions to be merged into the manual by the customer. Manuals will be reprinted as necessary to incorporate all prior updates. A reprinted manual is identical in content (but not in appearance) to the previous edition with all updates incorporated. No information is incorporated into a reprinting unless it appears as a prior update. The edition does not change.

First Edition . . . . . March 1980 (Software Revision Code 2001)

### NOTICE

The information contained in this document is subject to change without notice.

HEWLETT-PACKARD MAKES NO WARRANTY OF ANY KIND WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. Hewlett-Packard shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance or use of this material.

Hewlett-Packard assumes no responsibility for the use or reliability of its software on equipment that is not furnished by Hewlett-Packard.

This document contains priority information which is protected by copyright. All rights are reserved. No part of this document may be photocopied, reproduced or translated to another program language without the prior written consent of Hewlett-Packard Company.

## CONVENTIONS USED IN THIS MANUAL

Notation	Description
[ ]	An element inside brackets is optional.
{ }	When several elements are stacked within braces, the user must select one of these elements.  Example: $\left\{ \begin{array}{c} A \\ B \\ C \end{array} \right\}$ user must select A, B, or C.
<i>uppercase</i>	Uppercase letters mean that the information should be presented exactly as shown.
<i>lowercase</i>	Lowercase denotes a parameter which must be replaced by a user-supplied variable.  All item names, data set names and level code words consist of one to six ASCII characters restricted as follows: <ul style="list-style-type: none"><li>• only printable characters, ! through</li><li>• not allowed are: plus (+), minus (−), colon (:), comma (,), semicolon (;), left and right parentheses ( ( ) ), period (.), or equal sign (=).</li><li>• the first character must not be blank (space), or a number.</li><li>• embedded spaces are not allowed.</li></ul>
<i>file names</i>	All references to file names conform to RTE FMP conventions. Any file name referred to as an FMP namr consists of a file namr or a logical unit number. Any file name referred to as an FMP file namr consists only of a file namr; a logical unit number is not permitted. In an FMP file namr, negative lus are not permitted; only cartridge reference numbers can be used.

*log lu*

The log lu is LU 1 if the terminal being used is a Session terminal, the terminal lu if the terminal is a Multi-Terminal Monitor terminal and LU 1 if the terminal is neither a Session nor MTM terminal.

An underlined parameter means that you will be prompted for it if not passed in the original runstring.

Section I	Page
<b>DEFINITIONS AND CAPACITY INFORMATION</b>	
Section II	Page
<b>CREATING THE DATA BASE</b>	
Control Command .....	2-1
Data Base Privacy Levels .....	2-2
Item Definition .....	2-2
Set Definition .....	2-3
DBDS .....	2-4
DBBLD .....	2-5
Section III	Page
<b>QUERYING THE DATA BASE</b>	
Using QUERY .....	3-1
Terminating QUERY .....	3-2
Define Commands .....	3-2
Retrieving Data Base Entries .....	3-2
Modifying the Data Base .....	3-4
Generating Reports .....	3-5
Report Formatting .....	3-5
Report Editing .....	3-8
Storing and Maintaining Procedures .....	3-9
Utility Commands .....	3-10
Section IV	Page
<b>HOST LANGUAGE ACCESS—FORTRAN, ASSEMBLY, BASIC</b>	
Host Language Access .....	4-1
Parameter Definitions .....	4-1
IMAGE Subroutines .....	4-4
Section V	Page
<b>MAINTAINING THE DATA BASE</b>	
Utility Programs .....	5-1
Section VI	Page
<b>ERROR MESSAGES</b>	
IMAGE Numbered Error Messages .....	6-1
DBDS Error Messages .....	6-6
RECOV Error Messages .....	6-14
DBSPA Error Messages .....	6-14
QUERY Error Messages .....	6-16
IMAGE Error Codes and QUERY Messages .....	6-34



# DEFINITIONS AND CAPACITY INFORMATION

## Compound Data Item

Named group of identically defined, adjacent items within the same data entry.

Capacity: 255 elements, each of which may contain up to 255 bytes with the limitation that the data entry can only contain a total of 4096 bytes.

## Data Base

One or more data sets which have some logical relationship to one another.

Capacity: 50 data sets.

255 unique data item names (reusable in different data sets).

15 level words.

## Data Base Name

1 to 6 ASCII characters restricted as follows:

- only printable characters, ! through
- not allowed are: plus (+), minus (-), colon (:), comma (,), period (.), semicolon (;), left and right parentheses ( ( ) ), and equal sign (=)
- first character must not be blank (space) or a number
- embedded blanks not allowed

## Data Entry

See Entry

## Data Item

See Item

## Data Item type specifications for the item part of a data schema:

I1

Denotes an integer number in the range of -32768 to +32767.

R2

Denotes a real number in the range of  $\pm 1.70 \times 10^{38}$  to  $\pm 1.47 \times 10^{-39}$

Xinteger

Denotes an ASCII string of character length n, where n must be an even number not exceeding 255.

**Data Item Name**

See Item Name

**Data Set**

See Set

**Data Set Namr**

Same convention as Data Base Name.

**Edit Mask** (see QUERYING THE DATA BASE, REPORT EDITING, for format).

Capacity: Alphanumeric = 1 to 132 characters

Numeric = 1 to 20 characters

**Entry**

An ordered set of data items.

Capacity: 4096 bytes (media record plus data record).

127 unrepeated data item names.

**FIND Procedure**

Capacity: 50 logical relationships.

**FMP File Namr**

File namr. Must use cartridge reference number — negative plus are not permitted.

**FMP Namr**

File namr or logical unit number.

**Item**

Smallest accessible data element in a data base — may be expressed as numbers or ASCII character strings. Numeric values are integers (one 16-bit word long) or reals (two 16-bit words long).

**Item Name**

Same convention as Data Base Name.

**Level Word**

Code word corresponding to each data base privacy level. Convention is the same as Data Base Name.



**Path Count**

Integer from 0 to 16 indicating key data items and showing how many paths exist from the master data set to detail data sets through the key item link.

Capacity: 16 data paths from each master data set and to each detail data set.

**QUERY Command**

Capacity: 698 characters.

**Security Code**

Decimal integer 1 through 32767.

**Set**

Collection of data entries where each entry contains values for the same data items.

Capacity:  $2^{31} - 1$  data entries as long as the capacity of the disc cartridge is not exceeded.



# CREATING THE DATA BASE

## SCHEMA DEFINITIONS

**Control Command:** Optional. Specifies control options to be used by DBDS while processing the data base schema. Defaults are LIST, ROOT, SET, NOTABLE, and ERRORS=100.

Format: <b>\$CONTROL: [LIST,NOLIST,][ROOT,                   NOROOT,][SET,NOSET,][TABLE,                   NOTABLE,][ERRORS=nnn]                   [FIELD;]</b>
---

<b>LIST</b>	Lists each source record on the list device.
<b>NOLIST</b>	Suppresses the LIST option and, if an error occurs, prints the offending source record along with the error message.
<b>ROOT</b>	A root file is created if no errors are detected in the schema.
<b>NOROOT</b>	No root file will be created. The schema is merely checked for errors.
<b>SET</b>	Creates data sets if no errors are detected in the schema.
<b>NO SET</b>	No data sets will be created.
<b>TABLE</b>	A table of summary information is printed about the data sets after the schema has been scanned by DBDS (if no errors were detected).
<b>NOTABLE</b>	Suppresses the printing of the summary table.

**ERRORS=nnn** Sets the maximum number of errors allowed ( $0 < nnn < 999$ ). If this number is reached during processing, the schema processor prints an error message and terminates.

**FIELD** A table is printed for each data set describing the start and end of each data item if no errors are detected in the schema.

### Schema Structure

```
BEGIN DATA BASE: data base name;  
LEVELS: [level part]  
ITEMS: item part  
SETS: set part  
END.
```

### Data Base Privacy Levels:

Defined by a level word and level numbers (integers 1 to 15) in any order.

Format:	LEVELS:
	1 Level-word;
	. Level-word;
	. Level-word;
	15 Level-word;

### Item:

Defines the name, size, type, number of elements and privacy levels for each data item in a data base.

Format:	item name,[element count] type[(read level, write level)];
---------	--

**type** I1 = integer from -32768 to +32767.  
R2 = real from  $\pm 1.70 \times 10^{38}$  to  $\pm 1.47 \times 10^{-39}$ .  
Xinteger = ASCII string of character length n, where n must be an even number not exceeding 255.

**read level** integer from 1 to 15.

**write level** integer from 1 to 15; must be equal to or greater than the read level.

### Set:

Names the various data sets within the data base, indicates which data items belong to each set and links master data sets to detail data sets by defining key items. Items are stored in the order listed.

#### Formats:

##### Master Data Sets:

```
NAME: setname,type;
ENTRY:
    item name [(path count)],
    :
    item name [(path count)];
CAPACITY: capacity;
```

##### Detail Data Sets:

```
NAME: setname, D[ETAIL]
ENTRY: item name [(link)],
    :
    item name [(link)];
CAPACITY:capacity;
```

<b>type</b>	A[UTOMATIC] or M[ANUAL]
<b>path count</b>	integer in the range of 0 to 16, inclusive
<b>capacity</b>	integer 1 to $2^{31} - 1$
<b>link</b>	name of the master data set linked to the detail set item

## DBDS

Processes the data base schema.

<p>Run Format:        :RU,DBDS[,input,list,                           option]</p>
---

<b>input</b>	FMP file namr of file containing data base schema definition or the device lu where schema exists. Default is scheduling lu.
<b>list</b>	FMP namr where the listing is written. Default is LU 6.
<b>option</b>	Characters PU if existing data sets are to be purged before new ones are created.

---

*Note: Blanks may be inserted anywhere within the schema except within symbolic names, reserved words, or other syntactical entities.*

## DBBLD

Loads actual data item values into a data base structure and, optionally, confirms that they are presented in the proper format. Integer items occupy a six column field and real numbers a thirteen column field. Character type data item values appear in a field exactly as long as their defined length.

Run Format:       :RU,DBBLD[,input,list,  
                                  options]

- input**            FMP namr of data. Default is scheduling lu.
- list**             FMP namr where the list will be written.  
Default is LU 6.
- options:**        (entered in any order and separated by commas)
- ADD**             Input file error free entries are added to data base.
- ERRHLT**         Data processing stops when an error is detected.
- NOLIST**         Input file data is not listed to output device.
- n**                Integer from 1 to 512 which specifies line length. Default is 72.

### Input File Format:

```
data base namr,level word;  
$SET:set name 1  
data for set name 1  
  :  
  :  
$SET:set name n  
data for set name n  
$END
```





# QUERYING THE DATA BASE

## Using QUERY

Run	:RU,QUERY[,input,list,logn
Format	ECHO, node]

- input** The FMP namr from which the input will come. Default is LU 1 or user's terminal in an MTM environment.
- list** FMP namr where the list will be written. Default list is interactive input device or LU 6 if the input is a file or not an interactive device. If the list file does not already exist, QUERY will create it.
- log** FMP namr which specifies the logging device or file. ECHO lists each command to the log device or file before it is executed. Default log unit is the interactive input device or LU 1 when the input is a file or not an interactive device. QUERY creates the log file if it does not exist.
- node** User's node number. Allows the user to run QUERY remotely using the DS/1000 REMAT program. Input device must be either interactive or an FMP file namr local to the data base. All LU's specified must be system rather than session. Any files and data bases referenced by QUERY running remotely must exist on the node local to the QUERY program. Before requesting a remote copy of QUERY, the REMAT switch (SW) command must be executed with the source node as the node at which QUERY is to be run and the destination node as the user's node. The remote copy of QUERY can be invoked using either the RU or RW REMAT command.

Interactive input      **QUERY READY**  
response:              **NEXT?**

**Batch mode** Prompt is suppressed. Input is expected  
**response:** in the order it would be entered  
interactively.

## Terminating QUERY

### EXIT;

The EXIT; command, typed in response to NEXT?, terminates QUERY.

## Define Commands

Data base

definition: **DATA-BASE=data base namr;**

Select file

definition: **SELECT-FILE=file namr;**

When input is interactive, QUERY will prompt for level code word (**LEVEL=?**) and open mode (**OPEN MODE = ?**). Responses should be the appropriate level word (defined by **LEVELS**: when the data base was created) and open mode (1 = read/write, shared access, 3 = read/write, exclusive access, 8 = read only, shared access).

## Retrieving Data Base Entries

Find	<b>FIND retrieve procedure END;</b>
Command	or
Format:	<b>FIND NAME=procedure name;</b>

**procedure name** FMP file namr of a file stored on disc containing a FIND command which uses a retrieve procedure.

**retrieve procedure** Specifies a relation between a data item and data item value. QUERY will find the values which fit the stated relation and store the record addresses of those data entries.

Retrieve Procedure Format:

[set name.] item name operator  
"value"

- set name** Name of data set containing the item. Optional when the item is defined in only one set, or when the data set is specified earlier in the FIND command.
- item name** Name of the item. For a compound item, only the first element is used.
- operator** Relational operator indicating the type of comparison to be made.

Relational Operator	Meaning
IS	equals
IE	
ISNOT	is not equal to
INE	
ILT	is less than
INLT	is not less than
IGT	is greater than
INGT	is not greater than

- value** Value to be compared with each of the values of the data items named in the relationship. Value must be enclosed in quotes and appropriate to the data item type.

## Modifying the Data Base (Formats below)

### UPDATE A[DD]:

Adds a data entry to a detail or manual master data set. The user will be prompted with the name of each item in the set. Item values to be added should be enclosed in quotes and entered after each item name.

### UPDATE D[ELETE]:

Deletes all records selected by the FIND command.

### UPDATE R[EPLACE]:

Replaces the value of a data item with another value in all data entries in the current select file. The user will be prompted with

**ITEM?**

which must be followed by an item name, equal sign, and item value enclosed in quotes — item name="item value"

To Add, Delete, or Replace Data:

**UPDATE NAME=procedure name;**

(if procedure file is stored on disc)

or

**Add Data            UPDATE A[DD],data set name;**

**Format:                (to add data)**

or

**Delete Data         UPDATE D[ELETE];**

**Format:                (to delete data)**

or

**Replace Data        UPDATE R[EPLACE];**

**Format:                (to replace data)**

**data set name**        Name of data set to which entry will be added.

**procedure name**      Name of the UPDATE A,D, or R command stored as a procedure on the disc.

---

*Note: Path items cannot be modified.*

## Generating Reports

```
Report      REPORT ALL [,character];  
Command     or  
Format:    REPORT NAME=procedure name;  
           or  
           REPORT [,character];  
           body  
           END;
```

**character** Any ASCII printing character — determines the printing of certain optional information.

**procedure name** FMP file name of a file containing REPORT commands stored as a procedure in a procedure file.

**body** Header, detail, edit, sort, group and total statements.

## Report Formatting

Format Aid Parameters:

**SPACE space-control** Skips lines on report page before or after printing any line. A[number] causes that number of lines to be spaced after the line, while B[number] spaces that number of lines before a line. The number parameter is an integer from 1 to 5.

**SKIP skip-control** Skips to a new page before or after printing a detail, group or total line. Parameter A will page after a line while B will cause the page before a line.

**Print position** Integer from 1 to 132 defining the ending column position of a character string or data item value in a report.

## Edit

EZ specified will suppress any leading zeroes in numeric data item values.

Enumber (where number is an integer from 0 to 9) corresponds to the number of an edit statement that is defined after the statement containing the edit parameter.

### Example:

To page before and space after the header line "PAGE", ending in column 35:

```
H1,``PAGE``,35,SKIP B,SPACE A2
```

### Report Statements:

#### Header Statement Format:

```
Hnumber,data type,print  
position[,SPACE space-control];
```

**number** Integer from 1 to 5 specifying on which header line (out of five possible lines) the information is to appear. Header information in a header statement labeled H1 appears in the first line, H2 appears in the second line, etc.

**data type** Either an ASCII character string bracketed by quotes or the word PAGENO.

#### Detail Statement Format:

```
D[n],data type,print  
position[,SPACE space-  
control][,SKIP skip-control]  
[,edit];
```

**n** Integer from 1 to 9.

**data type** ASCII character string bracketed by quotes or the name of a data item.

Edit Statement Format:  
`Enumber,"edit mask";`

**number** Integer from 0 to 9 identifying the edit statement. No two edit masks may be identified by the same integer.

**edit mask** 1 to 20 numeric edit characters or 1 to 132 alphanumeric edit characters, bracketed by quotes. See REPORT EDITING for complete edit mask information.

Sort Statement Format:  
`S[level],data item name;`

**level** Integer from 1 to 5.

**data item name** Name of data item contained in the data entries selected by the FIND command.

Group Statement Format:  
`Glevel,data type,print  
position[,SPACE space-control]  
[,SKIP skip-control][,edit];`

**level** Integer from 1 to 5, corresponding to the level appearing in a sort statement.

**data type** Either an ASCII character string bracketed by quotes, or the name of a data item.

Total Statement Format:  
`Tlevel,data type,print  
position[,SPACE space-  
control][,SKIP skip-  
control][,edit] { ,AVERAGE }  
                                  { ,COUNT } ;  
                                  { ,ADD }`

<b>level</b>	Integer from 1 to 5 corresponding to the level appearing in a sort statement or the letter F.
<b>data type</b>	Either an ASCII character string bracketed by quotes or the name of a data item.
<b>ADD</b>	Instructs QUERY to add the values specified in data type. Accumulators are set up for each control level and, if the user specifies F as the level, for the final level. Capacity is 10 digits and only reals and integers may be added.
<b>AVERAGE</b>	Instructs QUERY to average the values of the data item specified in data type. Only reals and integers may be averaged.
<b>COUNT</b>	Instructs QUERY to print the number of data item values printed after the last and before the current control break.

## Report Editing

### Alphanumeric Editing

<i>Example:</i>	<b>Data Item</b>	<b>Edit Mask</b>	<b>Printed Result</b>
	<b>Value</b>		
	<b>ABCD</b>	<b>"X-X-X-X"</b>	<b>A-B-C-D</b>

Edit mask consists of X's as place holders with ASCII characters inserted. QUERY reads the mask from right to left, so fewer X's in the mask than characters in the data cause the leftmost data item characters to remain unprinted. More X's in the mask than characters in the data cause asterisks to be printed in the extra columns.



## Numeric Editing:

Edit mask consists of place holders (9,Z,\*, \$), sign characters (CR, -), and inserted ASCII characters. Each place holder is replaced by a decimal digit from the data item value in the corresponding position of the output field.

### *Explanation of place holders:*

- 9** Place holder. Does the same thing as an X in an Alphanumeric Edit.
- Z** Zero suppression place holder. If a zero appears to the left of the leftmost significant digit, QUERY inserts a blank in that position and all zeroes to its left are also replaced by blanks.
- \*** Asterisk place holder. Leading zeroes are replaced with asterisks.
- \$** Dollar sign place holder. The first zero to the left of the leftmost significant digit is replaced with a dollar sign. All zeroes to its left are replaced with blanks.
- CR** Sign character. Always appears in the two rightmost positions of the edit mask. If the data item value is positive, QUERY prints two blank characters in place of the CR.  
  
Sign character like CR. If the data item value is negative, QUERY prints the minus sign (-) in the rightmost position of the output field. If positive, QUERY prints a blank in place of the minus sign.

## Storing and Maintaining Procedures

Create a procedure:	<code>CREATE NAME=procedure namr;</code>
Display a procedure:	<code>DISPLAY NAME=procedure namr;</code>
Delete a procedure:	<code>DESTRDY NAME=procedure namr;</code>
Execute a program	(can be used to modify a procedure — default is EDITR): <code>EXECUTE[=program name];</code>

---

*Note: Real numbers cannot be edited.*

## Utility Commands

XEQ Command Format: **XEQ=file name;**

**file name** FMP file name of an ASCII file containing commands and parameters.

FORM Command Format: **FORM;**

HELP Command Format:

**HELP[,command][ ,FUNCTION]  
[ ,SYNTAX][ ,OPERANDS];**

**command** Either: **CREATE, DATA-BASE DESTROY, DISPLAY, EXECUTE, EXIT, FIND, FORM, HELP, LIST, REPORT, SELECT-FILE, UPDATE, or XEQ.**

LIST Command Format: **LIST=list device;**

**list device** FMP name where the list is to be written.

# HOST LANGUAGE ACCESS—FORTRAN, ASSEMBLY, BASIC

## Host Language Access

Calling an IMAGE Subroutine:

```
FORTRAN CALL name (parameter,  
                  parameter,...,parameter)
```

```
Assembly JSB name  
          DEF  *(n+1)  
          DEF  parm1  
          DEF  parm2  
          .  
          .  
          .  
          DEF  parmn
```

```
BASIC lineno CALL name(parameter,  
                       parameter,...,parameter)
```

## Parameter Definitions:

### IBASE

*(FORTRAN and Assembly Language)*

Integer array containing the data base FMP file namr. The first word is either a DS/1000 node number (if accessing a data base at a remote site), or two ASCII blanks. Words two through the end of the array contain at least the data base and the security code, separated by a colon and terminated by a blank or semi-colon.

```
Example: DIMENSION IBASE(7)  
          DATA IBASE /101,2HQA,2HDB,  
                    2H:1,2H00,2H:4,2H3;/
```

*(BASIC)*

One-dimensional string array containing two ASCII blanks followed by the data base namr. The first two bytes cannot contain a DS/1000 node number, but must be blank.

## ILEVL

Gives the user's level code word.

*(FORTRAN and Assembly Language)*

Integer array of three words, blank-filled if necessary.

*(BASIC)*

One-dimensional string array.

## ID

Supplies a data set name for data base manipulation calls. For DBINF calls, ID may also be a data item name.

*(FORTRAN and Assembly Language)*

Integer array of three words, blank-filled if necessary. May also contain an integer which is a data set or data item number.

*(BASIC)*

One-dimensional string array.

## IMODE

Passes information to the subroutine via an integer variable. It may be an access mode parameter, an information series indicator or a read option. For calls which do not have multiple modes, IMODE must contain the value 1.

## ISTAT

Returns status information to the user. If the operation requested was unsuccessful, a non-zero code is returned in the first word or element of the array. If the operation is successful, the first word or element of the ISTAT array is set to zero. Depending on the operation, additional information may be passed via the remainder of the array.

*(FORTRAN and Assembly Language)*

Ten words long.

*(BASIC)*

Variable length, ranging from one to six elements.

## IBUF

*(FORTRAN and Assembly Language only)*

Buffer used to pass information to and from the user, the size being dependent on the information to be passed.

## VALUE-LIST

*(Basic only)*

List of one or more variables separated by commas and corresponding in type and number to the data items given in NAME-LIST. The variables in VALUE-LIST are the values either retrieved from or added to the data-base as values for those items named in the NAME-LIST.

```
Format:  DIM I$[12]
          DATA "ASSY,#,FAIL#"
          READ I$
          .
          .
          .
          CALL DBGET(Q$,S$,M7,S[1],A$,
                   I$,Y$,F1)
```

## ITEM

Contains the name of the data item on which a search is to be made.

*(FORTRAN and Assembly Language)*

Integer array of at least three words or, alternately, an integer variable containing a data item number.

*(BASIC)*

One-dimensional string array.

## IARG

Real or integer variable, integer array, one-dimensional string array containing a value given to a key item, or a doubleword integer variable containing a relative record number.

## LIST

*(FORTRAN and Assembly Language only)*

Integer array containing a list of data item names or numbers. A list of data item names is a set of data item names separated by commas and terminated by a blank or semi-colon. A list of data item numbers is an integer N followed by a set of N data item numbers, where N is from 1 to the number of items in the data set being accessed.

```
Format:  DIMENSION LIST(9)
         DATA LIST/2HAS,2HSY,2H#,,
         2HPA,2HRT,2H#,,2HEM,
         2HP#,2H; /
```

## NAME-LIST

*(BASIC only)*

Used in retrieval, add, and update operations to specify the data items that will be affected. NAME-LIST is a one-dimensional string array containing a list of one or more data item names, separated by commas and terminated by a blank or semi-colon. No more than ten names may appear in a name-list, except in a DBGET operation which can contain only nine.

```
Format:  DIM G$[41]
         DATA "ASSY#,DATE,PART#,
         FAILCD,REFDES,PNTS,
         EMP#;"READ G$
```

## IMAGE Subroutines

### DBCLS

```
Format:  DBCLS(ibase,id,imode,istat)
```

Terminates access to a data base or closes a data set.

**imode values:** 1 — close the data base  
2 — close a data set

### DBDEL

```
Format:  DBDEL(ibase,id,imode,istat)
```

Deletes the current entry from a data set. The data base must be open in mode 1 or 3 and the level of access granted at open time must allow deleting from the data set. If open in mode 1, the data base must be locked. Imode must contain the value 1.

## DBFND

Format: **DBFND(ibase,id,imode,istat,  
item,iarg)**

Locates the master set entry that matches the specified key item value and sets up pointers to the detail data set chain in preparation for chained access to the data entries which are members of the chain. Imode must contain the value 1.

## DBGET

*(FORTRAN and Assembly Language)*

Format: **DBGET(ibase,id,imode,istat,  
list,ibuf,iarg)**

*(BASIC)*

Format: **DBGET(ibase,id,imode,istat,  
iarg,name-list,value-list)**

Provides seven different methods for reading the data items of a specified entry.

**imode values:** 1 — reread current record  
2 — forward serial read  
3 — backward serial read  
4 — directed read by record number  
5 — forward chain read  
6 — backward chain read  
7 — directed read by key item value

## DBINF

Format: **DBINF(ibase,id,imode,  
istat,ibuf)**

Returns information about the data base being accessed provided the user's access level is high enough. If it is not, data items, data sets, and data paths are considered nonexistent by IMAGE.

**imode values:** 101 — determine access to an item  
102 — obtain information on an item  
103 — get list of items in the data base  
104 — get list of items in a data set  
201 — determine access to a set

- 202 — obtain information on a set
- 203 — get list of sets in the data base
- 204 — get list of sets containing an item
- 301 — get list of paths for a data set
- 302 — determine master data set's search item
- 401 — obtain current chain pointers
- 402 — restore saved chain pointers

## DBLCK

Format: **DBLCK(ibase, id, imode, istat)**

Provides temporary exclusive control of a data base by locking the root file. A redundant call while in open mode 1 or a call in open mode 3 is ignored.

**imode values:** 1 — lock with wait  
2 — lock, no wait

## DBOPN

Format: **DBOPN(ibase, ilevl, imode, istat)**

Initiates access to the data base and establishes the access mode and level for all subsequent data base interaction.

**imode values:** 1 — shared read/write with lock access  
3 — exclusive read/write access  
8 — shared read-only access

## DBPUT

Format: **DBPUT(ibase, id, imode, istat, { list } , { ibuf } , { name-list } , { value-list } )**

Adds new entries to a manual master or detail data set. The data base must be open in mode 1 or 3 and the user's access level must allow write access. If opened in mode 1, the data base must be locked before DBPUT is called. Imode must contain the value 1.



## DBUNL

Format: **DBUNL(ibase,id,imode,istat)**

Relinquishes the temporary exclusive control a data base acquired by a previous call to DBLCK. DBUNL unlocks the root file and disables the calling program's data base write access until another DBLCK call is made. Redundant calls and calls not made in open mode 1 are ignored. Imode must contain the value 1.

## DBUPD

Format: **DBUPD(ibase,id,imode,istat,**

**{ list } , { ibuf }**  
**{ name-list } , { value-list }**

Modifies values of data items in the entry residing at the current record address of a specified data set when the data base is open in mode 1 or 3. If open in mode 1, the data base must be locked. Key items cannot be modified. Imode must contain the value 1.



# MAINTAINING THE DATA BASE

## Utility Programs

### DBL0D

Loads data base values from a file created by DBULD into a data base.

Run	:RU,DBL0D[,console][,storage]
Format:	[,root][,level][,abort]

- console** Console lu. Default is scheduling lu
- storage** FMP namr where the data has been stored. Default is lu 8.
- root** FMP namr of root file. No default — this is an interactive prompt.
- level** Highest level word defined for the data base. No default — this is an interactive prompt.
- abort** AB to abort if duplicate file name is encountered. CO to overwrite existing files. No default — this is an interactive prompt.

DBLOD interaction:

:RU,DBLOD,15,8,QADB:100:43,SECUR,CO

or

:RU,DBLOD,15,8

ROOT FILE NAMR ?QADB:100:43

HIGHEST LEVEL CODE WORD ?SECUR

OVERWRITE EXISTING FILES (YES OR NO)?YES

END of REEL (reel #) (DBLOD reaches the end  
of a magnetic tape reel)

Mount TAPE (data base name)

(reel #)

ON LOGICAL UNIT DEVICE

(lu #)

READY (YES OR NO)?

loading process  
not yet  
complete

END OF FILE (storage file  
name)

NEXT STORAGE FILE (AB TO  
ABORT)?

DBLOD has  
reached the  
end of the  
disc file

DATA BASE LOAD COMPLETED

load operation  
is complete

## DBULD

Stores data base information to magnetic tape or disc file.

Run	:RU,DBULD[,console][storage]
Format:	[,root][,level][,abort]

- console** Console lu. Default is scheduling lu.
- storage** FMP namr where the data has been stored. Default is lu 8.
- root** FMP namr of root file. No default — this is an interactive prompt.
- level** Highest level word defined for the data base. No default — this is an interactive prompt.
- abort** AB to abort if end of storage device is reached, CO to continue under this condition.

DBULD interaction:

```
:RU,DBULD,15,8,QADB:100:43,SECUR,AB
```

or

```
:RU,DBULD,15,8  
ROOT FILE NAMR ?QADB:100:43  
HIGHEST LEVEL WORD ?SECUR  
ABORT AT END OF STORAGE DEVICE  
(YES OR NO)? YES
```

SAVE TAPE ON LOGICAL DEVICE

(lu#)

AS (data base name)

(reel number)

DBULD reaches  
the end of a reel

MOUNT NEXT REEL ON LOGICAL  
DEVICE

(tape drive lu number)

READY (YES OR NO)?

Printed by DBULD  
when new tape  
reel should be  
mounted

DATA BASE UNLOAD COMPLETE

Completion of  
dump message

## DBSTR

Dumps a root file and associated detail, automatic and manual master data sets (including pointers) to a magnetic tape or disc file.

<b>Run</b>	<b>:RU,DBSTR[,console][,storage]</b>
<b>Format:</b>	<b>[,root][,level][,abort]</b>

**console** Console lu. Default is scheduling lu.

**storage** FMP namr where the data has been stored. Default is lu 8.

**root** FMP namr of root file. No default — this is an interactive prompt.

**level** Highest level word defined for the data base.

**abort** AB to abort if end of storage device is reached, CO to continue under this condition.

## DBRST

Loads a data base and associated root file from backup files created by DBSTR.

<b>Run</b>	<b>:RU,DBRST[,console][,storage]</b>
<b>Format:</b>	<b>[,root][,level][,abort]</b>

**console** Console lu. Default is scheduling lu.

**storage** FMP namr where the data has been stored. Default is lu 8

**root** FMP namr of root file. No default — this is an interactive prompt.

**level** Highest level word defined for the data base. No default — this is an interactive prompt.

**abort** AB to abort if duplicate file name is encountered, CO to overwrite existing files. No default — this is an interactive prompt.

## DBSPA

Indicates number of records used, number of free records, and the capacity of all data sets in a data base.

<b>Run</b>	<b>:RU,DBSPA[,console][,list]</b>
<b>Format:</b>	<b>[,root][,level]</b>

- console** Console lu. Default is the scheduling lu.
- list** lu of the list device. Default is the console lu.
- root** FMP file namr of the root file. No default — this is an interactive prompt.
- level** User's level word. No default — this is an interactive prompt.

DBSPA interaction:

```
:RU,DBSPA,15,6,QADB:100:43,SECUR
```

or

```
:RU,DBSPA,15,6
```

```
ROOT FILE NAMR ?QADB:100:43
```

```
LEVEL CODE WORD ?SECURE
```

DBSPA response to data:

Lists all data sets, capacity, records used, free records, and any discrepancies.

## RECOV

Displays the number of programs that have the data base open in any access mode and allows the user to close any data base inadvertently left open.

<b>Run</b>	<b>:RU,RECOV[,console]</b>
<b>Format:</b>	<b>[,list][,node]</b>

**console** Console lu. Default is the scheduling lu.

**list** List lu. Default is the console lu.

**node** Operator's node number in DS network. Default is the local node.

### Recov responses:

**DONE** (successful simulated closure)

Printout of coordinating table

**DATA BASE** <fnms,r6<sup>1</sup> RELEASED (user count is reduced to zero)

**CLOSURE UNSUCCESSFUL** (unsuccessful simulated closure)



## ERROR MESSAGES

### IMAGE NUMBERED ERROR MESSAGES:

- 100 Illegal data set reference
- 101 Illegal item reference
- 102 Data item specified is not a key item
- 103 Illegal data base parameter, or data base not open to user
- 104 The mode of operation for the data base (specified in the DBOPN call) is insufficient for the type of access requested in a subroutine call.
- 105 Detail data set is full; no empty record space exists
- 106 Master data set is full; no empty record space exists
- 107 A master data set does not contain an entry for a key item value
- 108 A request has been directed to an automatic master data set
- 110 A key data item value already exists in a manual master data set
- 111 Path has not been initialized for a chain read, or record number in directed read is illegal
- 112 User has attempted to change the value of a key item
- 113 User has attempted to delete a manual master data set entry that still has links to non-empty detail data set chains
- 114 Record accessed is empty
- 115 The mode specified by the user is invalid
- 116 File specified by the IBASE parameter is not a root file
- 117 The security code given does not match the security code for the data base being accessed
- 118 Data set is not write enabled for the user's access level
- 119 The root file for the data base being accessed does not exist
- 120 The data set being accessed is not a detail data set
- 121 The detail data set being accessed is not linked to any master data sets
- 122 A chain read cannot be performed
- 123 Data set being accessed is not a master data set
- 124 Illegal DBINF request

## ERROR MESSAGES (Continued)

### IMAGE NUMBERED ERROR MESSAGES:

- 125 Invalid data set or item reference used in DBINF request
- 128 Not enough space for IMAGE data buffers (solution: make program smaller, segment program or run program in a larger partition)
- 129 Root file is opened exclusively to another user
- 131 No activity table space available (more than 20 different data bases are open in the system) or the data base is already open to seven users
- 132 No resource number available
- 134 An attempt is being made to lock a data base that has been opened in mode = 8
- 135 Data base is locked to user and attempts to unlock it have failed
- 136 Data base is locked to another user (returned on lock without wait request)
- 137 Illegal resource number usage
- 140 Unable to schedule DBCOP
- 150 Data base already open to user
- 152 Data base open to another user in an incompatible mode
- 153 User has no access to any item or set in the data base with his given level
- 154 Data base corrupt — bad data path pointers in a data set
- 155 Beginning of chain or end of chain found on a chained read
- 156 Detail data set does not contain any entries along the specified chain
- 157 No current record for data set
- 159 User attempting to UPDATE, PUT or DELETE a record with an open mode of 1 without first locking the data base
- 160 IMAGE data structures are corrupt (could be caused by the structures being overwritten or by DBBUF not residing in the main segment)

## ERROR MESSAGES (Continued)

### IMAGE NUMBERED ERROR MESSAGES:

- 161 An internal subroutine passed DBCOP an illegal function code (could be caused by a user writing over a constant in an IMAGE segment)
- 162 Missing parameter in a library subroutine call
- 201 User passed DBBLD an illegal data base name
- 202 User passed DBBLD an incorrect security code
- 203 No data appears in the DBBLD input file between the first record and the \$END record
- 204 A data entry in the DBBLD input file is too short or omitted
- 205 A \$SET: or \$END record is missing or too many fields exist for a data entry in a DBBLD input file
- 206 A nonnumeric character or invalid number appeared in an integer data field in a DBBLD input file
- 207 A nonnumeric character appeared in a numeric data field in a DBBLD input file
- 208 An error has occurred during entry into a data base from a DBBLD input file and DBBLD cannot purge the data base
- 209 A physical end-of-file mark was encountered by DBBLD while processing an input file
- 210 The logical unit specified in the RUN command for one of the utilities (DBRST, DBSTR, DBLOD, DBUNL) was illegal
- 211 The user has failed to give the correct level word to DBRST, DBSTR, DBLOD or DBUNL
- 212 The input file presented to DBLOD or DBRST is of an illegal format
- 213 The user has failed to give the proper security code to DBRST, DBSTR, DBUNL or DBLOD
- 214 The root file and associated sets are on different subchannels during operation of DBSTR or DBRST
- 215 Either the file containing the data base root file or the file containing the data base cannot be found during operation of the DBSTR or DBRST routines

- 216 Either the root file size or the data base file size on the disc is not the same size as the root file and data base stored on the back-up file
- 217 The DBULD routine has discovered no data in the data base to unload
- 218 Illegal mode or column parameter
- 219 Not enough space to load root file
- 220 DBBLD has encountered an I/O abort
- 230 An attempt is being made to write to a tape that has no write ring
- 231 An attempt is being made to access a tape on a drive that is off-line
- 235 The utility is aborting because it reached the end of the FMP storage file without reaching the end of the data base and the user did not specify to continue
- 236 The utility is aborting because it reached the end of tape before reaching the end of the data base and the user did not specify to continue
- 241 The lu specified as the console lu is not interactive
- 242 The user mounted a reel with an incorrect reel number or specified a disc file out of sequence

## ERROR MESSAGES (Continued)

### IMAGE NUMBERED ERROR MESSAGES:

- 243 The root file name specified does not match the root file name on the storage device
- 244 The user specified the wrong cartridge to restore the data base (DBRST must load the data base back onto the same cartridge from which it came)
- 247 The program was terminated by the user entering the operator BR[EAK] command
- 248 An illegal word was entered as an abort option in the run string
- 302 Too many names in a NAME-LIST
- 303 Invalid name in NAME-LIST
- 304 The type or length of a variable in VALUE-LIST does not match the type or length of its corresponding item in NAME-LIST
- 305 Variable missing in VALUE-LIST
- 306 Invalid relative record number (nonnumeric)
- 310 Illegal IBASE parameter of a DBOPN call (does not start with two blanks)
- 324 Illegal DBINF request
- 7777 A catastrophic error has occurred in a utility — reload and rerun the program

## ERROR MESSAGES (Continued)

### DBDS ERROR MESSAGES:

#### MESSAGE

ALL PATHS TO MASTER ARE NOT DEFINED

AT LEAST ONE ITEM MUST BE DEFINED

AUTOMATIC MASTER MUST HAVE KEY ITEM ONLY

AUTO-MASTERS MUST HAVE AT LEAST 1 PATH

AUTO-MASTER NEEDS MORE WRITE CAPABILITY

BAD CAPACITY COUNT

BAD READ LEVEL

BAD TERMINATOR `;' EXPECTED

BAD TYPE DESIGNATOR

BAD WRITE LEVEL

`BEGIN DATA BASE' EXPECTED

CANNOT OPEN SCHEMA FILE

`CAPACITY:' EXPECTED

## MEANING

Not enough detail data sets defined a path to a master set.

The data base must have a minimum of one item defined.

An automatic master may have only one item defined and it must be a path item.

An automatic master data set must define a path item with a path count of at least 1.

An automatic master's write level must be equal to or greater than its related detail data set's key item (this is not true of manual masters).

The capacity count was not between 1 and  $2^{31} - 1$ .

The read level was not between 1 and 15.

A semi-colon was expected. DBDS scans until it finds a semi-colon before it continues processing.

An item has been defined as other than I1, R2, or Xn or a data set has been defined as other than Manual, Automatic, or Detail.

The write level on an item definition is not between 1 and 15 or is less than the read level.

The 'BEGIN DATA BASE:' command was not found.

DBDS cannot open the schema file because the wrong name was entered, the disc is not mounted or there was a *disc error*.

The 'CAPACITY:' command was expected.

DATA BASE HAS NO DATA SETS  
DATA SET FILES CREATED  
DATA SET MUST HAVE AN ITEM  
DUPLICATE ITEM NAME  
DUPLICATE LEVEL WORD  
DUPLICATE SET NAME  
END DATA BASE DEFINITION  
'END.' EXPECTED  
END OF FILE ENCOUNTERED  
  
'ENTRY:' EXPECTED  
ENTRY TOO BIG  
ILLEGAL CARTRIDGE NUMBER  
  
ILLEGAL CONTROL OPTION  
ILLEGAL LEVEL WORD  
ILLEGAL NAME  
  
ILLEGAL SECURITY CODE  
  
ILLEGAL SEPARATOR  
INCOMPLETE NAME



No data sets were defined for the data base.

All the data set files were successfully created.

Every data set must contain at least one data item.

The item name was already defined.

The level word was already defined.

The data set name was already used.

DBDS has terminated.

An 'END.' command was not found.

An end of file was unexpectedly found in the schema input file.

The 'ENTRY:' command was expected.

A data entry is larger than 4096 bytes.

The cartridge number is not an integer between 1 and 32767 or is not two ASCII characters.

A command found in the '\$CONTROL:' list was not legal.

An illegal character was found in the level word.

Some portion of the data base namr or some portion of the set namr is not legal.

The security code is not an integer between 1 and 32767 or is not two ASCII characters.

A comma, colon or parenthesis was expected.

A data base namr must have a security code.

## **ERROR MESSAGES (Continued)**

### **DBDS ERROR MESSAGES: (Continued)**

#### **MESSAGE**

**I/O ERROR - ALL DATA SETS NOT CREATED**

**ITEM NAME EXPECTED**

**ITEM TOO LONG**

**`ITEMS:' EXPECTED**

**KEY ITEMS NOT OF SAME LENGTH OR TYPE**

**LEVEL NUMBER ALREADY DEFINED**

**LEVEL OUT OF RANGE**

**`LEVELS:' EXPECTED**

**MASTER MUST HAVE A PATH**

**MAX ERRORS-SCHEMA PROCESSING**

**MISSING PROGRAM SEGMENTS**

**`NAME:' EXPECTED**

**NON-NUMERIC PATH COUNT**

**NOT ENOUGH MEMORY TO CREATE ROOT FILE**

**PATH ITEM CANNOT BE AN ARRAY**

## MEANING

There was an FMP error while trying to create a data set. The data set may have already existed or the disc was not mounted. Examine the disc to see which data sets were created — correct the disc error and retry DBDS.

A name was expected in an item definition.

An item has been defined with too many elements in its array or the n in an Xn item is longer than 255.

The 'ITEMS:' command was expected.

A key item in a detail data set is not the same type or length as the master's key item.

The level number was already defined.

A level number was not between 1 and 15.

The 'LEVELS:' command was not found.

All master data sets, both manual and automatic must have a key item defined. The path count for a manual master may be zero.

The number of errors is equal to the requested error count specified in the '\$CONTROL:' option list by the 'ERRORS=' command.

A segment to DBDS was not loaded.

The 'NAME:' command was expected.

The path count in a master data set is not an integer.

DBDS not loaded into a large enough partition. Increase the partition size using the SZ command.

An item which was an array was defined as a path item.

PATH TO MASTER ALREADY DEFINED

ROOT FILE CREATED

ROOT FILE NOT CREATED

ROOT FILE NOT CREATED - SCHEMA ERRORS

ROOT FILE OVERHEAD RECORD ERROR

SEARCH ITEM ALREADY DEFINED

`SETS:' EXPECTED

THE FOLLOWING ITEM(S) ARE UNUSED

TOO MANY DATA ITEMS

TOO MANY DATA SETS

TOO MANY ELEMENTS

TOO MANY PATHS

TOTAL ITEM LENGTH NOT INTEGRAL WORDS

UNDEFINED ITEM REFERENCED

UNDEFINED SET REFERENCED

UNEXPECTED `END.'

WARNING - SECURITY CODE IGNORED

`\$CONTROL:' EXPECTED

A detail data set has tried to define two paths to the same master data set.

The root file was successfully created.

The root file was not created because of the 'NOROOT' or 'NOSET' option in the '\$CONTROL:' list.

There were errors in the schema so the root file and all the data sets were not created.

There was an error writing the overhead record to the root file. Check for a possible disc error.

The master already has a path item defined. There can only be one path item per master.

The 'SETS:' command was not found.

The items that are listed were not used in any sets.

More than 255 data items have been defined.

More than 50 data sets were defined.

More than 255 elements were defined in an item definition.

More than 16 paths were defined in the master or detail data set or more paths from a detail were defined to a master than were allotted to the master.

An ASCII item defined without elements must have an even character count or the element count times the character count must be even.

An item was used in the set part that was not defined.

A set name was used in a detail data set's path definition that was not defined previously in the set part.

An 'END.' command was unexpectedly encountered.

A security code in a set namr is ignored. Data sets have the same security code as the data base.

The '\$CONTROL:' command was expected.

## ERROR MESSAGES (Continued)

### RECOV ERROR MESSAGES:

#### MESSAGE

BAD PROGRAM NAME

CLEAN UP UNSUCCESSFUL

RECOV INPUT ERR

RECOV OUTPUT ERR

UNABLE TO LOCK LIST LU

UNABLE TO OBTAIN CURRENT DATA  
BASE INFORMATION

### DBSPA ERROR MESSAGES:

#### MESSAGE

/DBSPA - ERROR <XXX> ON DATA SET

/DBSPA - ERROR <XXX> ON DBOPN

/DBSPA - ERROR ON INPUT <XXXX>

/DBSPA - ERROR ON OUTPUT <XXXX>

/DBSPA - ILLEGAL OR MISSING ROOT FILE NAMR

/DBSPA - UNABLE TO LOCK OUTPUT LU

/DBSPA - UNABLE TO OBTAIN INFORMATION ON  
DATA SETS

## MEANING

Program name misspelled or not in coordinating table.

DS error or program does not have data base opened.

I/O error from input LU.

I/O error to list device.

LU already locked.

Not enough SAM or DBCOP not loaded.

## MEANING

IMAGE error while accessing data set.

IMAGE error returned from DBOPN call.

I/O ERROR FROM INPUT LU

I/O error to list device.

Illegal or missing root file namr

LU already locked.

IMAGE error from DBINF call.

## QUERY ERROR MESSAGES:

### MESSAGE

A MASTER ENTRY WITH KEY VALUE EXISTS

ADD OVERFLOW

BAD DATA SET

BAD DATA SET OR DATA ITEM IN DBINF CALL

BAD OR MISSING SEGMENT

BAD TRACK IN WORK AREA

BATCH FILE ERROR

BEGINNING OR END OF CHAIN ENCOUNTERED

BREAK REQUESTED

CANNOT ADD OR AVERAGE AN ASCII VALUE

CANNOT ALTER THE VALUE OF AN ITEM

CANNOT EDIT REAL VALUES

CANNOT PERFORM A CHAIN READ

CARTRIDGE LOCKED

COMMAND TABLE OVERFLOW



## MEANING

Translation of IMAGE error 110.

The sum of the data item values exceeded 10 digits.

Data set specified does not exist in the data base being processed, or level word is not sufficient for data set.

Translation of IMAGE error 125.

A segment needed by QUERY does not have an ID segment.

A track in the work area of the disc is faulty or not available.

The batch file specified in the XEQ= command cannot be opened.

Translation of IMAGE error 155.

User entered an RTE BR[EAK] command while QUERY was processing a report command.

A TOTAL statement with an ADD or AVERAGE option specified an item that is type ASCII. QUERY will not perform arithmetic operations on ASCII fields.

Translation of IMAGE error 112.

A DETAIL, GROUP, or TOTAL statement contains a real data item, which cannot be edited.

Translation of IMAGE error 122.

Translation of FMP error 13.

Alter the REPORT procedure to contain less than 100 statements.

## ERROR MESSAGES (Continued)

### QUERY ERROR MESSAGES: (Continued)

#### MESSAGE

CONFLICT IN SST DEFINITION

CONSTANT LITERAL HAS EDIT OPTION

CONTROL BREAK INCONSISTENCY

DATA BASE ALREADY LOCKED TO ANOTHER USER

DATA-BASE ALREADY OPEN

DATA-BASE CANNOT BE CLOSED DUE TO LOCK

DATA BASE CORRUPT - BAD CHAIN POINTER

DATA BASE INACCESSIBLE WITH THAT LEVEL

DATA BASE MUST BE LOCKED

DATA BASE OPEN EXCLUSIVELY

DATA BASE NOT DECLARED

DATA BASE NOT ENABLED FOR LOCKING

DATA BASE NOT OPENED

DATA BASE RN IS BEING USED ILLEGALLY

DATA ITEM NOT MEMBER OF SET

DATA ITEM VALUE TOO LONG

DATA SET FOR OPERATION MUST BE A DETAIL

DATA SET FOR OPERATION MUST BE A MASTER

## MEANING

Translation of FMP error 39.

Alter the REPORT procedure to delete the edit mask reference or change the constant to a data item name.

The level numbers of a TOTAL, SORT, or GROUP statement do not match properly. Alter the REPORT to maintain control break consistency.

Translation of IMAGE error 136.

Translation of IMAGE error 150.

Translation of IMAGE error 135.

Translation of IMAGE error 154.

Level code insufficient.

Translation of IMAGE error 159.

Translation of IMAGE error 129.

Type a DATA-BASE= command, then reenter the QUERY command.

Translation of IMAGE error 134.

Translation of IMAGE error 103.

Translation of IMAGE error 137.

Data item specified is not a member of the data set being processed.

Data item value exceeds 255 characters.

Translation of IMAGE error 120.

Translation of IMAGE error 123.

DATA SET IS NOT WRITE ENABLED

DETAIL CHAIN IS EMPTY

DETAIL DATA SET HAS NO PATHS

DETAIL DATA SET IS FULL

DCB NOT OPEN

DETAIL LEVEL XX IS MISSING

DEVICE I/O ABORTED

DIRECTORY FULL

DISC CARTRIDGE NOT FOUND

DISC DOWN

DUPLICATE DATA ITEM IN SOFT STATEMENTS

DUPLICATE EDIT STATEMENT

DUPLICATE NAME

EDIT MASK TABLE OVERFLOW

XXXXXX ENTRIES FOUND

ERROR NO. XXX

FILE CURRENTLY OPEN OR EXCLUSIVE OR  
LOCK REJECT

FILE NOT FOUND

Translation of IMAGE error 118.

Translation of IMAGE error 156.

Translation of IMAGE error 121.

Translation of IMAGE error 105

Translation of FMP error 11.

DETAIL statements using level numbers must be contiguous starting at 1. A DETAIL statement using level 1 may be defined without defining a DETAIL statement with no level number.

QUERY had an I/O abort error.

Translation of FMP error 14.

Translation of FMP error 32.

Translation of FMP error 1.

Alter the REPORT to eliminate the double reference.

Alter the REPORT procedure to eliminate the double reference.

Translation of FMP error 2.

Alter the REPORT command to contain not more than 10 edit masks.

Number of entries selected by the FIND command.

An error occurred which is not in either of QUERY's translation tables. If the error number is greater than 99, see Table B-1 in the IMAGE manual for more information. If the error is less than 100, see the FMP File Manager Manual.

Translation of FMP error 8.

Translation of FMP error 6.

## ERROR MESSAGES (Continued)

### QUERY ERROR MESSAGES: (Continued)

#### MESSAGE

FILE SPECIFIED IS NOT A ROOT FILE

FILE EXPECTED

FIND PROCEDURE TOO LONG

GREATER THAN 255 EXTENTS

ILLEGAL ACCESS TO DATA SET

ILLEGAL ACCESS TO LU

ILLEGAL ACCESS TO SYSTEM

ILLEGAL CONTROL OPERAND

ILLEGAL DATA ITEM NAME

ILLEGAL DATA SET NAME

ILLEGAL DESTINATION LU

ILLEGAL INPUT PARAMETER

ILLEGAL LIST PARAMETER

ILLEGAL LOCK REQUEST

ILLEGAL LOG PARAMETER

ILLEGAL LU

ILLEGAL NAME

## MEANING

Translation of IMAGE error 116.

A procedure file was used in a FIND command that did not include the FIND command. Probable cause is an error in creating the procedure file or supplying the wrong procedure file name.

A FIND procedure contains more than 50 logical relationships.

Translation of FMP error 46.

User does not have write access to the data set. Reopen the data set using a higher level code word.

Translation of FMP error 20.

Translation of FMP error 19.

Internal error in QUERY's tables.

Data item name specified does not appear in the schema for the data base being accessed, or level code word is insufficient.

Data set name specified does not appear in the data base being processed, or level code word insufficient.

Translation of FMP error 21.

The input file declared in the RU,QUERY command cannot be opened.

The list file declared in the RU,QUERY command cannot be opened.

There was an error when QUERY tried to lock the list lu.

The log file declared in the RU,QUERY command cannot be opened.

Translation of FMP error 18.

Translation of FMP error 15.

ILLEGAL OPTION

ILLEGAL PATH MODIFICATION

ILLEGAL READ OR WRITE ON TYPE 0

ILLEGAL SECURITY CODE

ILLEGAL SELECT-FILE SIZE OR TYPE

ILLEGAL TERMINATOR

ILLEGAL TYPE OR SIZE = 0

INCONSISTENCY BETWEEN OPTIONS AND EDIT  
STATEMENTS

INPUT FILE ERROR - YYYYYYYYYY

INPUT MUST BE CONTAINED WITHIN MULTIPLE  
LINES OF 72 CHARACTERS

INPUT TOO LONG

INSUFFICIENT WORK AREA FOR SORT

INTERNAL BUFFERS ARE CORRUPT

INTERNAL DBCOP ERROR

INTERNAL IMAGE CALL IS INVALID



The option parameter in the RU,QUERY string did not begin with the letters "EC".

User tried to change a key item value in an UPDATE REPLACE command.

Translation of FMP error 17.

Translation of IMAGE error 117.

The select-file specified by the user existed, but is not a type 1 file, or is less than three blocks long.

An item value being entered must end with a semicolon. This error could also occur if too many values are entered for an array item.

Translation of FMP error 16.

A REPORT command specified an edit mask that did not exist, or an edit mask was used on a data item that is a type real.

File Manager error YYYYYYYYYY occurred when the RU,QUERY command was entered.

QUERY ignores commands of more than 72 characters per line.

The command entered exceeded 896 characters. Retype in a shorter form.

The area on the disc reserved for sorting data entries is not large enough to sort all of the desired entries. Use the FIND command to shorten the number of data entries found, reduce the number of items sorted, increase the number of system tracks available, or simply try again later when the system is less busy.

Translation of IMAGE error 160.

Translation of IMAGE error 161.

Translation of IMAGE error 162.

## ERROR MESSAGES (Continued)

### QUERY ERROR MESSAGES: (Continued)

#### MESSAGE

INVALID COMMAND

INVALID DATA ITEM NAME OR NUMBER

INVALID DATA SET NAME OR NUMBER

INVALID DATA ITEM VALUE OR TERMINATOR

INVALID DBINF MODE

INVALID LOGICAL CONNECTOR XXXXXX

INVALID MODE

INVALID # OF VALUES FOR RELATIONAL  
OPERATOR

INVALID OR MISSING SEARCH ITEM

INVALID PROCEDURE NAME

INVALID REQUEST

INVALID SECURITY CODE OR ILLEGAL WRITE  
ON LU

ITEM NOT MEMBER OF DATA SET OR DATA SET  
NOT SPECIFIED

LIST FILE ERROR

LU NOT FOUND IN SST

## MEANING

Command not recognized by QUERY.

Translation of IMAGE error 101.

Translation of IMAGE error 100.

A value is not enclosed in quotes, or a comma or logical connector is missing.

Translation of IMAGE error 124.

QUERY anticipated an AND, OR, or END and does not recognize the characters specified by XXXXXX.

Mode parameter entered was not an acceptable open mode value.

More than one value follows an IGT, ILT, or INLT where only one value is allowed.

Translation of IMAGE error 102.

Procedure name used in a command is invalid. Retype using the proper name.

Level code word or open access mode is illegal.

Translation of FMP error 7.

A qualified data item is not a member of the data set specified in the FIND command or a data item is a member of more than one data set and the desired data set is not specified.

The list file specified in the LIST= command could not be opened.

Translation of FMP error 40.

MASTER DATA SET IS FULL

MORE THAN 5 FIELDS ARE BEING TOTALED

MUST ENTER PATH ITEM VALUE

NEXT?

NO ACCESS TO DATA SETS

NO CURRENT CHAIN OR BAD RECORD NUMBER

NO CURRENT RECORD FOR OPERATION

NO MASTER ENTRY WITH KEY ITEM VALUE

NO RESOURCE NUMBER AVAILABLE

NO ROOM FOR DATA BASE ENTRY IN DBCOP

NON-NUMERIC IN INTEGER VALUE

NON-NUMERIC IN REAL OR INTEGER VALUE

NON-NUMERIC IN REAL VALUE

NOT ENOUGH ROOM ON DISC CARTRIDGE

OPEN MODE DOES NOT ALLOW THAT OPERATION

QUERY - OUTPUT FILE ERROR

PARTITION IS TOO SMALL

PROCEDURE NAME XXXXXX NOT FOUND

QUERY/1000 READY

Translation of IMAGE error 106.

TOTAL statements in a REPORT command reference more than five different data items.

User must enter values for path (or key) items in an UPDATE ADD command.

QUERY is ready for another command.

User does not have access to any data set in the data base.

Translation of IMAGE error 111.

Translation of IMAGE error 157.

Translation of IMAGE error 107.

Translation of IMAGE error 132.

Translation of IMAGE error 131.

A nondigit character was entered as a value for a type integer value for a type integer data item.

User entered a nonnumeric digit when a real or integer value was expected.

A nondigit character was entered as a value for a type real data item.

Translation of FMP error 33.

Translation of IMAGE error 104.

QUERY had a nonrecoverable file I/O error while trying to write to the list file or the log file.

Translation of IMAGE error 128.

Specified procedure does not exist.

Informs user that QUERY is executing.

## **ERROR MESSAGES (Continued)**

### **QUERY ERROR MESSAGES: (Continued)**

#### **MESSAGE**

**READ OR WRITE TO A RECORD NOT WRITTEN**

**RECORD HAS NOT YET BEEN FOUND**

**RECORD IS EMPTY**

**RELATIONAL OPERATOR INVALID**

**RELEASE TRACKS**

**REPORT CANNOT BE GENERATED DUE TO ERRORS**

**REPORT EXPECTED**

**REQUEST DIRECTED AT AN AUTOMATIC MASTER**

**RETRIEVAL FROM MORE THAN ONE DATA SET**

**ROOT FILE CANNOT BE FOUND**

**SAME LINES HAVE CONFLICTING REPORT OPTIONS**

**SELECT-FILE ERROR**

## MEANING

Translation of FMP error 5.

User attempted to UPDATE REPLACE, UPDATE DELETE, or REPORT without first using a FIND command.

Translation of IMAGE error 114.

User entered a relational operator other than IS, IE, INE, IGT, INGT, ILT, INLT or ISNOT.

More values are specified in a FIND command than can be stored in available system tracks. Either reduce the number of FIND data item values, make more system tracks available or retry when the system is less busy.

REPORT command generated contains errors.

A procedure file was used in a REPORT command that did not include a REPORT statement.

Translation of IMAGE error 108.

An attempt was made to execute a FIND command that would retrieve values from more than one data set.

Translation of IMAGE error 119.

Like statements in a REPORT command (such as all DETAIL statements of same level) have conflicting SPACE or SKIP options. All items or literals that print on one line must not space or skip more or less than other statements that print on that line.

An I/O error occurred while accessing the select-file.

SELECT-FILE NOT DECLARED

SELECT-FILE OVERFLOW

SERIAL READ IN PROGRESS

SOF OR EOF READ OR SENSED

SORT ERROR - XXXXXX

SORT LEVEL XX IS MISSING OR DUPLICATED

SORT VALUE SIZE EXCEEDS LIMITS

SYNTAX ERROR

UNABLE TO SCHEDULE DBCOP

UNOBTAINABLE OPEN MODE

UPDATE EXPECTED

USER ACCESS NOT HIGH ENOUGH

VALUE MUST HAVE QUOTES \_ ITEM IGNORED

VALUE TOO LARGE FOR PARAMETER

VALUE TOO LONG - ITEM IGNORED



User attempted to execute a FIND command before a select-file was specified. Enter the SELECT-FILE= command, then reenter the FIND command.

QUERY found more data entries than there is room for in the select file. A select file must be specified that is larger than the current one.

QUERY is reading each record in the data set in order to satisfy the requirements of a FIND command.

Translation of FMP error 12.

Numbered IMAGE error XXXXXX occurred while reading entries from the data base.

GROUP or TOTAL statements in a REPORT command reference sort levels that do not appear in a SORT statement.

Total SORT key size is longer than 80 characters.

Command is not in proper form.

Translation of IMAGE error 140.

Translation of IMAGE error 152.

A procedure file was used in an UPDATE command that did not include the UPDATE statement.

User did not have write access to the data set. Reopen the data base with a higher capability level.

A data item presented to QUERY must be enclosed in quotes.

Translation of FMP error 30.

Value entered for an UPDATE ADD or REPLACE command is longer than the length defined for the item.

## ERROR MESSAGES (Continued)

### IMAGE ERROR CODES AND QUERY MESSAGES:

NUMBERED CODE	ERROR MESSAGE
100	INVALID DATA SET NAME OR NUMBER
101	INVALID DATA ITEM NAME OR NUMBER
102	INVALID OR MISSING SEARCH ITEM
103	DATA BASE NOT OPENED
104	OPEN MODE DOES NOT ALLOW THAT OPERATION
105	DETAIL DATA SET IS FULL
106	MASTER DATA SET IS FULL
107	NO MASTER ENTRY WITH KEY ITEM VALUE
108	REQUEST DIRECTED AT AN AUTOMATIC MASTER
110	A MASTER ENTRY WITH KEY VALUE EXISTS
111	NO CURRENT CHAIN OR BAD RECORD NUMBER
112	CANNOT ALTER THE VALUE OF AN ITEM
113	ENTRY BEING DELETED HAS NON-EMPTY CHAINS
114	RECORD IS EMPTY
115	INVALID MODE
116	FILE SPECIFIED IS NOT A ROOT FILE
117	ILLEGAL SECURITY CODE
118	DATA SET IS NOT WRITE ENABLED
119	ROOT FILE CANNOT BE FOUND
120	DATA SET FOR OPERATION MUST BE A DETAIL
121	DETAIL DATA SET HAS NO PATHS
122	CANNOT PERFORM A CHAIN READ
123	DATA SET FOR OPERATION MUST BE A MASTER

124           INVALID DBINF MODE  
125           BAD DATA SET OR DATA ITEM IN  
              DBINF CALL  
128           PARTITION IS TOO SMALL  
129           DATA BASE OPEN EXCLUSIVELY  
131           NO ROOM FOR DATA BASE ENTRY  
              IN DBCOP  
132           NO RESOURCE NUMBER AVAILABLE  
134           DATA BASE NOT ENABLED FOR  
              LOCKING  
135           DATA BASE CANNOT BE CLOSED DUE  
              TO LOCK  
136           DATA BASE ALREADY LOCKED TO  
              ANOTHER USER  
137           DATA BASE RN IS BEING USED  
              ILLEGALLY  
140           UNABLE TO SCHEDULE DBCOP  
150           DATA BASE ALREADY OPEN  
152           UNOBTAINABLE OPEN MODE  
153           DATA BASE INACCESSIBLE WITH  
              THAT LEVEL  
154           DATA BASE CORRUPT - BAD  
              CHAIN POINTER  
155           BEGINNING OR END OF CHAIN  
              ENCOUNTERED  
156           DETAIL CHAIN IS EMPTY  
157           NO CURRENT RECORD FOR  
              OPERATION  
158           DATA ITEM IS NOT PART OF  
              GIVEN DATA SET  
159           DATA BASE MUST BE LOCKED  
160           INTERNAL BUFFERS ARE CORRUPT  
161           INTERNAL DBCOP ERROR  
162           INTERNAL IMAGE CALL IS INVALID



