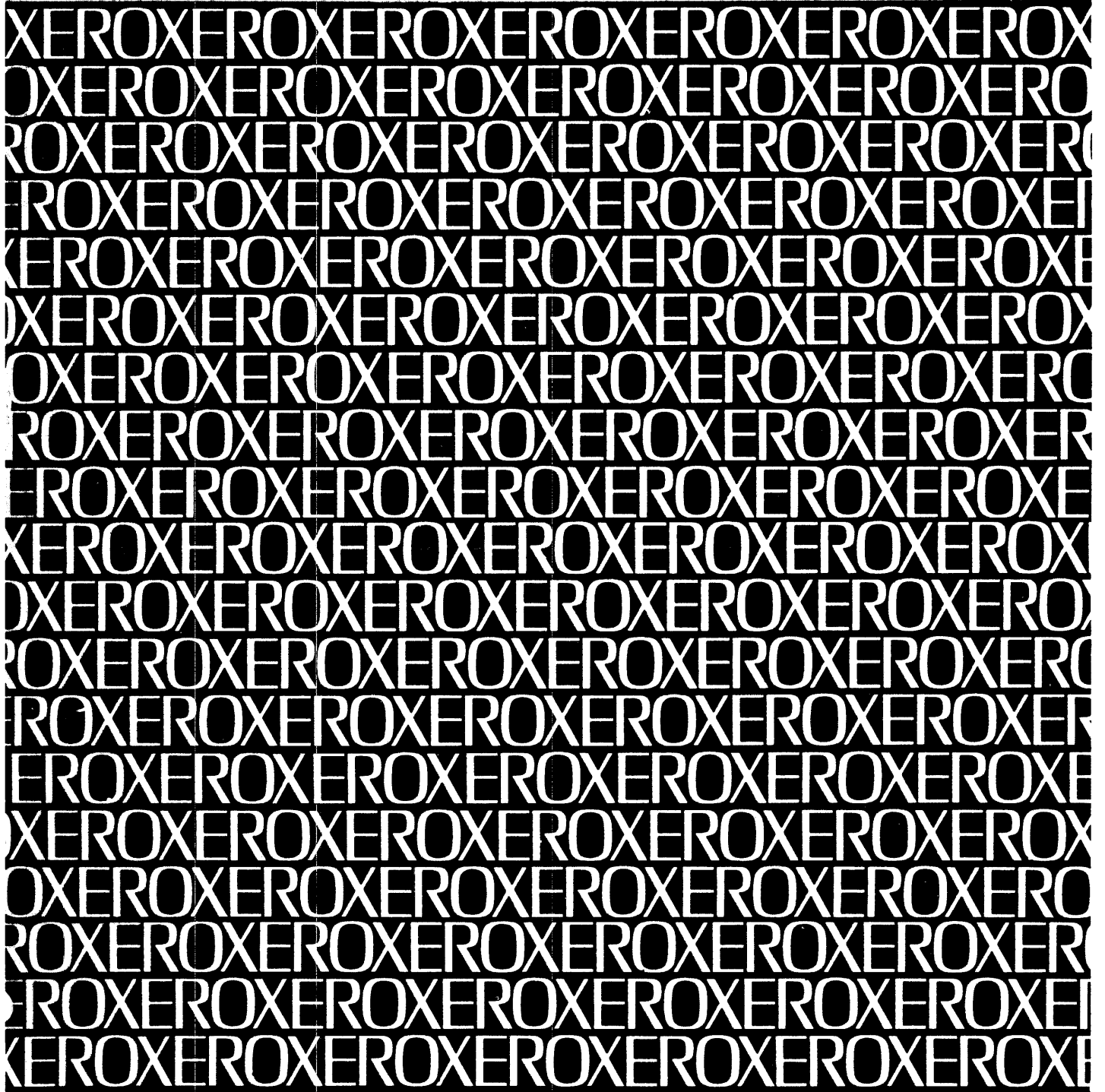


# Xerox Interactive Database Processor

Xerox 560 and Sigma 6/7/9 Computers

Language and Operations

Reference Manual



# **Xerox Interactive Database Processor (IDP)**

**Xerox 560 and Sigma 6/7/9 Computers**

## **Language and Operations Reference Manual**

90 30 66B  
90 30 66B-1

December, 1975

## REVISION

This publication is a revision to the Xerox Interactive Database Processor (IDP) Reference Manual, Publication Number 90 30 66A (dated December, 1973). This edition documents the B00 version of IDP for Xerox 560 and Sigma 6/7/9 computers. A change in the text from that of the previous manual is indicated by a vertical line in the margin of the page.

## RELATED PUBLICATIONS

<u>Title</u>	<u>Publication No.</u>
Xerox Sigma 6 Computer/Reference Manual	90 17 13
Xerox Sigma 7 Computer/Reference Manual	90 09 50
Xerox Sigma 9 Computer/Reference Manual	90 17 33
Xerox 560 Computer/Reference Manual	90 30 76
Xerox Extended Data Management System (EDMS)/Reference Manual	90 30 12
Xerox Extended Data Management System (EDMS)/User's Guide	90 30 37
Xerox Control Program Five (CP-V)/TS Reference Manual	90 09 07
Xerox Control Program Five (CP-V)/OPS Reference Manual	90 16 75
Xerox Control Program Five (CP-V)/TS User's Guide	90 16 92
Xerox Control Program Five (CP-V)/BP Reference Manual	90 17 64.

Manual Content Codes: BP – batch processing, LN – language, OPS – operations, RP – remote processing, RT – real-time, SM – system management, SP – system programming, TP – transaction processing, TS – time-sharing, UT – utilities.

The specifications of the software system described in this publication are subject to change without notice. The availability or performance of some features may depend on a specific configuration of equipment such as additional tape units or larger memory. Customers should consult their Xerox sales representative for details.

# CONTENTS

1.	INTRODUCTION	1	
2.	IDP OPERATIONS	3	
	On-Line Operations	3	
	Logging On	4	
	Logging Off	4	
	Break Control	5	
	On-Line Syntax Correction	5	
	Selected TEL Commands	5	
	Password Command	5	
	Platen Command	6	
	Batch Operations	7	
3.	IDP QUERY LANGUAGE	8	
	Character Set	8	
	Language Components	9	
	Keywords	9	
	Data-Names	10	
	Literals	10	
	Arithmetic Operators	10	
	Relational Operators	10	
	Logical Operators	11	
	IDP Commands	12	
4.	IDP COMMANDS	13	
	Format Notation	13	
	Punctuation Rules	13	
	IDP Retrieval Strategy	14	
	Multiple Report Specification	14	
	IDP Commands and Syntax Formats	15	
	QUERY Command	15	
	Format	16	
	DISPLAY Command	18	
	Format	18	
	Report Items	20	
	WHEN Clause	20	
	Literal	20	
	Database Item-name	21	
	Constraint List	21	
	Range	21	
	Arithmetic Expression	22	
	VIA Clause	23	
	SERIAL Clause	23	
	DIRECT Clause	23	
	PORT Clause	26	
	SORT Command	26	
	Format	26	
	COUNT Command	27	
	Format	27	
	TOTAL Command	27	
	Format	29	
	LIMIT Command	31	
	Format	31	
	OUTPUT Command	31	
	Format	31	
	Report Width	32	
	Default Width	33	
	Report Truncation	34	
	Report Length	35	
	REPORT Command	36	
	Format	37	
	TITLE Command	40	
	Format	40	
	Title Spacing	40	
	DEFINE Command	41	
	Format	41	
	Define Error Correction	44	
	DELETE Command	45	
	Format	45	
	PORTFLAG Command	46	
	Format	46	
	SHARE Command	46	
	Format	46	
	EJECT Command	46	
	Format	46	
	END Command	46	

## APPENDIXES

A.	IDP ERROR MESSAGES	47
B.	IDP LIMITATIONS	57
C.	RETRIEVAL STRATEGY	58

## FIGURES

1.	Overview of IDP System	2
2.	Example of Logging On and Logging Off IDP System	3
3.	Examples of Syntax Correction	6
4.	Creation of a Source File Using EDIT	36

## TABLES

1.	Default Widths	33
A-1.	IDP Error Messages	47

# 1. INTRODUCTION

This publication describes the Xerox Interactive Database Processor (IDP) query language, and general operations of the processor under control of Xerox Control Program-Five (CP-V). IDP is a shared processor which is provided for users of Extended Data Management System (EDMS) and offers the capability for on-line retrieval and display of data maintained within EDMS databases. It can be used for a multitude of applications and by people of varying degrees of proficiency. An IDP user need not have any knowledge of programming or of EDMS. All a user needs to know are the names of the items contained in the database. For a complete description of EDMS capabilities, see the Xerox Extended Data Management System (EDMS)/Reference Manual, Publication 90 30 12.

The IDP query language consists of keywords and operators that can be combined with database item-names and literals to form meaningful query statements. Of the fifteen basic commands in the IDP language, only two, QUERY and DISPLAY, are necessary to produce a basic report. The other commands are optional and provide additional capabilities, such as for sorting a report or for accumulating counts and totals.

Consider, for example, an educational environment where the database contains information on instructors and students. An IDP user can sit down at a terminal and cause a class roster for each instructor to be generated almost instantaneously. Each roster could contain information such as instructor's name, course name and number, number of units, building and room number, time and days of the classes, along with a complete list and a count of all the students enrolled in each course.

As another example, consider a typical industrial environment. Here IDP can be used to generate reports showing the names of all employees, their hire dates, salaries, date of last salary increase, etc., of a particular department.

In addition to such general purpose reports, IDP can obtain selective information by using logical expressions (e.g., equal to, greater than, less than) in the query. Example: Find the names of all the employees in the engineering department who possess the job title "Senior Engineer", and who have ten or more years of experience, and whose weekly salary is less than \$350.

IDP cannot be used to examine every combination of data that may be defined in an EDMS database. An EDMS database may consist of several independent portions or areas with no links to each other. IDP requires that any cited group must be reachable from any other cited group by following set linkages. Even where such linkages exist in the database and are defined in the subschema, IDP may not be able to identify the path under certain circumstances. In such a situation, the user is notified and may restructure his request. Appendix B describes this restriction in greater detail.

To query an EDMS database, the following files must be available:

1. Database area files containing the desired information.
2. Subschema file containing a name table.

Figure 1 shows an overview of the IDP system.

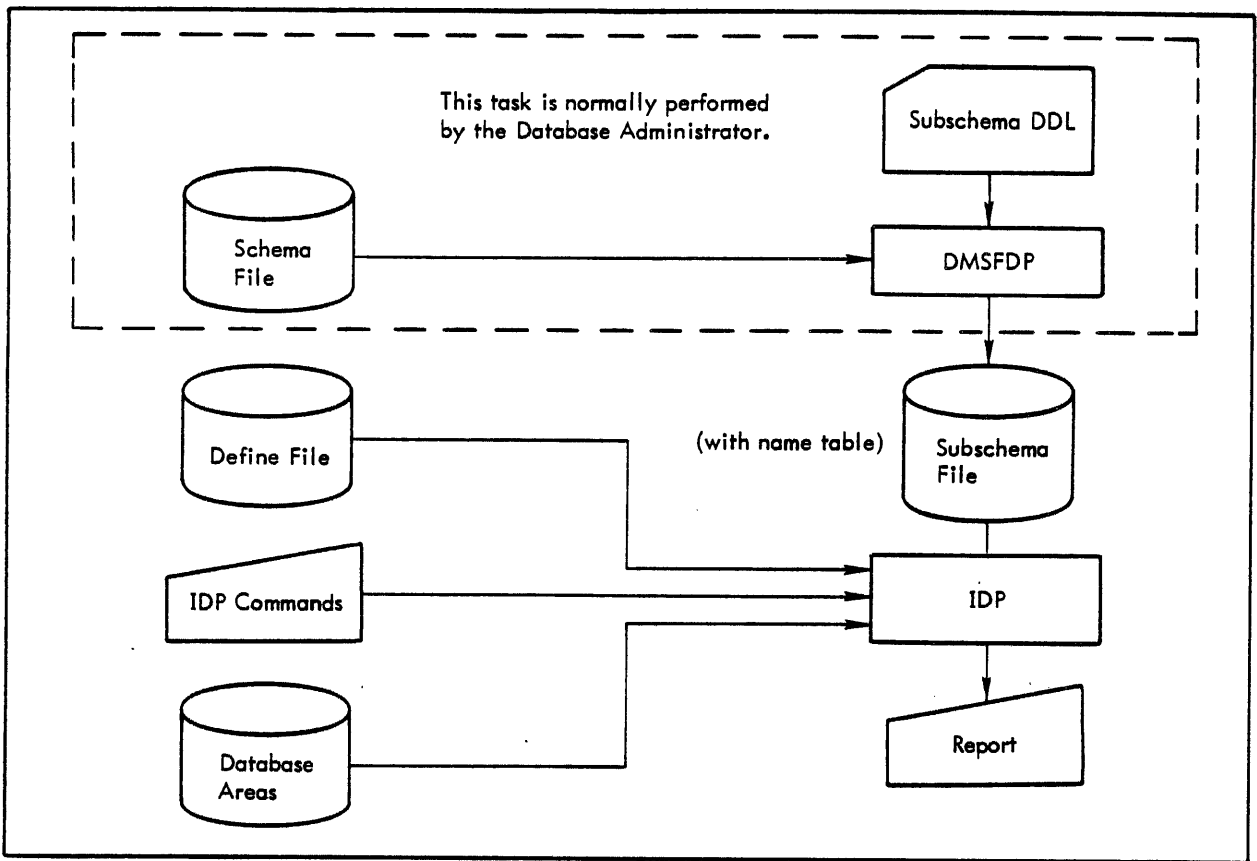


Figure 1. Overview of IDP System

## 2. IDP OPERATIONS

This chapter describes the operational procedures involved in running IDP either on-line or batch under Xerox Control Program-Five (CP-V).

### On-Line Operations

On-line time-sharing operational procedures are described in more detail in Xerox Control Program-Five (CP-V)/TS Reference Manual, Publication 90 09 07 and in Xerox Control Program-Five (CP-V)/TS User's Guide, Publication 90 16 92.

Figure 2 shows an annotated sample of an on-line IDP session.

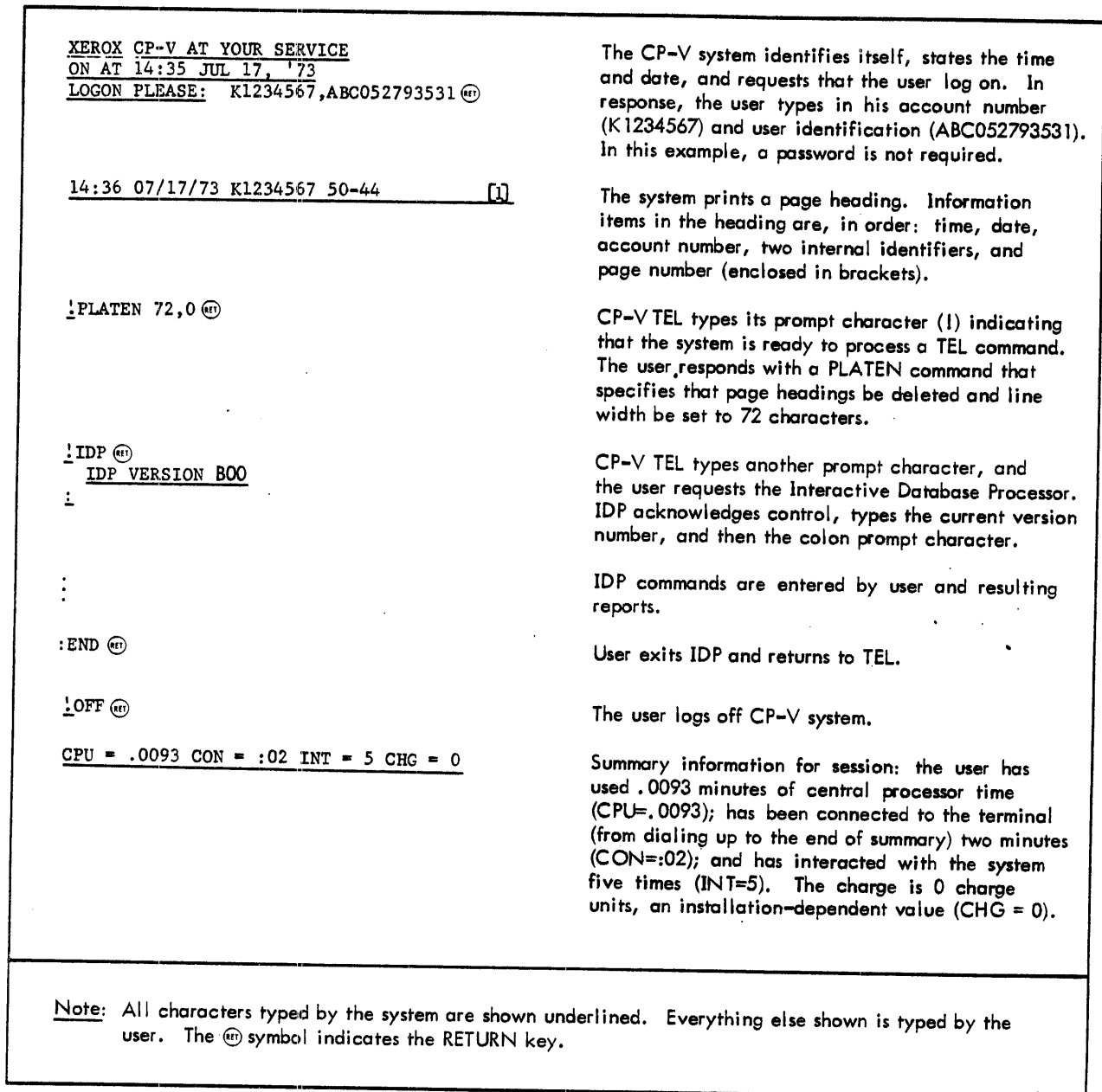


Figure 2. Example of Logging On and Logging Off IDP System

## Logging On

The user obtains on-line service from CP-V by activating a terminal and logging on. When connection with the computer is established, CP-V responds with the following message (shown underlined, as are all terminal messages throughout this chapter):

```
XEROX CP-V AT YOUR SERVICE  
ON AT (time and date)  
LOGON PLEASE:
```

CP-V then waits for entry of user identification which may consist of account, name, and password.

Account and password each consists of one to eight alphanumeric characters, whereas name consists of 1 to 12 alphanumeric characters. A comma must separate each item. The optional password and preceding comma are omitted if the account is not password-protected.

If the user's identification is valid, CP-V issues a page heading that contains various identifying information. CP-V then issues the exclamation mark (!) prompt character indicating that the user may now communicate with the Terminal Executive Language (TEL) of CP-V. The user requests the Interactive Database Processor (IDP) as follows:

```
!IDP (Ⓢ)  
  IDP VERSION B00  
:
```

The colon (:) prompt character indicates that the user may proceed with the on-line session by entering IDP commands such as QUERY, DISPLAY, etc.

A colon (:) prompt character will be issued on each new line with the exception of text string continuations (a text-string is a field enclosed with quotes); in which case, IDP prompts with an ampersand (&) until the end of the text string, indicated by a close quote.

### Example:

The following specifications are equivalent. In the first specification, the text string for the TITLE command is on two lines. The ampersand prompt character reminds the user that a close quote for the text string has not been found. In the second specification, the text string is on one line and no ampersand prompt character is issued.

```
!IDP (Ⓢ)  
  IDP VERSION B00 (Ⓢ)  
:QUERY HOSPSUB (Ⓢ)  
:TITLE 'SEPTEMBER INVENTORY (Ⓢ)  
& REPORT' (Ⓢ)  
:DISPLAY UNITCOST (Ⓢ)  
:
```

Note the space after & is as a result of input and not implied by previous carriage return.

```
!IDP (Ⓢ)  
  IDP VERSION B00  
:QUERY HOSPSUB (Ⓢ)  
:TITLE 'SEPTEMBER INVENTORY REPORT' (Ⓢ)  
:DISPLAY UNITCOST (Ⓢ)  
:
```

## Logging Off

The user may exit from IDP (and return to TEL) by either entering the END command, or "end-of-file" on teletype-compatible terminals, by depressing ESC and F keys in sequence; on Model 2741 terminals, by depressing F and ATTN keys in sequence. See Chapter 2, Terminal Operations, of the Xerox Control Program-Five (CP-V) Time-Sharing Reference Manual, Publication 90 09 07 for a description of the differences between the teletype and 2741 terminals.



When TEL gains control, it issues the exclamation mark (!) prompt character. The user may log off by simply entering OFF or BYE:

!OFF<sup>(RT)</sup> or !BYE<sup>(RT)</sup>

After the log-off, the system will display a summary of accounting information for the on-line session. The summary consists of

CPU minutes of central processor time used.  
CON hours of terminal connect time used.  
INT number of interactions with the system.  
CHG number of charge units for the session.

### Break Control

The user may halt the execution of his job by depressing the BREAK key. IDP will issue the following message:

\*\*\*53\*\*\* QUIT CONTINUE OR RESTART (Q/C/R)?

"Q" will terminate the IDP session and return to TEL. "C" will continue execution of the interrupted request. "R" will ignore the present request and accept a new request.

### On-Line Syntax Correction

Syntax checking is performed on a line-by-line basis. If the command contains an error, the word in error is flagged with a dollar sign (\$), and an appropriate error message is issued. The procedure for syntax correction depends on the message number, as shown below.

<u>Message Number</u>	<u>Procedure</u>
1 to 99	Retype only the portion of the command beginning with the word in error.
100 to 116	Retype the entire command or the affected clause, beginning with the keyword.
117 to 199	As specified in Appendix A.
200 and above	IDP returns to TEL.

Figure 3 illustrates some examples of syntax correction. See Appendix A for a complete list of error messages.

### Selected TEL Commands

The PASSWORD and PLATEN commands of Terminal Executive Language (TEL) are described below.

#### PASSWORD Command

A password is an optional part of the information by which a user identifies himself when logging onto the system. The purpose of a log-on password is to allow a user to protect his account number. The PASSWORD command allows the user to change his password as he wishes to protect the privacy of his password. The format of the command is

PASSWORD (password)

where password is the word to be uniquely associated with the user's name and account number. It must be one to eight characters in length and may consist of any of the following:

A-Z, a-z, 0-9, -, \$, \*, %, :, #, @, -, backspace

Once specified, the password must be used when logging on until it is changed or cancelled. If the password command is used without a specified password, the current password is cancelled.

```

!IDP(M)
  IDP VERSION B00

:QUERY INSTSUB.(M)
  1: QUERY INSTSUB.
      §

```

The word QUERY was mistyped. The command must be retyped.

```

***100*** UNABLE TO DECIPHER THIS COMMAND

:QUERY INSSUB.(M)
  2: QUERY INSSUB.
      §

```

The subschema name INSTSUB was misspelled. Only the name needs to be retyped.

```

***1*** SUBSCHEMA NAME IN ERROR

:INSTSUB.(M)
  3: INSTSUB.
      §

```

A required database password was omitted. The command must be retyped.

```

***109*** PASSWORD REQUIRED FOR THIS DATABASE; RETYPE QUERY

:QUERY INSTSUB PASSWORD = 'PW$$INST'.(M)
:SORT CRSNAME(M)
:COUNT.(M)
  6: COUNT.
      §

```

Note that the SORT command was not terminated by a period. Thus, the word COUNT was treated as the name of the second sort key.

```

***9*** THIS ITEM NAME NOT FOUND IN THE SUBSCHEMA

:.(M)

```

This period terminates the SORT command. (Note that line 6 is therefore deleted and must be retyped.)

```

:COUNT.(M)
:DISPLAY CRSNAME WHEN INST-NO = A5279.(M)
  9: DISPLAY CRSNAME WHEN INST-NO = A5279.
      §

```

INST-NO is an alphanumeric item; therefore, the literal must be enclosed in quotes. The absence of quotes indicates A5279 is an item-name.

```

***9*** THIS ITEM-NAME NOT FOUND IN THE SUBSCHEMA

:'A5279'.(M)

```

All commands are now syntactically correct. IDP will proceed to generate the desired report.

Figure 3. Examples of Syntax Correction

**PLATEN Command**

The IDP OUTPUT command enables the user to specify the desired page width and page length of his report (see Chapter 4, "Description of IDP Commands"). If output is to the terminal, automatic line folding can be accomplished by means of the width option on the PLATEN command, described below.

The PLATEN command can be used to change the page width or length, or both, for terminal input and output. The format of the command is:

PLATEN[w][,l]

where

- w is the maximum number of characters to be written per line on the terminal. If more than w characters are written, a line feed and carriage return character sequence is inserted to break up the output into segments no longer than specified by w. If w is 11 or less, no line feed and carriage return sequence is supplied. In this case, the width of the line is limited only by the physical constraints of the device on which the line is produced (up to a maximum of 140 characters). If the w field is omitted, the current width setting is retained.
- l is the number of lines per page of terminal output and must be within the range 0-256. If the l field is omitted, then the number of lines per page remains unchanged. If l is set to 11 or less, no system heading is produced and the page length is unlimited.

The length option of the PLATEN command controls the system heading only. This should not be confused with report headings which are generated by IDP based on specifications of the IDP OUTPUT command. Normally, the length option of the PLATEN command should be omitted.

## Batch Operations

Batch operational procedures are described in more detail in the Xerox Control Program-Five (CP-V)/BP Reference Manual, Publication 90 17 64.

When IDP runs in batch, no user interaction is provided. If a syntactical error exists, it is flagged with an appropriate error message and the request is aborted. Similarly, if any user-specified limits (LIMIT command) is exceeded, the request is aborted.

Report headings are generated in the same manner as for on-line mode. The default output will be to line printer; the default length of each page will be taken from the monitor.

In batch mode the user's source input will be listed on the LO device, along with the generated report. If the user does not want the source input listed, the NOLIST option on the IIDP card may be specified.

Format:

IDP [NOLIST]

Example:

```
IIDP NOLIST
QUERY SCHOOLSUB.
OUTPUT LENGTH = 38.
DISPLAY STUDENT, SPONSOR,
      SCHOLARSHIP-AMT,
      WHEN SCHOLARSHIP-AMT GE 2500.00.
END.
```

### 3. IDP QUERY LANGUAGE

The IDP query language is composed of simple English words, arithmetic operators, relational operators, and logical operators which can be combined with data-names and literals to form meaningful queries.

#### Character Set

The character set of the IDP language consists of

- Alphabetic characters: A through Z
- Numeric characters: 0 through 9
- Special characters:
  - (space or blank)
  - + (plus sign; addition)
  - (minus sign or hyphen; subtraction)
  - \* (asterisk; multiplication)
  - / (slash; division)
  - = (equal)
  - < (less than)
  - > (greater than)
  - & (ampersand; text string continuation or logical AND)
  - ( (left parenthesis)
  - ) (right parenthesis)
  - ' (apostrophe or single quote mark)
  - ,
  - : (colon; prompt character)
  - ;
  - ? (question mark)
  - .
  - | (vertical slash; inclusive OR)
  - ~ (tilde; not)
  - \$ (dollar sign; error flag)

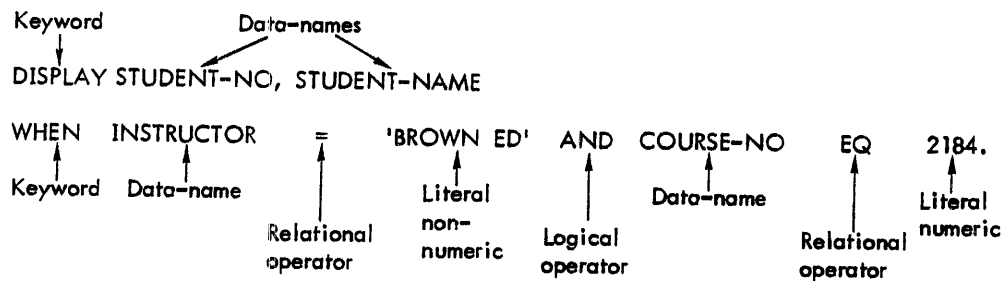
Nonnumeric literals may contain any characters including characters not in the set.

## Language Components

Components of the IDP language include

- Keywords
- Data-names
- Literals
- Arithmetic operators
- Relational operators
- Logical operators

The example below identifies the various components in a typical query statement:



### Keywords

A keyword is an English word in an IDP command that specifies a particular function. Every IDP command, and every clause within a command, begins with a keyword. All IDP keywords are listed below, including the reserved keywords VIA, WHEN, SERIAL, and DIRECT, which may not be referenced as data-names in a DISPLAY command.

<u>Keywords Identifying an IDP Command</u>	<u>Other Keywords</u>	
COUNT	AREA	OF
DEFINE	ASC	ON
DELETE	AVG	PASSWORD
DISPLAY	CIPHKEY	PORT
EJECT	DES	SERIAL
END	DIRECT	SIZE
LIMIT	IN	SN
OUTPUT	LENGTH	TO
PORTFLAG	LP	VIA
QUERY	MAX	WHEN
REPORT	ME	WIDTH
SHARE	MIN	
SORT		
TITLE		
TOTAL		

## Data-Names

A data-name can be a database item-name, a group-name, or a set-name as specified in the schema. A data-name consists of a string of not more than 30 characters selected from the letters A through Z, the digits 0 through 9, and the hyphen, but may not begin or end with a hyphen. A data-name must have at least one nonnumeric character. Every data-name used in a query must be defined in the subschema specified in the QUERY command. ...

## Literals

A literal is either numeric or nonnumeric. A numeric literal is a string of characters selected from the digits 0 through 9 (to a maximum of 31 digits) and, optionally, the plus or minus sign and a decimal point. An unsigned numeric literal is assumed to be positive. A numeric literal may also be described as a floating-point constant in "E" or "D" notation for single and double precision, respectively. Examples of numeric literals are

123      +123      123E0      1.23E02      .123E+3      1230D-01  
-45.67    -45.67D0    -4.567E1    -.4567E+02

A nonnumeric literal is a string of up to 255 EBCDIC characters enclosed in a pair of apostrophes (single quote marks). To include one apostrophe within a literal, two apostrophes must be used. Examples of nonnumeric literals are

'123'  
'MATH 103'  
'O"BRIEN'  
'\$\*\*/&+'

## Arithmetic Operators

An arithmetic operator specifies the type of operation to be performed between two operands in an arithmetic expression. In an expression, the operator must be preceded by a space and followed by a space. The allowable arithmetic operators are

<u>Arithmetic Operator</u>	<u>Meaning</u>
+	Addition
-	Subtraction
*	Multiplication
/	Division

## Relational Operators

A relational operator specifies the type of comparison to be made for two operands. Each relation is composed of

1. A database item-name (may be qualified with a group-name).
2. A relational operator.

3. One of the following:
  - a. A literal.
  - b. A database item-name (may be qualified with a group name).
  - c. A constraint list (a list of criteria separated by commas).
  - d. A range of constraint values.
  - e. An arithmetic expression.

The allowable relational operators are shown below.

<u>Relational Operator</u>	<u>Meaning</u>
EQ or =	equal to
NE or ~ =	not equal to
GT or >	greater than
LT or <	less than
GE or >=	greater than or equal to
LE or <=	less than or equal to

### Logical Operators

A logical operator is used to combine two or more relations. The allowable logical operators are

<u>Logical Operator</u>	<u>Meaning</u>
AND or &	Logical conjunction
OR or	Logical inclusive or

A relation (e.g., INSTRUCTOR = 'BROWN ED') is either true or false, just as a logical expression, consisting of multiple relations, is either true or false. The truth table below shows the various logical combinations and their resultant values.

IF		THEN	
RELATION		LOGICAL EXPRESSION IS	
a	b	a AND b	a OR b
true	true	true	true
false	true	false	true
true	false	false	true
false	false	false	false

An expression is considered "satisfied" if the expression is true. Thus, the expression "relation-1 AND relation-2" is satisfied if both relation-1 and relation-2 are true. The expression "relation-1 OR relation-2" is satisfied if either relation-1 is true, or if relation-2 is true.

## **IDP Commands**

IDP consists of fifteen commands. The keywords of these commands and their functions are shown below.

- COUNT – Specifies that the number of report lines is to be printed at the end of the report.
- DEFINE – Defines a synonym to be used in place of valid IDP words, names or phrases.
- DELETE – Deletes synonym previously described in a DEFINE statement.
- DISPLAY – Identifies report items, defines the selection criteria and specifies the format of a report.
- EJECT – Controls page eject after subtotals.
- END – Terminates the session and returns to TEL.
- LIMIT – Limits production of a report with respect to the number of report lines or CPU time that may be expended.
- OUTPUT – Directs the report to the line printer or to a file and, optionally, specifies the report dimensions.
- PORTFLAG – Prevents IDP from arbitrarily picking a PORT group.
- QUERY – Identifies the EDMS subschema, the database password; and the enciphering keys for each area.
- REPORT – Produces a report from a source file.
- SHARE – Sets file sharing mode.
- SORT – Identifies those items which are sort control fields in the report.
- TITLE – Specifies a title to be printed at the top of each report page.
- TOTAL – Identifies those report items whose totals are to be printed.



## 4. IDP COMMANDS

This chapter describes in detail the capability, and the required syntax format, of each IDP command. Format notation and punctuation rules are described below.

### Format Notation

The following notation applies to all formats:

1. An underlined word in uppercase is required if the part of the format containing it is used.
2. Uppercase words not underlined are optional, but are legal only in the indicated positions.
3. Words in lowercase represent names or values that are supplied by the user.
4. Brackets indicate that the enclosed part of the format is optional. If two or more language elements are vertically stacked within brackets, none of the elements is required and no more than one may be included. For example,

$\left[ \begin{array}{l} a \\ b \\ c \end{array} \right]$  signifies a, or b, or c, or none.

5. Braces indicate a required choice. Of the two or more elements vertically stacked within braces, only one may be used, and one is required. For example,

$\left\{ \begin{array}{l} a \\ b \\ c \end{array} \right\}$  signifies a, or b, or c.

6. An ellipsis indicates that repetition is allowed. The portion of a format that may be repeated is the total enclosed element whose outermost right bracket or brace immediately precedes the ellipsis. For example,

$\left[ [a b] c \right] \dots$  The whole sequence a b c may be repeated.

$\left[ [a b][c] \dots \right]$  Only c may be repeated.

### Punctuation Rules

The space, the comma, and the period are considered punctuation marks (except in literals) and are used as follows:

1. The space (blank) is a separator, required after words and literals in the absence of any other separator. A space may precede or follow any other separator, and two or more consecutive spaces are the same as one except in nonnumeric literals).
2. A comma is a separator that is legal only where it is specifically indicated in a language format. The comma, where it is legal, can also serve as a terminator for words and numeric literals. The comma is never required, except to separate criteria in a constraint list.
3. The period is required as the last character in an entry. Thus, the period (followed by carriage return) is considered as the command delimiter. The period also has two other functions: as a decimal point in a numeric literal, and to separate file-name, account, and password in the file identifier for the QUERY and REPORT commands.

Continuation of lines is permitted on the terminal. That is, an IDP command may consist of more than one line.

## IDP Retrieval Strategy

On analyzing and responding to a request, IDP goes through the following sequence of steps:

1. Checks the request for correct syntax. At the same time, the data-names cited in the request are checked against those in the subschema specified on the QUERY command. IDP verifies that names used in the request appear in the subschema as item-names, that group-name qualifiers are supplied where required, etc. Discrepancies are immediately brought to the attention of the user for resolution. (See Appendix A, IDP Error Messages.)
2. Analyzes the request to determine which groups and sets are required to respond to the request, and if the required groups and sets form a continuous path for retrieval. The first step on this process is to designate a "port" group. This group is the first one to be retrieved and serves as the entry point or the head of the network of groups and sets required by the request. The order in which groups are examined as potential port groups is as follows:
  - The group specified by the user in the SERIAL, DIRECT, or PORT clause of the DISPLAY command.
  - A group for which a value (or values) for the calc item (or items) have been cited as retrieval constraints (or selection criteria).
  - A group for which a value (or values) for an item upon which the group is inverted has been supplied as retrieval constraints.
  - The area-as owner group, if one was defined in the subschema.
  - An arbitrarily selected cited group to be used in a serial search if none of the above conditions apply.

A "cited" group is one that contains an item to be displayed (a "target" item, hence a "target" group), or an item to be used in the selection process (a "constraint" item, hence a "constraint" group), or both, or a group that has been specified by the user as the serial search port group. Additionally, the required network may contain 'don't care' groups. These groups do not contain any information of interest to the user, but are needed to connect the target or constraint groups or both, into a continuous path.
3. Generates code for the retrieval function of IDP.
4. Opens the appropriate area(s) of the database and calls upon the Database Manager (DBM) to get the initial occurrence of each cited group. The 'don't care' groups are accessed by the DBM to locate the necessary set pointers but are not read into the data buffer. One occurrence of each cited group comprises a 'logical record'.
5. Compares the values for the constraint items in the logical record with the request selection criteria.
6. If the constraint item values in the logical record meet the selection conditions, IDP gives the target item values to the display program (if no SORT is required) or writes them into a file for subsequent sorting by the display program.
7. After the display program has listed the target items or after the target items have been written into the sort file, or if the logical record did not qualify for selection, IDP reads in one or more group occurrences, depending upon the request, to form a new logical record and resumes the process at step 5.

## Multiple Report Specification

Multiple reports may be specified in a session. Each DISPLAY command defines a report and initiates the report production. Upon completion of that report, control returns to the user and additional commands may be entered for the next report, or an END or end-of-file may be keyed in to terminate the session.

When a session consists of multiple DISPLAY commands, the following rules apply:

1. The QUERY, LIMIT, OUTPUT, EJECT, and PORTFLAG commands apply to every report, unless they are respecified prior to the next DISPLAY command.
2. The SORT, COUNT TITLE, and TOTAL commands apply only to one report. They must be respecified for each subsequent report.

As an example, assume the following on-line session:

```
IIDP (M)
:QUERY UNIVSUB. (M)
:LIMIT TIME = .05. (M)
:OUTPUT ON LP. (M)
:SORT INST-NAME. (M)
:COUNT. (M)
:TOTAL TENURE. (M)
:DISPLAY INST-NO, INST-NAME, TENURE. (M)
```

After this first report is completed, the user specifies the second report as follows:

```
:LIMIT TIME = .30. (M)
:DISPLAY STUDENT-NO, STUDENT-NAME, ADDRESS WHEN CLASS = 4. (M)
:END. (M)
```

Thus, the second report will use the same subschema and also be produced on the line printer. However, it will have a different time limit, will not accumulate a count of the number of report lines, nor a total on TENURE, and will not be sorted.

## IDP Commands and Syntax Formats

**QUERY Command** The QUERY command specifies the user's subschema file name which is checked for validity. If a password is required the user must supply it. Optionally, it may also specify the area-names, the enciphering keys and the user's define file. The QUERY command must be specified prior to any command that references a database item-name, namely, the DISPLAY, SORT, and TOTAL commands.

Since the subschema contains the names of the areas associated with it, it is necessary to give the area-names only if the area

1. Was created under an account different from that of the user currently running IDP.
2. Has a system password.
3. Is on a private volume.
4. Has an enciphering key.

It is assumed that the subschema named in the QUERY command has been created by a Database Administrator, (defined as the person or group of persons responsible for the structure and, in general, the context of the database). It is likely that there are several subschemas, each describing a portion of the total database. Initially, it may be necessary to consult with the Database Administrator to determine which subschema to use, which data-names to use, the information pertinent to points 1-4 above, etc.

At least one QUERY command is required for each IDP session. If all the requests can be satisfied with the same subschema, then only one QUERY command is needed. If additional reports are to be generated that require another portion of the database, as defined in a different subschema, another QUERY command is necessary.

Format:

QUERY

```
subschema-name [. [account-1] [.file-password-1]] [, SN = value-1]
[, PASSWORD = 'database-password']
[, AREA = area-name [. [account-2] [.file-password-2]] [, SN = value-2 [, value-3] ...]
[, CIPHKEY = 'enciphering-key'] ...
[, DEFINE FILE = define-file-name [. [account-3] [.file-password-3]] [, SN = value-4] .
```

where

**subschema-name** is the name of the subschema that contains information to access the database.

**account-1** is the account under which the subschema was created. Required only if different from the current account the user is logged on to.

**file-password-1** is the system password for the subschema. Required only if a password exists.

**value-1** is volume serial number of the disk pack that contains the subschema. Required only if the subschema is on a private volume.

**database-password** is a nonnumeric literal (enclosed by single quote marks) of up to eight characters that is a password of the database.

**area-name** is the file name of the area to be accessed. It is required only if the options following the AREA clause are required. The maximum number of areas is four.

**account-2** is the account under which the area was created. Required only if the account is different from the current account the user is logged on to.

**file-password-2** is the system password for the area. Required only if a password exists.

**value-2, value-3** are volume serial numbers of the disk pack(s) that contain the area. Required only if the area is on private volume(s).

**enciphering-key** is the enciphering key for that area. It is expressed as a nonnumeric literal (enclosed by single quote marks) and is required if the area is enciphered.

**define-file-name** is the name of the define file that contains synonyms for IDP names, words or phrases. If define-file-name is not specified, the define-file-name defaults to IDPDEFINE.

**account-3** is the account under which the define file was created. Required only if different from the current account the user is logged on to. If it is different, the user may not use the DEFINE or DELETE command unless the user has write access to that account or file.

**file-password-3** is the system password for the define file. Required if and only if a password exists.

**value-4** is the volume serial numbers of the disk packs that contain the define file. Required if and only if the define file is on a private volume.

If the user wishes to maintain the security of the passwords or enciphering key, CP-V provides a method for suppressing the echoing (typing on the user's console) of these or any character strings through the ESC-E sequence. This sequence operates like a toggle switch; that is, successive uses of ESC-E turn the echo suppression off if it is on or turns it on if it is off.

Example 1:

QUERY INSTSUB.

This is the simplest form of the QUERY command. It identifies the subschema name as INSTSUB. The subschema, database areas, and define file reside on public disk storage, and they were created under the same account as the user's log-on account. No database-password, file-passwords, or enciphering keys are required for this IDP session.

Example 2:

QUERY INSTSUB.ACCT1234.

This is similar to Example 1, except that the subschema was created under account ACCT1234, which is different from the user's log-on account. This example assumes that the database areas are in the same account as the user's log-on account.

Example 3:

QUERY INSTSUB.ACCT1234.SESAME.

This is similar to Example 2, except that the system password SESAME was used in creating the subschema.

Example 4:

QUERY INSTSUB..SESAME.

This is similar to Example 3, except that the account number is omitted because it is the same as the user's log-on account. Note that the period normally preceding the account number must be included if a system password is specified.

Example 5:

```
QUERY INSTSUB.ACCT1234.SESAME SN = PK27
    PASSWORD = 'PW$$INST'
    AREA = INSTAREA.G957153 SN = 5472
        CIPHKEY = 'BETA'
    AREA = STUAREA..SOCRATES SN = STU1,STU2
        CIPHKEY = 'IOTA'
    DEFINE = DEFFILE..PASSW.
```

This example illustrates the various combinations permitted for the QUERY command. The subschema name is INSTSUB; it was created under the account ACCT1234 with system password SESAME, and resides on private volume PK27. The database password is PW\$\$INST. The database area INSTAREA is in account G957153 with no system password; it resides on private volume 5472; the enciphering key is BETA. The database area STUAREA is in the same account as the user's log-on account; the system password is SOCRATES; it resides on two private volumes, STU1 and STU2; and the enciphering key is IOTA. The define file name is DEFFILE and it has a password PASSW.

**DISPLAY Command** The DISPLAY command provides the facility for defining the contents of a report. It specifies the report fields and their selection criteria. It must be preceded by a QUERY command. A period terminates the DISPLAY command signaling the end of the request and initiates the retrieval and report cycle. Optionally, the command may specify the retrieval path to effect a more efficient search of the database.

Format:

```

DISPLAY [ item-name-1 [ { OF } group-name-1 ]
        (arithmetic expression)
        { TOTAL } [ item-name-2 [ { OF } group-name-2 ] ]
        { MAX }
        { MIN }
        { AVG } (arithmetic expression) ] [ , column-heading-n ... [ , SIZE = integer ] ... ]
        [ , WHEN logical expression ] [ , VIA set-name-1 [ , set-name-2 ] ... ]
        [ , SERIAL ON group-name-3 [ (first-page, last-page) ]
        [ , DIRECT ON group-name-4 (page-number, line-number) ]
        [ , PORT ON group-name-5 ] ]

```

where

item-name-1 is the name of the database item to be displayed.

group-name-1 is a qualifier for the preceding item-name. It is required only when there is more than one item in the database with the same name.

item-name-2 is the name of the database item whose total value, maximum value, minimum value or whose average is to be displayed at the end of the report.

group-name-2 is a qualifier for the preceding item-name. It is required only when there is more than one item in the database with the same name.

arithmetic-expression is the addition, subtraction, multiplication or division of any combination of numeric database item names and numeric literals. Addition is represented by +, subtraction by -, multiplication by \* and division by /.

column-heading-n may be used to specify up to three column headings for this item (i.e., column-heading-1, column-heading-2, column-heading-3). If omitted, the column heading is the database item-name. Each column heading must be bounded by single quote marks. Up to three column headings may appear for each item. The second or third column-heading will appear on the second or third heading line of the report respectively. There is no default column heading for an arithmetic expression, if omitted, blanks will appear over the arithmetic expression. If column heading is omitted from the TOTAL arithmetic expression clause, no identifier will be printed on the total message.

integer specifies the size of the display field. If less than the size of the data item, the item will be truncated on the right if it is alphanumeric, and truncated on the left if it is numeric. If any nonzero digits are truncated on a numeric item, the report field will contain all asterisks. If integer is greater than the size of the data item, the item will be blank-filled on the right. If size is not specified for an arithmetic expression, the default size will be 16 characters.

logical-expression dictates the criteria for selecting the report data.

set-name-1, set-name-2 are the sets to be used for retrieval of data from a database. Since multiple retrieval paths may exist in the database structure, the user has the option of specifying the set-names that would result in a more efficient search of the database, or that would yield a report in a desired sequence.

group-name-3 is the name of the group to be located by means of a serial search of the database. This clause instructs IDP to select the port group by using the DBM routine FINDS (find-serial-search). In certain situations, a serial search can be more efficient than retrieval by set structures.

first-page is the first page of the database to be searched for an occurrence of the port group.

last-page is the last page of the database to be searched for an occurrence of the port group. The default-page range will be that specified in the subschema. The entire database will be searched if none was specified in the subschema.

group-name-4 is the name of the port group to be located by REF-CODE. DBM will locate this group by a FINDD (find direct) using the REF-CODE of the group.

page-number is the page in which the DIRECT port group is located.

line-number is the line number of the DIRECT port group. IDP constructs the appropriate REF-CODE using page number and line number.

group-name-5 is the name of the group that IDP uses as its port.

Example:

Assume the following DISPLAY specification:

```
DISPLAY STUDENT, SCHOLARSHIP, AMOUNT,  
TOTAL AMOUNT, AVG AMOUNT,  
MIN STUDENT, MAX SCHOLARSHIP,  
WHEN AMOUNT GE 2500.00.
```

The report might appear as follows:

STUDENT	SCHOLARSHIP	AMOUNT
GILLESPIE CH	ALCOA	4000.00
BARNES RH	EXXON	3333.33
PEDERSON SA	AMERICAN TELEPHONE	3000.00
ANDERSON PA	GENERAL MOTORS	3000.00
ELDRIDGE RI	XEROX	3000.00
MANAHAN GE	DOW CHEMICAL	2500.00

TOTAL OF AMOUNT = 18833.33  
AVERAGE OF AMOUNT = 3138.88  
MINIMUM OF STUDENT = ANDERSON RA  
MAXIMUM OF SCHOLARSHIP = XEROX

Note that for alphanumeric items, MIN and MAX refer to the minimum and maximum values, respectively, according to the EBCDIC collating sequence. If "MIN AMOUNT" and "MAX AMOUNT" were specified, the following summary totals would have been displayed:

```
MINIMUM OF AMOUNT = 2500.00  
MAXIMUM OF AMOUNT = 4000.00
```

## Report Items

Each item-name specified in the DISPLAY command will appear as a separate column in the report. Item-name identifies the name of the database item to be displayed. An item-name that is not unique must be qualified by the group-name to which the item belongs. Qualification takes the following format:

item-name  $\left\{ \begin{array}{l} \text{OF} \\ \text{IN} \end{array} \right\}$  group-name

### Examples:

PART-NO OF STK-ON-HAND

INSTRUCTOR IN SUMMER-SESSION

The title for each column will be the item-name itself, unless a column-heading is specified. In the above example, the column-heading SUMMER INSTRUCTOR might be preferred to the default title INSTRUCTOR.

The size of the print field is either that of the database item or the number of characters in the title, whichever is greater. Note that there are two spaces between each item.

The default database item size is the number of bytes of a nonnumeric item or the picture size of a numeric item other than binary or floating-point. The default size for a binary item is 10 digits. The SIZE = parameter overrides either of these default sizes. The size of a floating-point short item is fixed at 14 characters and that of a floating-point long item at 22 characters.

The default title size is the number of characters in the database item-name. If another title is specified, the title size changes to the number of characters in the new title specification.

## When Clause

The WHEN clause specifies the criteria for selecting the report items. The criteria are in the form of a logical expression, consisting of one or more data selection relations. Each relation is composed of

1. A database item-name (may be qualified with a group-name).
2. A relational operator.
3. A literal, a database item-name (may be qualified with a group-name), a constraint list, a range of constraint values, or an arithmetic expression.

The allowable relational operators are shown below:

<u>Operator</u>	<u>Meaning</u>	<u>Operator</u>	<u>Meaning</u>
EQ or =	equal to	LT or <	less than
NE or ~ =	not equal to	GE or >=	greater than or equal to
GT or >	greater than	LE or <=	less than or equal to

Literal. A literal is either numeric or nonnumeric. The type of literal used in the relation is determined by the type of the database items which is the subject of the comparison. A numeric item must be compared to a numeric literal, and a nonnumeric item to a nonnumeric literal. A numeric literal is a string of characters selected from the digits 0 through 9 (to a maximum of 31 digits), and optionally the plus or minus sign and a decimal point. A numeric literal may also be described as a floating-point constant in 'E' or 'D' notation for single and double precision, respectively.

A nonnumeric literal is a string of up to 255 EBCDIC characters bounded by single quote marks.



Examples:

GRANT-AMT GE 500.00

INSTRUCTOR-NO = 9636

LASTNAME OF STUDENT EQ 'POTTER'

Database Item-name. A database item-name used as the right term of a data selection relation is subject to the same rule as a literal in that an alphabetic or alphanumeric database item cannot be compared to a numeric database item.

Example:

WHEN GRADE LT CLASS-AVG

Constraint List. The constraint list defines the data values to be searched for. It consists of a list of right-term elements separated by commas. The elements may be literals, database item-names or intermixed literals and database item names. A 'true' condition exists if at least one element of the list satisfies the relationship, except when the relational operator is 'NE'.

For example:

WHEN COURSE = 'FRENCH', 'SPANISH'

would select students who are taking either French or Spanish or both.

When the relational operator is 'NE', all the right-term elements must satisfy the relation for the condition to be 'true'.

For example:

WHEN COURSE NE 'FRENCH', 'SPANISH', 'RUSSIAN'

would select students who are not taking any of the cited languages.

Range. The range of constraint values is indicated by the semicolon. Either end-point can be included or excluded by the appropriate relational operator. For instance, to list students with scholarships of at least 1000 dollars and less than 3000, one would say.

WHEN SCHOLARSHIP-AMT GE 1000.00; < 3000

Note that 'GE' and 'LE' include the end points and 'GT', 'LT' exclude the end points of a range.

The semicolon also allows for selecting on values greater than or less than a cutoff value with specific exclusions.

For example:

WHEN STUDENT-NO > 10; NE 100, 200, 300

would select students whose student numbers are greater than 10 but would exclude students numbered 100, 200, and 300.

The range of constraint values may also be expressed by the words THRU or THROUGH. This is equivalent to a less than or equal condition indicated by a semicolon

For example:

```
WHEN STUDENT-NO > 10 THRU 100
```

is equivalent to

```
WHEN STUDENT-NO > 10; LE 100
```

Arithmetic Expression. The arithmetic expression permits the user to construct a constraint value by adding, subtracting, multiplying and/or dividing any combination of numeric database items and numeric literals. Although allowed, the use of numeric literals only would serve no purpose, since the result could be expressed as a single numeric literal. The expression evaluation is conventional – clearing of parentheses inner to outer, then proceeding from left to right, with multiplication and division taking precedence over adjacent addition or subtraction.

Examples:

```
WHEN PAY-INC GE SALARY * 0.06
```

would select employees with salary increases of six percent or more (assuming that SALARY refers to prior rate of pay).

```
WHEN STUDENT-NO = 2 * (((2 * 7) + 4) / 3) + 5
```

is equivalent to:

```
WHEN STUDENT-NO = 17
```

This example is used to show the evaluation procedure, and is not recommended usage.

If STUDENT-NO was a CALC key or an inverted item, IDP would optimize the retrieval with a keyvalue of 17. However, no optimization would take place with  $2 * (((2 * 7) + 4) / 3) + 5$ .

The selection relations may be connected by the logical operators AND and OR. The AND operator connects relations that are included in a report if, and only if, each is satisfied (true). The OR operator connects those of which only one need be satisfied for the item to be included. In the evaluation of a logical expression, AND has precedence over OR. For example, the expression

```
A AND B OR C
```

is evaluated as (A AND B) OR C. If both A and B are true, the request is satisfied. If either A or B is not true, but C is true, the request is also satisfied. Similarly, the expression A OR B AND C is evaluated as A OR (B AND C).

Examples:

```
WHEN SCHOLARSHIP-AMT GE 2500.00.
```

```
WHEN INSTRUCTOR-NO = 9636 AND LASTNAME OF STUDENT EQ 'POTTER'.
```

### Via Clause

The VIA clause specifies the set-names to be used for retrieval of data from the database whenever multiple retrieval paths exist. The user who is knowledgeable of the database structure and its contents can use this clause to effect a more efficient search. Also, it enables him to specify a retrieval path that would generate the report in a desired sequence. See Example 3 below.

The VIA clause must specify sets lying on a direct path from "port" group (the entry group or first group accessed) to a "target" group (a group containing a display item).

### Serial Clause

The SERIAL clause specifies the port group that is to be located by means of a serial search of all or part of the database. This clause can be used in those situations where a serial search is more efficient than retrieval by set structures. See Example 4 below. The portion of the database to be searched may be limited to the range of pages specified in the clause. If the user does not specify a page range, the page range in the subschema, if present, will be used to limit the serial search.

### Direct Clause

The DIRECT clause specifies that the port group is a single occurrence of a group that is to be located by means of a DBM FINDD (Find direct) call using the REF-CODE containing the page number and line number specified in the clause.

The examples below illustrate a variety of DISPLAY commands.

#### Example 1:

```
DISPLAY STUDENT, SPONSOR, SCHOLARSHIP-AMT,  
      WHEN SCHOLARSHIP-AMT GE 2500.00.
```

This query generates a three-column report of all the students who have scholarships of \$2,500.00 or more. The report might appear as follows:

STUDENT		SPONSOR	SCHOLARSHIP-AMT
ANDERSON	RA	GENERAL MOTORS	3000.00
PEDERSON	SA	AMERICAN TELEPHONE	3000.00
ELDRIDGE	RI	XEROX	3000.00
MANAHAN	GE	DOW CHEMICAL	2500.00
GILLESPIE	CH	ALCOA	4000.00
BARNES	RH	EXXON	3333.33

This example is intended to illustrate the contents of the report, not the column spacing which will vary depending on the sizes of the respective items in the database.

#### Example 2:

```
DISPLAY STUDENT, SIZE=10  
      SPONSOR, 'SCHOLARSHIP', SIZE=12  
      SCHOLARSHIP-AMT 'AMOUNT'  
      WHEN SCHOLARSHIP-AMT GE 2500.00.
```

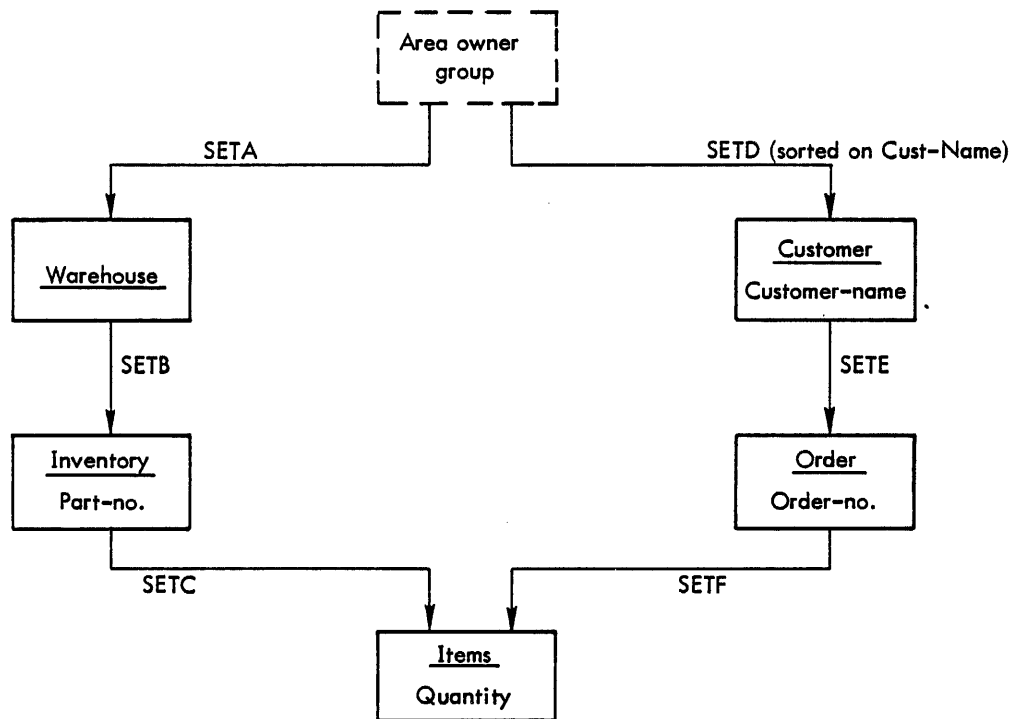
The example illustrates the use of the "column-heading" and SIZE options to produce a report with different column headings and different column sizes. This report, which is more compact, might appear as follows:

STUDENT	SCHOLARSHIP	AMOUNT
ANDERSON	GENERAL MOTO	3000.00
PEDERSON	AMERICAN TEL	3000.00
ELDRIDGE	XEROX	3000.00
MANAHAN	DOW CHEMICAL	2500.00
GILLESPIE	ALCOA	4000.00
BARNES	EXXON	3333.33

In conjunction with the DISPLAY command, the SORT command may be used to rearrange the order of the report lines. For example, it may be desirable to sort the report, in either ascending or descending sequence, by student name, or by scholarship sponsor, or by scholarship amount. The COUNT command could be used to display a count of the number of report lines generated, that is, the number of students satisfying the selection criteria (SCHOLARSHIP-AMT GE 2500.00). The TOTAL command would be used to accumulate and display the sum of the AMOUNT column, i.e., the total value of all scholarships greater than or equal to \$2500.00. Capabilities of the SORT, COUNT, and TOTAL commands are described later in this chapter.

Example 3 (VIA clause):

Consider a manufacturing application with the following database structure



DISPLAY ORDER-NO, QTY, PART-NO, CUST-NAME

VIA SETD, SETE, SETF.

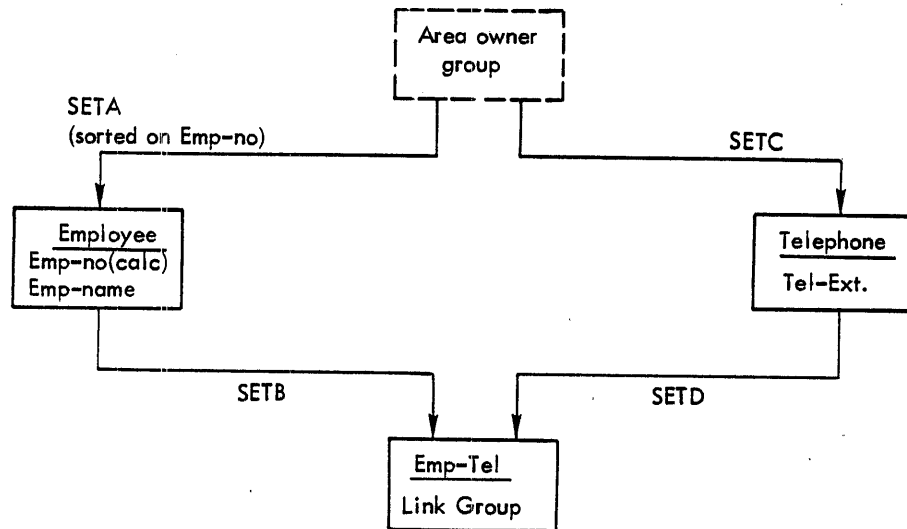
This query produces a report listing all the orders carried in the database. The report might appear as follows:

ORDER-NO	QTY	PART-NO	CUST-NAME
K2387	24	35M215	ACME FABRICATORS
K2402	100	Y68377	ACME FABRICATORS
K2402	60	B22416	ACME FABRICATORS
K2390	36	C17225	BAY HARDWARE
K2310	72	18K357	BAY HARDWARE
⋮	⋮	⋮	⋮

Assume that there are 10,000 occurrences of the INVENTORY group and 100 occurrences of the CUSTOMER group. Having such knowledge of the database, the user knows the retrieval process can be substantially shortened by starting with the CUSTOMER group. He was, therefore, able to specify the more efficient retrieval path using the VIA clause. Also, the user wanted a report that listed all orders grouped by customer. With the alternate retrieval path (starting with the INVENTORY group), the report would probably not be in this desired sequence unless a time-consuming sort was performed.

Example 4 (SERIAL clause):

Consider a general personnel application with the following database structure:



DISPLAY EMP-NO, EMP-NAME, TEL-EXT  
 SERIAL ON EMPLOYEE  
 VIA SETB.

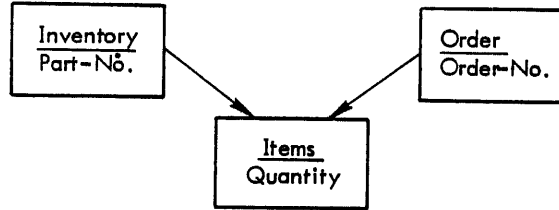
This query produces a report listing all the employees and their telephone extensions. Assume that the database contains 10,000 employee group entries distributed across a 1,000-page database. Having this knowledge, the user knows that a serial search of the database would be more efficient than retrieval by set structures, since a serial search would involve 1,000 I/O accesses to find the EMPLOYEE records, while retrieval by set structure may involve up to 10,000 I/O accesses. If the sequence of this report is desired to be alphabetic by EMP-NAME, the report would then have to be sorted (see SORT command).

Assume that the report is desired in EMP-NO sequence. The user would specify VIA SETA SETB since the EMPLOYEE group entries in SETA are sorted by EMP-NO. This would eliminate the sorting requirement. If the report is desired in TEL-EXT sequence, the user would specify VIA SETC SETD with no SERIAL clause, assuming that the TELEPHONE group entries in SETC are sorted by TEL-EXT.

### Port Clause

The PORT clause is used to specify the group with which the retrieval will start (the port group). It is used when constraints provide for direct retrieval of more than one group. It may also be used in lieu of the SERIAL clause. If the PORT clause is specified, then the SERIAL or DIRECT clause may not be specified.

### Example



Suppose in the above database, that INVENTORY and ORDER were CALC groups and that PART-NO and ORDER-NO were the respective CALC keys.

Consider the following IDP QUERY:

```
QUERY INVFILE.
```

```
DISPLAY QUANTITY WHEN PART-NO = 'H516' AND ORDER-NO = '35M215'.
```

In this case IDP will optimize on one of the CALC keys. If PORTFLAG is OFF (see the PORTFLAG command) IDP will arbitrarily pick either INVENTORY or ORDER as the port group. If PORTFLAG is ON, IDP will not retrieve anything, and will allow the user to respecify the query.

The user may specify.

```
QUERY INVFILE.
```

```
DISPLAY QUANTITY WHEN PART-NO = 'H516' AND ORDER-NO = '35M215' PORT ON INVENTORY.
```

In this case IDP will use INVENTORY as the port group.

**SORT Command** The SORT command allows the user to have the report sorted on one or more report item sequences. SORT may be either ascending (ASC/item-name), or descending (DES/item-name). If neither ASC nor DES is specified for a field, ASC is assumed. In the SORT command, the major control field is specified first, followed by successive minor control fields. The QUERY command must precede the SORT command. A maximum of 16 items may be specified as SORT keys.

### Format:

$$\text{SORT} \left\{ \begin{array}{l} \text{ASC} \\ \text{DES} \end{array} \right\} \text{item-name} \left\{ \begin{array}{l} \text{OF} \\ \text{IN} \end{array} \right\} \text{group-name} \} \dots$$

where

item-name is the name of the report item to be sorted.

group-name is a qualifier for the preceding item-name. It is required only when the database contains more than one item with the same name.

Example 1:

Using Example 1 of the DISPLAY command, assume that the report is desired in alphabetic student name sequence. The user would specify

```
    SORT STUDENT.  
    DISPLAY STUDENT, SPONSOR, SCHOLARSHIP-AMT  
    WHEN SCHOLARSHIP-AMT GE 2500.00.
```

The report then would appear as follows:

STUDENT		SPONSOR	SCHOLARSHIP-AMT
ANDERSON	RA	GENERAL MOTORS	3000.00
BARNES	RH	EXXON	3333.33
ELDRIDGE	RI	XEROX	3000.00
GILLESPIE	CH	ALCOA	4000.00
MANAHAN	GE	DOW CHEMICAL	2500.00
PEDERSON	SA	AMERICAN TELEPHONE	3000.00

The original report was not shown in this sequence because, in all likelihood, the student group entries would be stored in the database in some student number sequence, rather than by student name.

Example 2:

Using the same example, assume that the report is desired in scholarship amount sequence, in descending order, then by sponsor in ascending order. The user would specify the following SORT command:

```
    SORT DES/SCHOLARSHIP-AMT, ASC/SPONSOR.
```

This report would appear as follows:

STUDENT		SPONSOR	SCHOLARSHIP-AMT
GILLESPIE	CH	ALCOA	4000.00
BARNES	RH	EXXON	3333.33
PEDERSON	SA	AMERICAN TELEPHONE	3000.00
ANDERSON	RA	GENERAL MOTORS	3000.00
ELDRIDGE	RI	XEROX	3000.00
MANAHAN	GE	DOW CHEMICAL	2500.00

**COUNT Command** The COUNT command specifies that the number of report lines is to be counted. The count will be displayed at the end of the report.

Format:

COUNT.

In example 1 below, for the TOTAL command, illustrates the use of the COUNT command.

**TOTAL Command** The TOTAL command names the report items whose column contents are to be summed and, optionally, the items which are report break items. The totals will be displayed at the end of the report, and whenever the value of another item (the report break item) changes. The report breaks are arranged in a hierarchical order so that a change in a higher level report break will trigger totals for all lower level report breaks. The level

of the report break will be, as in the SORT command, implied by the order that the report break item appears in the TOTAL command. The first report break is the highest, the second is the next highest and the last item named is the lowest level.

Items totaled using the subtotal feature will automatically give final totals at the end of the report. If no report break items are given, only final totals will be displayed.

In order to get meaningful reports, the user should sort the report on the items named as report break items.

If at the end of the DISPLAY command, the user has not entered a SORT command or has entered a SORT command not specifying report break items or if the SORT command items are of a different level than the report break items, IDP will warn the user with the following message:

```
***56*** BREAK CTL ITEMS ARE NOT SORTED; DO YOU WANT TO SORT (Y/N)?
```

A Y response causes the report to be sorted (ascending) on the report break items. If a SORT command had been entered previously it will be ignored. After an N response IDP continues processing either without sorting or as previously specified in the SORT command.

Totals appear on a separate line. Those totals caused by a change in value of a report break item will be displayed with the message:

```
TOTAL OF column-heading-1 FOR column-heading-2 value 1 = value 2
```

where

column-heading-1 is the item being totaled.

column-heading-2 is the report break.

value-1 is the value of the report break which has just changed.

value-2 is the actual total.

The maximum number of characters for each of the following items, column-heading-1, column-heading-2 and value-1 is 32. If necessary these items are truncated on the right to 32 characters.

Final totals appear at the end of the report with the message:

```
TOTAL OF column-heading = value
```

The following restrictions apply to the TOTAL command.

1. Alphanumeric or alphabetic items may not be totaled.
2. The QUERY command must precede the TOTAL command.
3. A maximum of 16 items may be totaled.
4. A maximum of 16 items may be report breaks.



Format:

$$\underline{\text{TOTAL}} \left\{ \text{item-name-1} \left[ \left\{ \frac{\text{OF}}{\text{IN}} \right\} \text{group-name-1} \right] \right\} \left[ , \text{item-name-2} \left[ \left\{ \frac{\text{OF}}{\text{IN}} \right\} \text{group-name-2} \right] \right] \dots$$
$$\left[ \underline{\text{ON}} \text{item-name-3} \left[ \left\{ \frac{\text{OF}}{\text{IN}} \right\} \text{group-name-3} \right] \right] \left[ , \text{item-name-4} \left[ \left\{ \frac{\text{OF}}{\text{IN}} \right\} \text{group-name-4} \right] \right] \dots$$

where

item-name-1, item-name-2 are the item(s) to be totaled.

group-name-1, group-name-2 are group-name qualifiers for item-name-1 and item-name-2, respectively.  
Required only when the item names are not unique.

item-name-3, item-name-4 are report break items that control the subtotals. Item-name-3 is a higher level report break than item-name-4. A change in the value of item-name-3 implies a control break for item-name-4.

group-name-3, group-name-4 are group-name qualifiers for item-name-3 and item-name-4, respectively.  
Required only when the item names are not unique.

Example:

This example illustrates both the COUNT and TOTAL commands. Again, using Example 1 of the DISPLAY command, assume that it is desired to have a count of the number of report lines, and a total of the SCHOLARSHIP-AMT column.

The user would specify

```
COUNT. (REF)
TOTAL SCHOLARSHIP-AMT. (REF)
DISPLAY STUDENT, SPONSOR, SCHOLARSHIP-AMT (REF)
WHEN SCHOLARSHIP-AMT GE 2500.00. (REF)
```

The report would appear as follows:

STUDENT	SPONSOR	SCHOLARSHIP-AMT
ANDERSON RA	GENERAL MOTORS	3000.00
PEDERSON SA	AMERICAN TELEPHONE	3000.00
ELDRIDGE RI	XEROX	3000.00
MANAHAN GE	DOW CHEMICAL	2500.00
GILLESPIE CH	ALCOA	4000.00
BARNES RH	EXXON	3333.33
TOTAL OF SCHOLARSHIP-AMT = 18833.33		
LINE COUNT FOR THIS REPORT = 6		

This report shows that six students have scholarships of \$2500.00 or more, and that the total amount of these scholarships is \$18,833.33.

Example 2:

    SORT DEPT.  
    TOTAL BUDGET ON DEPT.  
    DISPLAY PROJ-NO, DEPT, BUDGET.

In this report, a subtotal of BUDGET will be displayed every time DEPT changes, and a grand total will be displayed or the end of the report. The report would appear as follows:

PROJ-NO	DEPT	BUDGET
K1311700	3500	6800.00
K1311600	3500	5500.00
K1311500	3500	9000.00
TOTAL OF BUDGET FOR DEPT 3500 = 21300.00		
PROJ-NO	DEPT	BUDGET
K1322200	3800	900.00
K1313500	3800	7200.00
K1315400	3800	4500.00
TOTAL OF BUDGET FOR DEPT 3800 = 12600.00		
TOTAL OF BUDGET = 33900.00		

Example 3:

    SORT DEPT, SECTION.  
    TOTAL COMMISSION, SALES-AMT ON DEPT, SECTION.  
    DISPLAY SALESMAN, COMMISSION, SALES-AMT DEPT SECTION  
        WHEN ORDER-DATE GE '740101'.

In this report, both COMMISSION and SALES-AMT are to be totaled. Whenever the value of SECTION changes, the subtotals of COMMISSION and SALES-AMT will be displayed for that SECTION. Whenever the value of DEPT changes, the subtotals of COMMISSION and SALES-AMT will be displayed for that DEPT and for that SECTION. In addition, the grand totals of COMMISSION and SALES-AMT will be displayed at the end of the report. The report would appear as follows:

SALESMAN	COMMISSION	SALES-AMT	DEPT	SECTION
ANDERSON	115.34	1007.35	66K5	B17
PEDERSON	76.25	807.36	66K5	B17
ANDERSON	91.32	900.35	66K5	B17
TOTAL OF COMMISSION FOR SECTION B17 = 282.91				
TOTAL OF SALES-AMT FOR SECTION B17 = 2715.06				
SALESMAN	COMMISSION	SALES-AMT	DEPT	SECTION
WILKES	100.25	935.00	66K5	G17
AVOGADRO	217.00	1786.55	66K5	G17
ANTHONY	66.62	773.32	66K5	G17
TOTAL OF COMMISSION FOR SECTION G17 = 383.87				
TOTAL OF SALES-AMT FOR SECTION G17 = 3494.87				
TOTAL OF COMMISSION FOR DEPT 66K5 = 666.78				
TOTAL OF SALES-AMT FOR DEPT 66K5 = 6209.93				
TOTAL OF COMMISSION = 666.78				
TOTAL OF SALES-AMT = 6209.93				

**LIMIT Command** The LIMIT command provides the facility for limiting the production of a report with respect to the number of report lines created or CPU time expended.

Format:

LIMIT [ LINES = integer ][ TIME = value ].

where

integer is the maximum number of lines.

value is the maximum amount of CPU time in minutes. The value may be expressed as a number with two decimal places (e.g., 1.25).

One or both options may be specified. When one of the limits is exceeded, one of the following messages will be displayed:

\*\*\*51\*\*\* TIME LIMIT EXCEEDED; QUIT, CONTINUE OR RESTART (Q/C/R)?

\*\*\*52\*\*\* LINES LIMIT EXCEEDED; QUIT, CONTINUE OR RESTART (Q/C/R)?

Thus, the user is notified that one of the limits has been exceeded, and he is given the choice of continuing execution, aborting the job, or entering another query. If he chooses to continue execution, the limits are reinstated and he will be notified again each time the limits are exceeded.

Example:

LIMIT LINES = 100, TIME = .30.

This command informs IDP that the user is to be notified whenever the number of report lines has exceeded 100, or CPU time has exceeded 0.30 minutes, whichever occurs first.

**OUTPUT Command** The OUTPUT command provides the facility for outputting the report to the line printer or to a file and specifying the dimensions of the report page.

Format:

OUTPUT [ ON ] [ TO ] { ME  
LP  
file-id  
NON-REPORT file-id-2 } [ WIDTH = integer-1 ] [ LENGTH = integer-2 ]

where

ME is user terminal (need not be specified since it is default).

LP is line printer.

file-id-1 is name of file on which report is to be saved.

integer-1 is the maximum width (number of print spaces) of the report.

integer-2 is the page length (number of lines) of the report.

file-id-2 is the name of the file on which the raw output is to be saved. File-id-2 is limited to seven characters. In addition, file-id-2 does not contain titles or column headings. No data conversion will be performed for file-id-2.

If OUTPUT is to a NON-REPORT file, at the end of the query the following will be displayed

ITEM-NAME	START	SIZE	MODE
-----------	-------	------	------

A list of the file and its attributes follows.

NUMBER OF RECORDS =

RECORD SIZE (BYTES) =

### Example

Suppose a user wants to extract certain items from a database to be used as input to another program. A NON-REPORT file may be created by IDP as follows.

```
IIDP
IDP VERSION B00
:OUTPUT NON-REPORT NEWFILE.
:QUERY HOSPSUB.
:DISPLAY PATNUM PATNAME AGE SOSECNO.
          ITEM NAME   START   SIZE   MODE
PATNUM                0       4   BINARY
PATNAME               4      24  ALPHAN
AGE                   28       3  NUMERIC
SOSECNO               31       7  NUMERIC
```

```
NUMBER OF RECORDS = 35
RECORD SIZE (BYTES) = 40
```

A file NEWFILE exist with the above attributes.

If OUTPUT is to a NON-REPORT file, any SORT, COUNT, TOTAL, and TITLE commands are ignored.

### Report Width

If the WIDTH option is specified, and the width of the report exceeds the value of integer-1, or if the width option is not specified and the width of the report exceeds the default width, the following diagnostic will be issued

```
***54*** WIDTH OF REPORT EXCEEDS LIMIT; QUIT, CONTINUE, OR RESTART (Q/C/R)?
```

If the user responds with a C (Continue), the user will be prompted for additional search criteria, however those items which exceed the maximum width will not be displayed. If the user responds with a Q (Quit) the user will be returned to TEL. If the user responds with an R (Restart), the user may enter another report.

The PLATEN command (see Chapter 2) can be used at log-on time to accomplish automatic line folding on the terminal. The PLATEN command enables the user to specify the maximum number of characters to be written per line. Note that this is the width of the line, and not the width of the report.

### Example:

```
IPLATEN 72
IIDP
IDP VERSION B00
:OUTPUT ME WIDTH = 122.
```

In this example the width of the report is 122 characters, the first 72 characters will be printed on the first line and the remaining 50 characters will be printed on the next line. If the platen setting is not 0, IDP will ensure that an item or header is not split on two lines. If an item will not completely fit within the line, the remainder of the line will be blank filled and that item will start on the next line.

If OUTPUT is to ME or LP and WIDTH is greater than 140 characters, those characters in excess of 140 will be lost. Vertical format control (VFC) characters will be included in the report lines.

Default Width

Table 1 describes the default widths.

Table 1. Default Widths

Batch	On-Line			
	LP	file-id	ME	
			Teletype MOD 33, 35, 37 Xerox 7015	IBM 2741
127	127	127	72	127

The width (w) of the PLATEN command of the Terminal Executive Language (TEL) is not the same as the maximum width of the report specified in the width clause of the OUTPUT command of IDP.

The PLATEN command of TEL can affect the appearance of the report. Below are three examples, each with an identical IDP specification but each with a different PLATEN setting. HOSPNAME is a 64 character alphanumeric item, PATNUM is a binary item and needs 10 print characters, SOSECNO is a 9 character item, AGE is a 3 character item, and PATNAME is a 24 character item. In addition, the items are separated from each other by two spaces.

```
!PLATEN 72

!IDP.
  IDP VERSION 800
:QUERY HOSPSUB.
:OUTPUT WIDTH = 140.
:DISPLAY HOSPNAME 'HOSPITAL NAME'
:PATNUM SOSECNO AGE
:PATNAME 'PATIENT NAME'
:WHEN PATNUM = 1.

      HOSPITAL NAME
PATNUM  SOSECNO  AGE      PATIENT NAME
ANYTOWN HOSPITAL
      1 550302132  35  MARCUS TOM J
:END
```

```
!PLATEN 80

!IDP.
  IDP VERSION 800
:QUERY HOSPSUB.
:OUTPUT WIDTH = 140.
:DISPLAY HOSPNAME 'HOSPITAL NAME'
:PATNUM SOSECNO AGE
:PATNAME 'PATIENT NAME'
:WHEN PATNUM = 1.

      HOSPITAL NAME          PATNUM
SOSECNO  AGE      PATIENT NAME
ANYTOWN HOSPITAL
550302132 35  MARCUS TOM J
:END
```

```

!PLATEN 0

!IDP
  IDP VERSION 800
:QUERY HOSPSUB.
:OUTPUT WIDTH = 140.
:DISPLAY HOSPNAME 'HOSPITAL NAME'
:PATNUM SOSECNO AGE
:PATNAME 'PATIENT NAME'
:WHEN PATNUM = 1.

      HOSPITAL NAME          PATNUM  S
      AGE      PATIENT NAME

ANYTOWN HOSPITAL          1  5
32  35  MARCUS TOM J

:END
!

```

In the above example, item SOSECNO is printed on two separate lines and characters are lost ('OSECNO' and '503021') because the terminal initiated a carriage return when it reached the right margin. The number of characters lost (while traversing to left margin) is terminal dependent.

#### Report Truncation

If the items in the above example are rearranged so that item HOSPNAME does not fit on the first report line, then, in order that HOSPNAME not be split and printed on two lines IDP moves this item to line two. In this case there is not enough room to print items PATNUM and PATNAME. This is indicated by the message REPORT TRUNCATION. To avoid this, the user may 1) rearrange the items as illustrated in the previous example, 2) change the size of some of the items, 3) change the PLATEN setting or 4) increase the WIDTH of the report.

#### Example of REPORT TRUNCATION

```

!PLATEN 72

!IDP.
  IDP VERSION 800

:QUERY HOSPSUB.
:OUTPUT WIDTH = 140.
:DISPLAY SOSECNO AGE HOSPNAME 'HOSPITAL NAME' PATNUM PATNAME
:' PATIENT NAME
:WHEN PATNUM = 1.

***163*** REPORT TRUNCATION
***163*** REPORT TRUNCATION
SOSECNO  AGE          HOSPITAL NAME

550302132  35
ANYTOWN HOSPITAL

:END

```

Example: Avoiding REPORT TRUNCATION by decreasing SIZE of items.

```
!PLATEN 72

!IDP.
  IDP VERSION B00
:QUERY HOSPSUB.
:OUTPUT WIDTH = 140.
:DISPLAY SOSECNO AGE
:HOSPNAME 'HOSPITAL NAME' SIZE = 20
:PATNUM 'PAT' 'NUM' SIZE = 3
:PATNAME 'PATIENT' 'NAME' SIZE = 12
:WHEN PATNUM = 1.

SOSECNO  AGE      HOSPITAL NAME      PAT      PATIENT
          AGE      HOSPITAL NAME      NUM      NAME
550302132 35  ANYTOWN HOSPITAL      1  MARCUS TOM J

:END
```

Example: Avoiding REPORT TRUNCATION by changing the PLATEN.

```
!PLATEN 0

!IDP.
  IDP VERSION B00

:QUERY HOSPSUB.
:OUTPUT WIDTH = 140.
:DISPLAY SOSECNO AGE
:HOSPNAME 'HOSPITAL NAME'
:PATNUM PATNAME 'PATIENT NAME'
:WHEN PATNUM = 1.

SOSECNO  AGE      HOSPITAL NAME      PATNUM      PATIENT NAME
550302132 35  ANYTOWN HOSPITAL      1  MARCUS TOM J

:END
```

### Report Length

If the user has entered a TITLE command (described subsequently), the title will always be displayed at the beginning of each report. This is followed by a line feed. The column headings will be displayed followed by another line feed. This is followed by the report lines. If column headings are to be displayed at the top of each page, the LENGTH option can be used for this purpose. Whenever the number of lines displayed exceeds the value of integer-2, the column headings will be displayed again.

If output is to LP, the default length will be taken from the monitor. If output is to ME or file-id continuous format is assumed. Note that length refers to the total number of lines printed, this includes title lines, headers and the spaces in between. It is not equal to the line output specified in the LIMIT command or the count in the COUNT command.

Example 1:

OUTPUT WIDTH = 72.

As a result of this command the report is displayed on the user's terminal (by default), and the report width is 72 characters, as specified.

Example 2:

OUTPUT LP, WIDTH = 132, LENGTH = 36.

This command specifies that the report is to be output to the line printer, that report width is 132 characters, and that page length is 36 lines. Since the LENGTH option is specified, column headings will be output at the top of each page.

**REPORT Command** The REPORT command provides the facility to produce a report from a cataloged file. This cataloged file is a source file that contains all the commands (QUERY, DISPLAY, etc.) necessary to define the reports. This capability is useful in situations where a specific report is desired on a regular basis. Having established a cataloged file, the user can produce the report by means of a single REPORT command; he need not re-define the report each time. REPORT commands may not be nested, i.e., the source file may not contain a REPORT command. Figure 4 illustrates creation of a source file using EDIT.

```
!BUILD ACTIVE-ORDERS (M)
  1.000 QUERY ORDERSUB.(M)
  2.000 LIMIT TIME = 0.25.(M)
  3.000 OUTPUT WIDTH = 72.(M)
  4.000 DISPLAY ORDER-NO,QTY PART-NO, CUST-NAME (M)
  5.000 VIA SETD,SETE,SETF.(M)
  6.000 END.(M)
  7.000 (M)

!IDP

IDP VERSION B00

:REPORT ACTIVE-ORDERS.(M)

ORDER-NO   QTY   PART-NO   CUST-NAME
K2387      24   35M215   ACME FABRICATORS
K2402     100   Y68377   ACME FABRICATORS
K2402      60   B22416   ACME FABRICATORS
K2390      36   C17225   BAY HARDWARE
K2310      72   18K357   BAY HARDWARE
```

Figure 4. Creation of a Source File Using EDIT



Format:

REPORT source-file [. [account] [ file-password ]].

where

source-file is the source file containing the report definition.

account is the account under which the source file was created.

file-password is the password of the source file.

Example 1:

REPORT ACTIVE-ORDERS.

This command would produce a report as defined by a source file named ACTIVE-ORDERS (see Figure 4). The format and content of the report might appear as shown in the lower half of Figure 4 (also see Example 3 of the DISPLAY command).

Example 2:

REPORT ACTIVE-ORDERS .K1311301.

This is similar to Example 1, except that the source file was created under account K1311301, which is different from the user's log-on account.

Example 3:

REPORT ACTIVE-ORDERS.K1311301.APOLLO.

This is similar to Example 2, except for the system password APOLLO, used in creating the source file.

Example 4:

REPORT ACTIVE-ORDERS..APOLLO.

This is similar to Example 3, except that the account number is omitted because it is the same as the user's log-on account. Note that the period normally preceding the account number must be included if a system password is specified.

A REPORT file may have unspecified parameters which can be entered at execution time to produce a specific desired report. This allows a single prestored file to produce many reports. An unspecified parameter is indicated by the special character "?". Only one "?" may appear on each line, and it must be at the end of that line. The presence of a "?" implies the end of a report line. Only the period is allowed immediately following the "?". A period following the "?" implies a period will be placed at the end of the user's response.

There is no limit to the number of "?" in a prestored file.

Upon execution of the REPORT file, the presence of an "?" causes IDP to type that line and wait for an appropriate response. This response may be a single word, literal, operator, or a character string. The examples below illustrate the range of flexibility of this feature. Note that all characters typed by the system are shown underlined. Everything else shown is typed by the user.

Example 5:

Assume there is a prestored file called REP1 which contains the following report definition:

```
QUERY      SCHOOLSUB.
DISPLAY    STUDENT-NAME, COURSE, GRADE
          WHEN STUDENT-NO = ?.
END.
```

If the user wants to produce this report, the on-line session proceeds as follows:

```
_IDP (M)
:
:
:REPORT REP1.(M)
  WHEN STUDENT-NO = ? 44639 (M)
```

IDP types the line containing "?" (WHEN STUDENT-NO = ?) and waits for a response. After the user types the desired student number (44639) (M), IDP produces the report. This construct uses the prestored IDP query terms to remind the user which missing data is needed.

Example 6:

Assume there is another prestored file called REP2 which contains the following report definition.

```
QUERY SCHOOLSUB.
DISPLAY STUDENT-NAME, COURSE, GRADE WHEN STUDENT-NO =
?.
END.
```

In this case, the on-line session proceeds as follows:

```
_IDP (M)
:
:REPORT REP2.(M)
? 44639 (M)
```

Since there were no words preceding the "?", IDP types only a "?". This construct permits high-speed interaction where the user is presumed to know what missing data is needed.

Example 7:

The user response to "?" is not limited to a single word or literal. It may consist of any valid IDP phrase or character string. Assume the following prestored file called REP3:

```
QUERY      INSTSUB.ACCT1234.SESAME
          SN = PK37
          PASSWORD = ?
          AREA = INSTAREA
          CIPHKEY = 'BETA'.
          OUTPUT LENGTH = ?.
          SORT ?.
DISPLAY STUDENT-NO, STUDENT-NAME, COURSE, GRADE
          WHEN
          INSTRUCTOR-NO ?.
END.
```

An on-line session would proceed as follows:

```
␣IDP Ⓜ  
⋮  
:REPORT REP3. Ⓜ  
  PASSWORD = ? 'PW$$INST' Ⓜ  
  OUTPUT LENGTH = ? 38 Ⓜ  
  SORT ? COURSE Ⓜ  
  INSTRUCTOR-NO ? = 1255 Ⓜ
```

In this example, the user supplied the information "=1255" in response to "INSTRUCTOR-NO ?". If this report is desired for instructors 1255, 1810, and 1620, the user response would be:

```
INSTRUCTOR-NO ? = 1255, 1810, 1620 Ⓜ
```

If the report is desired for instructors whose numbers are between 1200 and 1500, inclusive, the user response would be:

```
INSTRUCTOR-NO ? GE 1200; LE 1500 Ⓜ
```

If the desired report is a list of all the students who failed in those courses taught by instructor 1255, the user response would be:

```
INSTRUCTOR-NO ? = 1255 AND GRADE = 'F' Ⓜ
```

The REPORT file with unspecified parameters combined with the user's response acts like any other query. The absence of a period, where required, or a parameter, where required, will cause errors.

#### Example 8:

Consider a prestored file called REP4 which contains the following report definition.

```
QUERY SCHOOLSUB.  
SORT ?  
DISPLAY STUDENT-NAME.
```

If the user interacted in the following manner, an error would result.

```
␣IDP Ⓜ  
⋮  
:REPORT REP4. Ⓜ  
  SORT ? Ⓜ
```

A null response is a valid response, but since the SORT command requires a period, and no period was found, IDP will read the next record in the REPORT file, "DISPLAY STUDENT-NAME". If the word DISPLAY is not an item-name in the database, an error will result since IDP interprets the query to be SORT DISPLAY. At this point the user should restart the query by hitting the break key on the terminal.

If on the other hand the user interacted with REP4 in the following manner, the following would result.

```
!IDP (11)
:
:REPORT REP4 (11)
SORT? STUDENT-NAME.(11)
```

where STIDENT-NAME is a misspelling of STUDENT-NAME, the following would occur;

```
STIDENT-NAME.
$
***9*** THIS ITEM-NAME NOT FOUND IN THE SUBSCHEMA
SORT ? STUDENT-NAME.(11)
```

This time STUDENT-NAME is spelled correctly and accepted; IDP continues reading the REPORT file.

In the first case of example 8, the user's response was accepted, but combined with the next line of the report, it became an unacceptable query. In the second case, the user's response was not accepted and the user was prompted again for the correct response.

Finally if REP4 were set up in the following manner the error resulting from the null response on the SORT command would not occur.

```
QUERY SCHOOLSUB.
SORT ?.
DISPLAY STUDENT-NAME.
```

Since the period following the "?" implies a period at the end of the user's response. Any response must be accepted before the next record of the REPORT file is to be read.

**TITLE Command**      The TITLE command allows the user to specify a title which will appear at the top of each page.

Format:

```
TITLE ['text-string']... .
```

where text-string is any set of characters enclosed in quotes. The maximum number of characters in text-string is, for the first title line, the page width minus 22; for all other lines, the page width.

Included with the TITLE command is an implicit request for the page number and current date to be placed on the same line as the first title line.

Title Spacing

The TITLE line will have the current date left justified and the page number right justified. The page number is a zero suppressed 10 digit field. The user's text-string is centered on the page using the value named in the WIDTH clause of the OUTPUT command as the total line size (see OUTPUT command, for default width). In order to accommodate the date and page number, the user text-string must be no longer than the WIDTH-22.

Example:

```
OUTPUT WIDTH = 50.
TITLE 'STUDENT GRADE REPORT'.
```

At the top of each page, the following title line will appear, spaced as indicated.

Print	}	0	1	2	3	4	5														
Position	}	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0
Numbers	}	mmm			dd,	yy	STUDENT GRADE REPORT										n				

where

mmm is the current month.

dd is the current day.

yy is the current year.

n page number.

**DEFINE Command** The DEFINE command provides the facility for predefining words to enable easier and faster query entries. A predefined word is a unique user constructed word which can be translated into a valid IDP phrase. The DEFINE capability may be used for abbreviating long database item-names, or for constructing a new IDP language to suit the user's specific environment.

The predefined words and their equivalent IDP phrases are stored and maintained on a permanent keyed file in the user's account. The DCB name is F:DFINE and the file name is IDPDEFINE. The DEFINE command enables creation or addition of predefined words into this file, while the DELETE command (described in the next section) enables deletion of words from the file.

In essence, a public DEFINE file may be created to serve as the central IDP library file to which all members of an installation would have access. For a typical installation, the Database Administrator would be responsible for maintaining this file in a specified account. As such, he would control all additions and deletions of predefined words to this file. He could utilize any of the security measures provided by CP-V, such as, passwords, write-protection, etc.

Format:

```
DEFINE [['_word' as 'IDP-phrase'] ...
      [FILE = define-file-name [.account][file-password]] [, SN = value] ] .
```

where

word a unique user-constructed word, up to a maximum of 31 characters. The word must be enclosed by single quote marks.

IDP-phrase any valid IDP phrase, up to a maximum of 255 characters. The IDP phrase must be enclosed by single quote marks.

define-file-name is the name of the user's define file. If no define-file-name is given, the define-file-name defaults to IDPDEFINE.

account is the account under which the define-file was created. Required if and only if different from the current user's logon account. If different from the user's logon account, the user must have write access in order to use the first form of the DEFINE command or the DELETE command.

file-password is the system password for the DEFINE file. Required if and only if a password exists.

value is the volume serial number of the disk pack that contains the define file.

These phrases must conform to the following rules:

1. The 'word' must be a unique word which does not conflict with data-names in the database or with IDP keywords. IDP will first check for a valid word or name, if none is found, then the define file will be checked. Thus, if the user defines;

DEFINE 'SORT' AS 'DISPLAY'. The SORT command will still be a SORT command and not a DISPLAY command.

2. Nesting of DEFINES is not allowed for 'IDP-phrase', i.e., the phrase may not contain other predefined words previously established.
3. The 'IDP-phrase' may consist of a single blank character. This defines the 'word' as a 'null' word, i.e., a word which has no meaning but is used merely for readability. For example, the word 'THE' may be defined as a 'null' word.
4. 'word' may not contain embedded blanks or special characters except '-'.  
5. 'word' may not be used in place of a file-id or part of a file-id (name.account.password (SN = nnnn). Equivalently, 'IDP-phrase' may not begin with a file-id although a file-id may be contained within the 'IDP-phrase'.

For example:

If SCHOOLSUB were the name of a subschema file and the following DEFINE statement were made:

```
DEFINE 'SS' AS 'SCHOOLSUB'.
```

The QUERY;

```
QUERY SS.
```

would result in an error

```
QUERY SS.  
  $  
***]*** SUBSCHEMA NAME IN ERROR
```

However, if SCHOOLSUB were not at the beginning of the phrase, the defined word would be allowed as in,

```
DEFINE 'GET' AS 'QUERY SCHOOLSUB.'.
```

The following IDP session would be allowed.

```
IIDP  
:  
:  
:  
:GET DISPLAY STUDENT-NAME.
```

6. 'word' may not be used in place of a literal. Equivalently, 'IDP-phrase' may not begin with a literal although a literal may be contained within 'IDP-phrase'.
7. 'word' may not be used in place of any valid special character. Equivalently, 'IDP-phrase' may not begin with a special character.
8. If the DEFINE file is in another account, the user must have write access to that account in order to use the DEFINE or DELETE command.

### Example 1:

This example illustrates the abbreviation of a long database item-name. Assume that a query session consists of the following commands:

```
IIDP
:
:
QUERY SCHOOLSUB.
SORT SCHOLARSHIP-AMT.
DISPLAY STUDENT, SPONSOR, SCHOLARSHIP-AMT,
      TOTAL SCHOLARSHIP-AMT,
      WHEN SCHOLARSHIP-AMT GT 1000.00.
```

By use of the DEFINE command, the item 'SCHOLARSHIP-AMT' may be redefined by the shorter name 'AMT' as follows:

```

:
:
DEFINE 'AMT' AS 'SCHOLARSHIP-AMT'.
QUERY SCHOOLSUB.
SORT AMT.
DISPLAY STUDENT, SPONSOR, AMT,
      TOTAL AMT,
      WHEN AMT GT 1000.00.
```

The above DEFINE command causes the name 'AMT', along with its equivalent 'SCHOLARSHIP-AMT', to be entered on a permanent keyed file in the user's account. Hereafter, any reference to 'AMT' is interpreted by IDP to refer to the database item 'SCHOLARSHIP-AMT'.

### Example 2:

This example illustrates a simplification of the IDP language. Since the IDP keywords are well known to IDP users, these keywords may be redefined by single letters purely for convenience.

```
IIDP
:
:
DEFINE 'Q' AS 'QUERY'
      'S' AS 'SORT'
      'D' AS 'DISPLAY',
      'T' AS 'TOTAL',
      'W' AS 'WHEN'.
Q SCHOOLSUB.
S AMT.
D STUDENT, SPONSOR, AMT,
  T AMT,
  W AMT GT 1000.00.
```

Note that IDP 'remembers' the previous definition of 'AMT' since the permanent keyed file containing the DEFINEd information is cumulative. Entries may be deleted, however, by using the DELETE command which is described in the next section.

### Example 3:

The DEFINE command may be used to redefine text string by the use of double quotes.

To enter the phrase QUERY SCHOOLSUB SN = 'AD01' PASSWORD = 'SAILBOAT'. in the define file the text strings 'AD01' and 'SAILBOAT' are placed in double quotes in the DEFINE command as follows.

```
DEFINE 'MYQUERY' AS 'QUERY SCHOOLSUB SN = "AD01" PASSWORD = "SAILBOAT".'. '
```

#### Example 4:

Neither is the DEFINE command limited to one-for-one replacement of IDP words. To continue from Example 2, assume the following definition had been given:

```
DEFINE 'SCHOLARSHIPS' AS 'STUDENT, SPONSOR,  
SCHOLARSHIP-AMT, TOTAL SCHOLARSHIP-AMT,  
WHEN SCHOLARSHIP-AMT GT 1000.00'.
```

The query session would have been simplified to the following:

```
IIDP  
:  
:  
Q SCHOOLSUB.  
S AMT.  
D SCHOLARSHIPS.
```

Note that two rules must be observed:

1. The predefined name 'SCHOLARSHIPS' must not conflict with a data-name in the database or with an IDP keyword.
2. There must be no nesting of DEFINES. In the definition of 'SCHOLARSHIPS', the previously defined 'AMT' may not be used as a replacement of 'SCHOLARSHIP-AMT'.

To illustrate the use of the DEFINE command to an extreme, assume that 'SCHOLARSHIPS' instead were defined as follows:

```
DEFINE 'SCHOLARSHIPS' AS 'QUERY SCHOOLSUB.  
SORT SCHOLARSHIP-AMT. DISPLAY STUDENT, SPONSOR,  
SCHOLARSHIP-AMT, TOTAL SCHOLARSHIP-AMT,  
WHEN SCHOLARSHIP-AMT GT 1000.00'.
```

The query session would have been further implied to the following:

```
IIDP  
:  
:  
SCHOLARSHIPS.
```

The desired report would be produced by the single command 'SCHOLARSHIPS'! The only restriction is that the IDP phrase may not exceed 255 characters.

A listing of the DEFINE file may be obtained by using the PCL processor as follows:

```
IPCL  
COPY IDPDEFINE TO ME(K)  
END
```

#### Define Error Detection

Since there may not be any files available at the time the DEFINE command is entered, 'IDP-phrase' cannot be checked for validity. Hence, an error in 'IDP-phrase' will not be detected until that phrase is used in an IDP session.

When an error in 'IDP-phrase' is detected the following will be issued.

1. Message 41  
\*\*\*41\*\*\* DEFINE PHRASE ERROR; DEFINED PHRASE/WORD/ERROR MSG
2. The defined phrase.
3. A flag (\$) indicating the error in the defined phrase.



4. The line that the user keyed in.
5. A flag (\$) indicating the defined word error.
6. The error message.

The user may continue the request by keying in the correct response for that particular error message. At the end of a command, the user may DELETE the word and then re-DEFINE the word again correctly.

Example 5:

Again assume there is a subschema named SCHOLSUB.

```
IIDP
:
:DEFINE 'FETCH' AS 'QUERY SCHOLSUB.'.
:FETCH DISPLAY STUDNAME.
***41*** DEFINE PHRASE ERROR; DEFINED PHRASE/WORD/ERROR MSG
QUERY SCHOLSUB.
$
FETCH DISPLAY STUDNAME.
$
***1*** SUBSCHEMA NAME IN ERROR
```

The user may continue the request by keying in the correct subschema name.

```
:SCHOLSUB.
```

Since this is the end of the QUERY command, the user may now correct the IDP-phrase as follows.

```
:DELETE 'FETCH'.
:DEFINE 'FETCH' AS 'QUERY SCHOLSUB.'.
```

**DELETE Command** The DELETE command provides for the deletion of predefined words from the DEFINE file. Since the DEFINE file is a permanent keyed file, all predefined words remain in the file unless specifically deleted. If an existing word has to be redefined, it must be deleted first (DELETE command), then redefined (DEFINE command). This procedure is necessary in order to prevent accidental destruction of predefined words.

Format:

```
DELETE {'word'} ...
```

where word is a predefined word existing in the DEFINE file. The word must be enclosed by single quote marks.

Example:

```
DELETE 'AMT', 'SCHOLARSHIPS'.
```

This deletes the definitions of 'AMT' and 'SCHOLARSHIPS' previously established in Examples 1 and 4 of the DEFINE command.

The user may inspect the contents of the DEFINE file by using PCL to display the predefined words and their equivalent phrases.

**PORTFLAG Command** The PORTFLAG command specifies if IDP will arbitrarily pick a default port when a choice exists.

Format:

PORTFLAG { ON }  
                  { OFF }.

where

**ON** indicates that IDP will not arbitrarily pick a port group by default. If no port group is specified, IDP issues the following message.

\*\*\*164\*\*\* REQUIRED PORT GROUP NOT SPECIFIED

**OFF** indicates that IDP will arbitrarily pick a default port group.

The default mode is OFF.

**SHARE Command** IDP may be assembled with or without the ability to use file sharing. File sharing allows other users to concurrently update or retrieve from a given area in shared mode. In the nonshared mode, other programs may access the area for retrieval, but only one program may access the area for update. No matter what the assembled default may be, the user may set the shared/nonshared mode at run time by using the SHARE command. For further discussion of file sharing refer to Extended Database Management (EDMS) Reference Manual, Publication Number 90 30 12.

Format:

SHARE { ON }  
                  { OFF }.

where

**ON** means IDP will open an area in shared mode by using the EDMS Call OPRETSHD when opening the areas.

**OFF** means IDP will open an area in nonshared mode by using the EDMS Call OPENRET when opening the areas.

The default is an installation dependent parameter. See Program Description 706466-11B00 for information on how to change the default.

**EJECT Command** After a change in value for a total break control item has occurred, new titles and column headings will be printed and if output is to the line printer, a page eject is issued. The user may control the page eject and column titles by the EJECT command.

Format:

EJECT { ON }  
                  { OFF }.

where

**ON** indicates a new page on a total break change.

**OFF** indicates no page change occurs. IDP continues printing without new titles or new column headings.

**END Command** The END command terminates the query session and returns to the Terminal Execution Language (TEL). End-of-file (ESC and F keys on-line) is synonymous with END. In a REPORT file, END or end-of-file return control to IDP, not to TEL.

## APPENDIX A. IDP ERROR MESSAGES

The messages printed by IDP are shown in Table A-1. In summary, messages numbered 1-50 report errors detected in the IDP syntax. They indicate an improper format or an unknown name found while analyzing the IDP commands. Messages numbered 51-58 require a user response on whether to continue or restart the job or to stop execution. Those numbered 100-120 report syntax errors discovered at the command level that cause IDP to return to the point before the last command and ignore the information given in the last command.

Messages numbered 150-199 report conditions discovered during the retrieval phase of IDP, and messages numbered 200 and above indicate fatal errors from which IDP cannot recover. In these instances, IDP returns to the Terminal Executive Language (TEL).

In addition, since IDP uses the Database Manager of the Extended Database Management System (EDMS) to retrieve items from the user's database, IDP may transmit DBM error messages. Please refer to Xerox Extended Data Management System, Publication 90 30 12, for a list of the DBM error codes.

When a syntax error occurs (messages 1-50), the incorrect line will be typed out on the terminal and a dollar sign (\$) will be placed under the point where the error occurred. Unless otherwise indicated, the user should continue his request starting at the word flagged by the \$.

```

IDP (M)
_IDP VERSION B00
:QUERY INVSUB. (M)
:DISPLAY ITEM-NAME, ITEM-CAST WHEN ITEM-NO = '3500'. (M)
  2: DISPLAY ITEM-NAME, ITEM-CAST WHEN ITEM-NO = '3500'.
      $
***9*** THIS ITEM NAME NOT FOUND IN THE SUBSCHEMA
:ITEM-COST WHEN ITEM-NO = '3500'. (M)
  
```

There is no item names ITEM-CAST in the subschema INVSUB. The error occurred after IDP looked at the name ITEM-CAST. The flag (\$) is at the point where the error is detected. The user corrects his command by retyping the incorrect word which should be spelled ITEM-COST.

```

  3: ITEM-COST WHEN ITEM-NO = '3500'.
      $
***12*** PREVIOUS ITEM REQUIRES A GROUP-NAME QUALIFIER; RETYPE ITEM
:ITEM-COST OF ITEM-GROUP WHEN ITEM-NO = '3500'. (M)
  
```

In the subschema INVSUB, the item ITEM-COST requires a group-name qualifier. IDP was expecting a group-name qualifier in the form 'OF/IN group-name'.

Since ITEM-COST is a valid name, the error occurred when IDP did not find the word 'OF' or 'IN' following ITEM-COST. IDP goes back to accept another item-name and ignores the previous item-name. In the next line the user corrects his request. IDP begins processing the request after the final period.

Table A-1. IDP Error Messages

<p>***1*** SUBSCHEMA NAME IN ERROR</p> <p>The file named in the QUERY command 1) does not exist; 2) is not a subschema file; 3) does not have the subschema or name tables; or 4) has a checksum error. Retype correct subschema name. If 3 or 4 is the cause, exit from IDP and rebuild the subschema file.</p>
<p>***2*** SN IN ERROR</p> <p>The serial number given for a file is; 1) greater than four characters; or 2) an illegal name. Retype correct volume serial number.</p>

Table A-1. IDP Error Messages (cont.)

<p><b>***3*** DATABASE PASSWORD IN ERROR</b></p> <p>The database password supplied by the user; 1) is not a text string (enclosed in quotes); 2) is greater than eight characters long; or 3) is not correct. Retype correct password in quotes.</p>
<p><b>***4*** PASSWORD NOT REQUIRED FOR THIS DATABASE</b></p> <p>A password was used but none was required. Information only, no user action necessary.</p>
<p><b>***5*** CIPHERKEY IN ERROR</b></p> <p>The cipher key; 1) is not a text string (enclosed in quotes); or 2) is greater than four characters long. Retype correct cipher key.</p>
<p><b>***6*** FILE NAME IN ERROR</b></p> <p>A file name, account or password has illegal characters. Retype correct file name, account, or password.</p>
<p><b>***7*** SERIAL NUMBER LIMIT EXCEEDED FOR THIS DCB</b></p> <p>The user has given too many pack serial numbers for this DCB. The limits are; 1) three for the subschema; or 2) 16 for each area. Information only, no user action necessary, the flagged serial number is ignored.</p>
<p><b>***8*** FLOATING POINT ITEMS MAY NOT BE SORT KEYS</b></p> <p>Item in SORT command is a floating point item. Information only, no user action necessary, the item is ignored.</p>
<p><b>***9*** THIS ITEM-NAME NOT FOUND IN THE SUBSCHEMA</b></p> <p>Item-name is either misspelled or invalid. Retype correct item-name.</p>
<p><b>***10*** THIS GROUP-NAME NOT FOUND IN THE SUBSCHEMA</b></p> <p>Group-name is either misspelled or invalid. Retype correct group-name.</p>
<p><b>***11*** THIS SET-NAME NOT FOUND IN THE SUBSCHEMA</b></p> <p>Set-name is either misspelled or invalid. Retype correct set-name.</p>
<p><b>***12*** PREVIOUS ITEM REQUIRES A GROUP-NAME QUALIFIER; RETYPE ITEM</b></p> <p>A group-name qualifier is required for the previous item. The previous item is not a unique name in the database and must be qualified. An 'OF' or 'IN' group name qualifier was expected by IDP. Retype the ITEMNAME OF/IN group name.</p>
<p><b>***13*** ONLY ONE WHEN CLAUSE IS ALLOWED IN A DISPLAY COMMAND</b></p> <p>The user has given more than one WHEN clause. Information only, no user action necessary, the second WHEN clause is ignored.</p>
<p><b>***14*** FIRST OR LAST PG OUT OF RANGE; RETYPE (FIRST PG, LAST PG)</b></p> <p>In a SERIAL clause the user specified pages are not within the range of the group or the database. Retype (first-page, last-page).</p>

Table A-1. IDP Error Messages (cont.)

<p><b>***15*** ILLEGAL OPERATOR IN A LOGICAL EXPRESSION</b></p> <p>In the WHEN clause of a DISPLAY command, a legal item-name was found followed by a string of characters not one of the allowable operators: EQ, =, NE, ~=, GT, &gt;, LT, &lt;, GE, &gt;=, LE, &lt;=. Retype one of the above operators.</p>
<p><b>***16*** ALPHABETIC ITEMS MAY NOT BE CROSSFOOTED</b></p> <p>An alphabetic or alphanumeric item was entered as a term in a crossfooted item. Retype correct item.</p>
<p><b>***17*** THIS COMPARISON REQUIRES A NUMERIC LITERAL OR ITEM NAME</b></p> <p>In the WHEN clause of the DISPLAY command, if the item on the left side of the logical expression was numeric, the right side, too, must be numeric. Number can be signed, have decimal point, or be a valid floating-point number. Key in numeric literal.</p>
<p><b>***18*** ERROR IN FILE NAME</b></p> <p>A report file name is either misspelled or invalid. Retype correct file name.</p>
<p><b>***19*** YOU MAY SORT ONLY 16 ITEMS</b></p> <p>A maximum of 16 items may be sorted. Items 17 on are ignored. Information only, no user action necessary.</p>
<p><b>***20*** YOU MAY TOTAL ONLY 16 ITEMS</b></p> <p>A maximum of 16 items may be totaled. Items 17 on are ignored. Information only, no user action necessary.</p>
<p><b>***21*** YOU MAY NOT TOTAL ALPHANUMERIC ITEMS</b></p> <p>The item flagged is alphabetic or alphanumeric and may not be totaled. Information only, no user action necessary, item is ignored.</p>
<p><b>***22*** ILLEGAL VALUE FOR 'TIME' CLAUSE</b></p> <p>The TIME clause in the LIMIT command does not have a valid number (one decimal point accepted). Retype valid number.</p>
<p><b>***23*** ONLY ONE SERIAL DIRECT OR PORT CLAUSE ALLOWED</b></p> <p>Only one of the above mentioned clauses may be specified. The last to be specified is ignored. Information only, no user action necessary.</p>
<p><b>***24*** FIRST PAGE &gt; LAST PAGE; RETYPE (FIRST PAGE, LAST PAGE)</b></p> <p>In a SERIAL clause, SERIAL GROUP-NAME (n, m) n &gt; m. Retype (first-page, last-page).</p>
<p><b>***25*** INTEGER REQUIRED</b></p> <p>An integer value is required. Retype integer.</p>
<p><b>***26*** ILLEGAL NAME</b></p> <p>IDP has detected a character string with an illegal character. Retype name.</p>

Table A-1. IDP Error Messages (cont.)

<p>***27*** THIS SET-NAME PREVIOUSLY CITED</p> <p>The set-name was entered twice in a VIA clause. Information only, no user response necessary.</p>
<p>***28*** UNBALANCED PARENTHESES; RETYPE 'WHEN' CLAUSE</p> <p>The number of right parentheses did not equal the number of left parentheses. Retype WHEN clause.</p>
<p>***29*** ILLEGAL (PAGE-NO, LINE-NO)</p> <p>In a DIRECT clause, the page-number or line-number was illegal. Retype either the line-number or page-number, line-number, depending on the error flag (\$).</p>
<p>***30*** ILLEGAL (FIRST PAGE, LAST PAGE)</p> <p>In a SERIAL clause, the first-page or last-page was illegal. Retype either last-page or first-page, last-page, depending on error flag (\$).</p>
<p>***31*** PERIOD REQUIRED</p> <p>A period was expected. Key in a period.</p>
<p>***32*** NOTHING RETRIEVED</p> <p>Nothing met the users retrieval criteria.</p>
<p>***33*** DEFINED WORD MUST BE ENCLOSED IN QUOTES</p> <p>In a DEFINE command, either the defined word or IDP-phase was not enclosed in quotation marks. Retype with quotation marks.</p>
<p>***34*** DEFINED WORD MUST NOT CONTAIN BLANKS OR SPECIAL CHARACTERS</p> <p>In a DEFINE command, the defined word has embedded or trailing blanks. Retype word without blanks.</p>
<p>***35*** DEFINED WORD EXISTS; USE DELETE TO DELETE THE WORD</p> <p>In a DEFINE command, the defined word exists in the DEFINE file. To redefine the word exit the DEFINE command by keying in a period, use DELETE to delete the word, and then use DEFINE to redefine the word.</p>
<p>***36*** DELETE WORD DOES NOT EXIST</p> <p>In a DELETE command, the word to delete does not exist. Exit DELETE by keying in a period.</p>
<p>***37*** ? MAY APPEAR ONLY WITHIN A REPORT FILE</p> <p>A ? which indicates variable report parameters was keyed in. Information only, no user action necessary.</p>
<p>***38*** YOU MAY NOT HAVE MORE THAN 16 BREAK ITEMS</p> <p>A maximum of 16 items may be break control keys. Information only, no user action necessary.</p>
<p>***39*** YOU HAVE EXCEEDED YOUR HEADER LIMIT</p> <p>Maximum number of header lines has been exceeded. Information only, no user action necessary.</p>

Table A-1. IDP Error Messages (cont.)

<p><b>***40*** ERROR IN BATCH MODE; RUN ABORTED</b></p> <p>An error has occurred while running in batch mode. Information only.</p>
<p><b>***41*** DEFINE PHRASE ERROR; DEFINED PHRASE/WORD/ERROR MSG</b></p> <p>In a previous DEFINE command, DEFINE 'word' as 'IDP-phrase', an error has been found in IDP-phrase when word was used in the present request. See DEFINE COMMAND ERROR DETECTION for further details.</p>
<p><b>***42*** YOU MAY NOT AVERAGE ALPHABETIC ITEMS</b></p> <p>The item flagged is alphabetic or alphanumeric and may not be averaged. Information only, no user response necessary, item is ignored.</p>
<p><b>***43*** YOU MAY NOT HAVE VARIABLE REPORT PARAMETERS IN BATCH MODE</b></p> <p>A ? was found in a report file while running in batch. Run is aborted.</p>
<p><b>***44*** THIS COMPARISON REQUIRES A NON-NUMERIC LITERAL OR ITEM NAME</b></p> <p>In the WHEN clause of the DISPLAY command, the item on the left side of the logical expression was alphabetic or alphanumeric. IDP requires a non-numeric literal enclosed by single quotation marks.</p>
<p><b>***45*** TITLE MUST BE LESS THAN YOUR SPECIFIED WIDTH</b></p> <p>The number of characters in a title text string must be less than the value contained in the WIDTH clause of the OUTPUT command. The title is ignored. Retype TITLE command as OUTPUT, WIDTH = nnn, where nnn will be greater than the title text string. Then retype the TITLE command.</p>
<p><b>***46*** YOU DO NOT HAVE AUTHORITY TO ACCESS THIS ITEM</b></p> <p>The database password does not give retrieve access to the flagged items. Information only.</p>
<p><b>***47*** REPORT IGNORED</b></p> <p>A syntax error was discovered while scanning a report file. Correct error, run IDP again.</p>
<p><b>***48*** THIS REPORT FILE DOES NOT EXIST</b></p> <p>In a REPORT command, the file was nonexistent. Retype correct file-id.</p>
<p><b>***49*** ACCESS TO DEFINE FILE DENIED</b></p> <p>The user does not have write access because of account or other restrictions. The DEFINE or DELETE commands may not be used. Information only.</p>
<p><b>***50*** PATH WARNING MESSAGE INTEGER MUST BE 1 OR 2 OR 3</b></p> <p>User must key in a 1 or 2 or 3.</p>
<p><b>***51*** TIME LIMIT EXCEEDED; QUIT CONTINUE OR RESTART (Q/C/R)?</b></p> <p>Key in Q to quit, C to continue, or R to restart. If user continues, the time limit is reinstated.</p>

Table A-1. IDP Error Messages (cont.)

<p><b>***52*** LINES LIMIT EXCEEDED; QUIT CONTINUE OR RESTART (Q/C/R)?</b></p> <p>Key in Q to quit, C to continue or R to restart. If user continues, the lines limit is reinstated.</p>
<p><b>***53*** QUIT CONTINUE OR RESTART (Q/C/R)?</b></p> <p>This message is issued when the Break key is depressed. Key in Q to quit, C to continue, or R to restart. Quit returns to TEL. Continue resumes processing. Restart ignores the present request and will accept a new request.</p>
<p><b>***54*** WIDTH OF REPORT EXCEEDS LIMIT; QUIT CONTINUE OR RESTART (Q/C/R)?</b></p> <p>The width of the report will exceed the user-specified or default width (132). Key in Q to quit, C to continue, or R to restart. If user continues, no more display items will be accepted, and the item that caused the overflow is ignored.</p>
<p><b>***55*** LIMIT EXCEEDED IN BATCH; QUIT RESPONSE TAKEN</b></p> <p>A limit has been exceeded while running in batch mode. The processor exits. Information only.</p>
<p><b>***56*** BREAK CTL ITEMS ARE NOT SORTED; DO YOU WANT TO SORT (Y/N)?</b></p> <p>The request specifies totaling on break control items and no SORT command has been given. Y will sort the report ascending on all the break control items with break control field 1 major key. N will continue with no sorting.</p>
<p><b>***57*** DEADLOCK HAS OCCURRED; QUIT OR RESTART (Q/R)</b></p> <p>Deadlock has occurred using file sharing. Key in Q to quit or R to restart. Refer to Extended Database Management System Reference Manual for a discussion on file sharing and deadlock.</p>
<p><b>***58*** AREA LIMIT EXCEEDED; QUIT CONTINUE OR RESTART (Q/C/R)</b></p> <p>The user has exceeded the area limit that has been sysgened for the system. Key in Q to quit, C to continue, R to restart. Quit returns to TEL. Continue ignores the last AREA = clause. Restart allows the user to key in another request.</p>
<p><b>***59*** SIZE LESS THAN MINIMUM FOR FLOATING ITEM: SET TO MINIMUM</b></p> <p>The minimum size for a floating short item 14, for a floating long item it is 22. A SIZE = has been specified for a floating point item which is less than these values.</p>
<p><b>***100*** UNABLE TO DECIPHER THIS COMMAND</b></p> <p>The keyword is probably misspelled. Retype command.</p>
<p><b>***101*** QUERY COMMAND MUST PRECEED ABOVE COMMAND</b></p> <p>QUERY command must be specified prior to any command referencing a database item-name and TITLE command. Enter QUERY command.</p>



Table A-1. IDP Error Messages (cont.)

<p>***102*** DISPLAY COMMAND IN ERROR                  ***103*** WHEN CLAUSE IN ERROR, RETYPE 'WHEN' CLAUSE                  ***104*** REPORT COMMAND IN ERROR                  ***105*** SORT COMMAND IN ERROR                  ***106*** TOTAL COMMAND IN ERROR                  ***107*** LIMIT COMMAND IN ERROR                  ***108*** QUERY COMMAND IN ERROR                  ***109*** A PASSWORD IS REQUIRED FOR THIS DATABASE; RETYPE QUERY                  ***110*** DIRECT CLAUSE IN ERROR; RETYPE DIRECT CLAUSE                  ***111*** SERIAL CLAUSE IN ERROR; RETYPE SERIAL CLAUSE                  ***112*** VIA CLAUSE IN ERROR; RETYPE VIA CLAUSE</p> <p>For any of the above error messages, the detected error is undetermined. Retype entire command or clause.</p>
<p>***113*** (This message number is reserved.)</p>
<p>***114*** ALPHABETIC/NUMERIC INCONSISTENCY</p> <p>In a WHEN clause the user has tried to compare numeric and alphanumeric or alphabetic items or in an crossfooted item, an alphanumeric or alphabetic item was included. Retype entire WHEN clause or cross-footed item.</p>
<p>***115*** ERROR IN ARITHMETIC EXPRESSION</p> <p>An error has occurred while formatting a crossfooted item. Retype crossfooted item.</p>
<p>***116*** ERROR IN DELETE COMMAND</p> <p>Detected error is undetermined. Retype DELETE command.</p>
<p>***117*** KEY IN REPORT PARAMETERS AGAIN</p> <p>In a report file with variable parameters, an error has occurred in the user response. Retype parameters.</p>
<p>***118*** NO DISPLAY OR TOTAL ITEMS IN QUERY</p> <p>No retrieval items were entered. Restart QUERY.</p>
<p>***119*** TOPOLOGY COMMAND MUST SPECIFY ON OR OFF</p> <p>Key in ON or OFF.</p>
<p>***120*** ERROR IN DEFINE COMMAND</p> <p>Detected error undetermined. Retype DEFINE command.</p>
<p>***121*** ILLEGAL ZONED/DECIMAL NUMBER; TOTAL IGNORED</p> <p>While totaling a zoned or packed decimal number, an illegal digit was detected, the total is ignored. This message will appear on M:DO, and may appear within the body of the report.</p>
<p>***122*** SHARE MUST SPECIFY ON OR OFF</p> <p>Key in ON or OFF.</p>

Table A-1. IDP Error Messages (cont.)

<p>***123*** OUTPUT COMMAND ERROR</p> <p>Detected error undetermined. Retype OUTPUT command.</p>
<p>***124*** I/O ERROR xx yy ON DEFINE FILE</p> <p>Refer to CP-V Batch Processing Manual for meaning of the error codes.</p>
<p>***125*** IMPOSSIBLE CONDITIONS IN WHEN CLAUSE; RETYPE WHEN CLAUSE</p> <p>The user has keyed in constraints which could never be satisfied, for example DISPLAY ITEM WHEN ITEM EQ 1 AND ITEM NE 1. Retype the entire WHEN clause.</p>
<p>***126*** PATH WARNING MESSAGE ERROR</p> <p>Detected error undetermined. Retype path warning message again.</p>
<p>***127*** PORTFLAG MUST SPECIFY ON OR OFF</p> <p>Key in ON or OFF.</p>
<p>***128*** EJECT MUST SPECIFY ON OR OFF</p> <p>Key in ON or OFF.</p>
<p>***129*** NON-REPORT FILE NAME TRUNCATED TO SEVEN CHARACTERS</p> <p>In the NON-REPORT file only the first seven characters of the users file-id are used.</p>
<p>Message numbers 130 through 149 are reserved.</p>
<p>***150*** I/O ERROR xx/yy ON RETRIEVED OUTPUT FILE</p> <p>CP-V I/O error xx, subcode yy, was encountered when writing retrieved output for sorting. Closes file and returns to IDP for the next command. Refer to CP-V Batch Processing Manual for meaning of the error codes.</p>
<p>***151*** EDMS ERROR xx</p> <p>Error code xx returned from the Database Manager in EDMS. Returns to Report processor with end-of-file indication if output is not to be sorted. If sort is required, closes retrieved output file and returns to IDP for the next command. Refer to EDMS Reference Manual for meaning of error code.</p>
<p>***152*** NOT ENOUGH MEMORY FOR RETRIEVAL PROGRAM</p> <p>IDP unable to acquire sufficient dynamic memory for generating the retrieval execution program. Returns to IDP for the next command.</p>
<p>***153*** INVALID RETRIEVAL KEY LITERAL MODE</p> <p>The literal criterion for a packed or zoned decimal on binary CALC or invert key item was expressed in floating-point notation. Returns to IDP for the next command.</p>
<p>***154*** UNABLE TO USE ALL CITED VIA SETS IN QUERY</p> <p>Information only. IDP continues.</p>
<p>***155*** TOO MANY AREAS IN REQUEST</p> <p>The query requires more than four areas. Returns to IDP for the next command.</p>

Table A-1. IDP Error Messages (cont.)

<p><b>***156*** UNABLE TO OPTIMIZE WITH CALC KEY OR GROUP INDEX</b></p> <p>The area-as-owner group will be used as the retrieval port (if one exists). Information only. IDP continues.</p>
<p><b>***157*** QUERY ITEMS HAVE NO LOGICAL RELATIONSHIP</b></p> <p>IDP was unable to build a network containing all query groups (see Appendix B). Returns to IDP for the next command.</p>
<p><b>***158*** WRONG REF-CODE FOR DIRECT GROUP</b></p> <p>The group found at the page and line number specified in the DIRECT clause was not an occurrence of the group named in the clause. Returns to syntax checker error entry.</p>
<p><b>IDP VERSION B00</b></p> <p>This is message 159.</p>
<p><b>***160*** ILLEGAL DECIMAL DIGIT</b></p> <p>An illegal packed decimal number was encountered, the field is filled with asterisks and the report continues. Note that this message will appear on M:DO and may be within the body of the report.</p>
<p><b>***161*** MULTIPLE LAMBDA STRUCTURES IN QUERY</b></p> <p>IDP will handle only one lambda structure in the covering network. Control is returned to the user to reformulate the query.</p>
<p><b>***162*** ILLEGAL CONSTRAINTS IN LAMBDA STRUCTURE</b></p> <p>Constraints between legs in a lambda structure must be either all "OR" or all "AND".</p>
<p><b>***163*** REPORT TRUNCATION</b></p> <p>While formatting a report, extra spacing was allocated to ensure an item was not displayed on two lines causing some items to be truncated. The user may exit from IDP, then reset the PLATEN setting or increase the width of the report before rerunning.</p>
<p><b>***164*** REQUIRED PORT GROUP NOT SPECIFIED</b></p> <p>PORTFLAG was ON and the port group was not specified. IDP returns to the system checker and user may enter another request.</p>
<p><b>***165*** DISPLAY COMMAND DID NOT DESCRIBE COMPLETE PATH</b></p> <p>If Path Warning flag is = 1, IDP returns to syntax checker and user may enter another request. If Path Warning flag = 2, message 166 is issued.</p>
<p><b>***166*** CONTINUE OR RESTART (C/R)</b></p> <p>Key in C to continue or R to restart. See message 165.</p>
<p>Message numbers 167 through 200 are reserved.</p>

Table A-1. IDP Error Messages (cont.)

<p><b>***201*** IRRECOVERABLE I/O ERROR; IDP CANNOT CONTINUE</b></p> <p>An I/O error other than a nonexistent file has occurred. IDP returns to TEL.</p>
<p><b>***202*** REPORT FILE MAY NOT CONTAIN A REPORT COMMAND</b></p> <p>Nesting of REPORT commands is not allowed. IDP returns to TEL.</p>
<p><b>***203*** NOT ENOUGH SPACE TO BUILD TABLES; IDP CANNOT CONTINUE</b></p> <p>More memory space is required. IDP returns to TEL.</p>
<p><b>***204*** IDP SYSTEM ERROR; IDP CANNOT CONTINUE</b></p> <p>An error has occurred in the IDP program. Diagnostic data will be dumped out on the line printer to aid the Xerox analyst with the problem. IDP returns to TEL.</p>
<p><b>***205*** I/O ERROR ON REPORT FILE; IDP CANNOT CONTINUE</b></p> <p>An error, other than a nonexistent file, was encountered opening or reading the report file. IDP returns to TEL.</p>
<p>Message number 206 is reserved.</p>
<p><b>***207*** IDP SYSTEM ERROR ON SORT</b></p> <p>IDP returns to TEL.</p>
<p><b>***208*** MEMORY MANAGEMENT SYSTEM ERROR; RUN ABORTED</b></p> <p>IDP returns to TEL.</p>

## APPENDIX B. IDP LIMITATIONS

This appendix describes some of the types of queries and data structures that IDP is not able to handle.

### 1. Implicit Commands Not Allowed

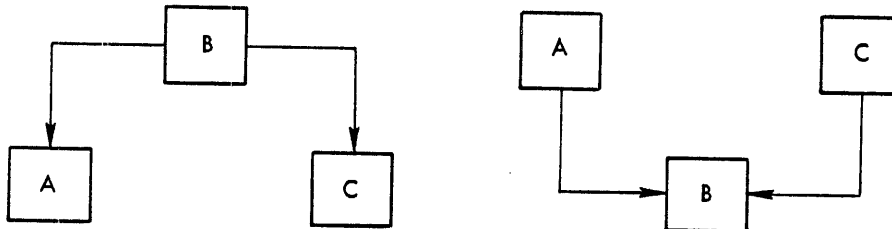
All IDP commands must be explicit. An example of what is meant by an implicit command would be to query DISPLAY PART WHEN PART = 'ASSEMBLY' and expect a list of the components constituting the assembly, such as a bill of materials parts explosion report.

### 2. Multiple Occurrences of a Single Group Cannot be ANDed to Satisfy a Single Constraint

IDP constructs a "logical record" consisting of one occurrence of each request group. The values of the constraint items in this logical record are compared with the selection criteria to determine whether or not the target items in this logical record qualify for inclusion in the report. Since only one occurrence of each group is available at any one time, and since a given item can have only one discrete value, more than one ANDed constraint value for the same item will always result in a "false" condition. For example, IDP cannot satisfy the AND condition of a request such as DISPLAY NAME COURSE = 'ENGLISH' AND COURSE = 'BOTANY'.

### 3. Query Network Paths are Terminated by Cited Groups

The covering network built by IDP is one in which every complete path begins and ends with a cited group. A cited group is defined as one that contains either a constraint item, or a target item, or both, or is an owner or member of a cited VIA set. A path is defined as a series of groups which can be traversed by using either successive FINDNs or successive HEADs, but not both. The limitation can best be explained by an illustration. Given one of the following simple structures:



Unless group B is cited, data in groups A and C cannot be associated.

### 4. Fixed Point Numeric Item Retrieval Key Values Cannot be Expressed in Floating-Point Notation

Because the range of numeric values expressible in floating-point notation is much greater than 30 packed decimal digits or 31 bits in binary, the numeric literal used with either a packed or zoned decimal or a binary item used as a retrieval calc key or inverted group key must be expressed in fixed point notation. (See the EDMS manual for a complete definition of these terms). This limitation does not exist for non-key numeric items.

5. A maximum of 16 items may be totaled.
6. No more than 16 items may be specified as sort keys.
7. No more than 16 items may be specified as break control keys.
8. The subschema used by IDP must have been created by a version of DMSFDP which provides the table of item names as part of the subschema.
9. Multiple occurrence items (those defined in the subschema with an OCCURS clause) cannot be properly processed.

## APPENDIX C. RETRIEVAL STRATEGY

### IDP Topology Analysis

The topology analysis module of IDP will determine the sets to be traversed and groups to be retrieved in order to satisfy a display request. This module has as input a list of all data items referenced in the SORT, TOTAL or DISPLAY command; the data items and types of constraints in the WHEN clause; the group name specified in a SERIAL, DIRECT or PORT clause; a list of sets specified in a VIA clause and the subschema specified in the QUERY command. Using this information, the module first selects a port group, that is the first group to be retrieved. It then accesses the subschema adding sets to the port group which will connect the port with other groups containing cited items until all cited items have been taken care of.

### Port Selection

Selection of a port group is done on the following basis, listed first to last:

1. A group specified in a DIRECT, SERIAL or PORT clause.
2. A group which may be retrieved directly based upon its CALC or INDEX control items or an INVERT item being specified in the WHEN clause.
3. A set defined with the AREA AS OWNER which provides a direct path to a group containing cited data items.
4. One of the groups containing cited data items will be selected as a default port.

If the port group is selected due to a SERIAL clause without the page range option, IDP will use the page range specified by the subschema in a FINDS command to access the port group. However, if the selected port group has INDEXED location mode IDP will use a FINDFRST command followed by FINDN to retrieve the port. If the SERIAL clause includes a page range then IDP will use a FINDS command for just the specified page range to retrieve the port group.

A group may be selected as a port group for direct retrieval if all of its CALC or INDEX control items or an INVERT item are included as constraints in the WHEN clause. In order for them to be used the constraint must contain an EQUAL relationship to one or more specified literals. Thus, 'data-control-item EQ 10, 11, 12' would satisfy for direct retrieval while 'data-control-item GE 10; LE 12' would not. It is possible that the items cited in the WHEN clause may provide more than one group as a direct port. The module will select the first group it finds to satisfy the requirements unless the PORT clause specifies which group to use. The PORT clause may thus be used to cause selection between multiple groups which are candidates as a port based upon the constraints of the WHEN clause. If a PORT clause is used and the constraint values in the WHEN clause do not provide for direct retrieval of the specified port, then the PORT clause is treated as if it were a SERIAL clause without a page range specified.

If the topology module selects a default port either through a set with AREA AS OWNER or for a serial search then the retrieved data items resulting from the selected topology may not be in the order expected by the user. The topology module may be prevented from selecting a default port through use of the PORTFLAG command. The format of this command is:

PORTFLAG {ON  
OFF}

If the port flag is ON topology will only use the first two criteria listed above for selecting a port. If no port is found, an error message is given and processing of the DISPLAY command is terminated.

### Set Selection

Once a port group has been selected, topology will select sets of which the port group is AN OWNER OR MEMBER, to form branches to other groups containing cited items. After a new group has been established, branches are formed from this group to other required groups. The process then is one of forming branches from the current node (group) to a new node, replacing current node with the new node and continuing until groups containing all cited

data items have been established as nodes. During the process of creating branches, the module first tests sets in which the current node is a member. Thus, preference is given in the topology to traversing a set using a FINDM command over traversal using a FINDN command.

It is during the process of selecting sets that those sets specified in a VIA clause are used. The module first looks at all sets of which the current node is a member to find a set specified in the VIA clause. If none is found one of these sets will be picked as a default and inserted in the branch as a trial path. The owner group of the set is marked as a potential participant of the branch. All sets of which the potential participant is a member are then examined, again, for a set listed in the VIA clause. This process continues until a cited set is found or all sets above the current node in the subschema have been examined. If a cited set is found, all groups marked as potential participants are changed to nodes and normal processing continues at the last node established. If no cited set is found, then all potential branch participants are removed and the process is repeated from the current node only now searching sets in which the current node group is defined as owner. If no branches can be established off of the current node in either an up or downward direction then topology will make the previous node current and continue to search for sets cited in the VIA clause. Use of the PORT command and the VIA clause thus allow the user to control the topology that is generated by IDP to satisfy a query request.

If the VIA clause is not used the module proceeds in the same way as described above, however, instead of adding branches due to sets cited in the VIA clause, node groups are added which contain items cited in the query request.

During the process of adding sets to the topology the module will sometimes add sets which are not specified in the VIA clause and do not contain cited data items. These sets are only added when necessary to complete a branch in one direction between two cited sets or groups containing cited data items. The user may control the addition of these sets through use of the path warning message command. The format of this command is:

PATH WARNING MESSAGE EQUALS  $\left\{ \begin{array}{l} 1 \\ 2 \\ 3 \end{array} \right\}$

The value '3' is the default and allows topology to add any sets it finds necessary to complete a branch. The value '2' declares that if a set is added which was not specified in a VIA clause or whose owner or member do not contain cited data items then the module will inform the user and the user will have the option of terminating processing of the display command or preceding with its execution. The sets added may be determined by displaying the topology table (see below). The value '1' specifies that if such sets are added the user will be informed and processing of the display command is automatically terminated.

## Topology Table Display

The topology module builds a table containing the groups and sets that will be accessed by the retrieve module to satisfy the display request. This table may be displayed through use of the topology command. The format of this command is:

TOPOLOGY  $\left\{ \begin{array}{l} \text{ON} \\ \text{OFF} \end{array} \right\}$

'OFF' is the default and results in no display while 'ON' causes the table to be displayed on the terminal.

Following is a sample display and its interpretation.

A	BCDE
0 1 0 0	* 1 0 1
0 1 0 3	0 0 0
0 1 0 5	2 0 0
0 1 0 5	* 0 1 0
0 1 0 4	2 1 0
0 1 0 4	0 0 0
0 1 0 6	0 0 0
0 1 1 0	2 0 0
0 1 1 0	* 0 1 0
0 1 0 7	0 1 0
0 1 0 2	3 1 0

Column A	List of the group numbers of the groups to be retrieved. The first group listed is the port.
Column B	Branch indicator. Retrieval is in one direction, that is head or next, until a node is encountered at which point the direction may be reversed.
Column C	Reason for including group in branch. <ul style="list-style-type: none"> <li>0 group was included to complete a branch or it is the node from a previous branch.</li> <li>1 group contains items cited in the WHEN clause.</li> <li>2 group contains items for display, arithmetic operations or as sort control keys.</li> <li>3 combination of class 1 and 2.</li> </ul>
Column D	Type of traversal of branch; '0' equals FINDN, '1' equals FINDM.
Column E	Indicates type of port retrieval and is only significant for the first entry. <ul style="list-style-type: none"> <li>0 AREA OWNER set.</li> <li>1 FINDG of CALC or INDEX group.</li> <li>2 FINDX using secondary index.</li> <li>3 FINDD or serial retrieval.</li> </ul>

The preceding sample may thus be interpreted as follows:

1. Retrieve the port group number 100 using a FINDG command. A GET command will be issued on group 100 as it contains items specified in the WHEN clause.
2. Traverse sets from group 100 to group 103 and then to group 105 using a FINDN command. A GET command will be issued on group 105 as it contains data items cited in the display request.
3. Using a FINDM command traverse from group 105 to group 104. Issue a GET on group 104.
4. From group 104 use a FINDN command to group 106 and then to group 110. Issue a GET on group 110.
5. From group 110 traverse to group 107 and then to group 102 using a FINDM command. A GET will be issued on group 102 as it contains items both to be displayed and used as constraints.

## Lambda Structure

There is a particular structure which may exist in a topology which deserves special attention. This structure occurs whenever two or more branches in a next direction are used from the same node group. It deserves special attention due to possible ambiguities that may result from retrieval using this structure. This structure has been termed the lambda structure, however, it is not limited to two branches as appear in the character, lambda. Below is a data-base diagram of a sample lambda structure (see Figure C-1).

A topology with traversal from GROUPA to GROUPB using FINDN and from GROUPA to GROUPE using FINDN would contain a lambda structure. When a lambda structure is present, possible ambiguities may result in the display request due to implied relationships that may exist between group occurrences contained in the branches of the structure. A display request containing constraints in both GROUPB and GROUPE of the sample may assume some relationship between occurrences of GROUPB and GROUPE which is not explicitly defined by the structure. Special retrieval logic is thus used within IDP when a lambda structure exists in the topology.



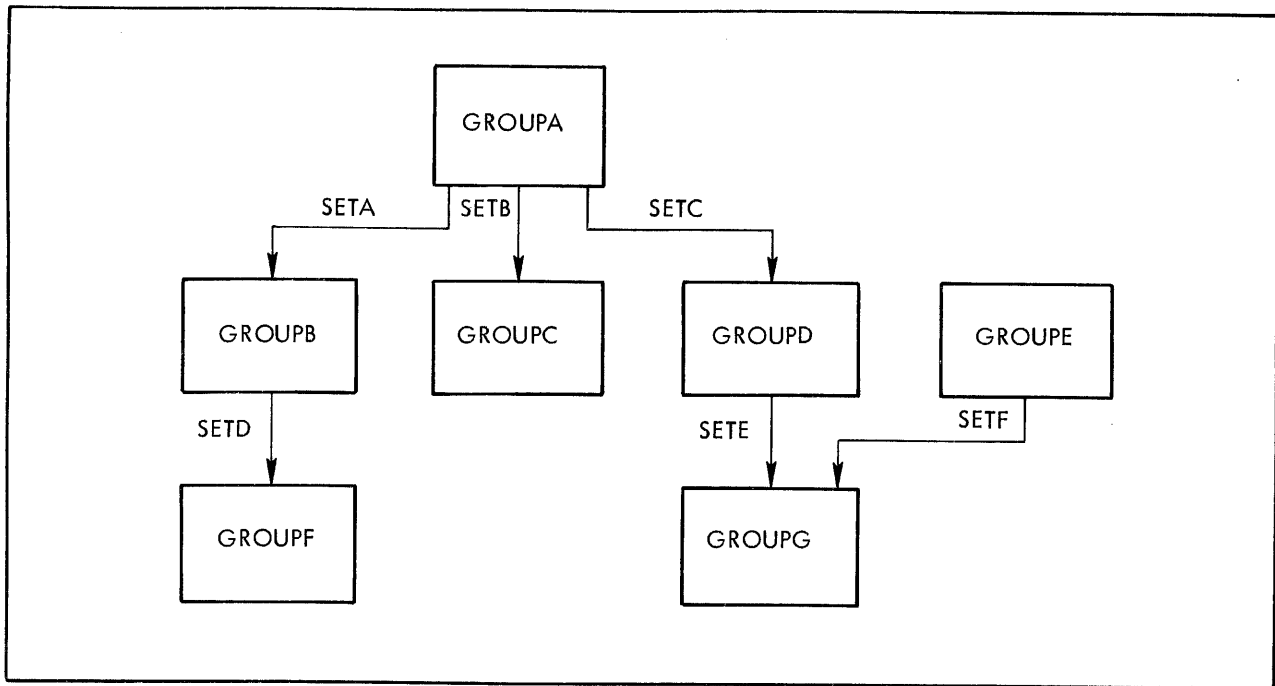


Figure C-1. Database Diagram of a Sample Lambda Structure

For purposes of this logic a branch of the lambda structure is that total path which starts in a next direction and proceeds through one or more groups and may then change direction to retrieve one or more additional groups. In the above sample there are three branches from GROUPA.

1. SETA to GROUPB and SETD to GROUPE.
2. SETB to GROUPE.
3. SETC to GROUPD and SETE to GROUPE and then SETF to GROUPE.

Two or more of these branches or their subsets must be present to constitute a lambda structure for the sample.

Some minimal constraints have been placed on the ability of IDP to handle the lambda structure. These are:

1. A maximum of five branches from the node group are allowed.
2. Only one lambda structure is permitted in a topology.
3. If data items cited in the WHEN clause are contained in branches of the lambda structure then Boolean connectors between items on a branch may be of mixed type (AND or OR) however, connectors between items on different branches must all be of one type.

Following are examples of legal and illegal WHEN clauses using the above sample diagram.

#### LEGAL

1. WHEN item OF GROUPB EQ value AND item OF GROUPE GT value...
2. WHEN item OF GROUPB LT value AND item OF GROUPE EQ value OR item of GROUPE GE value...

#### ILLEGAL

When item OF GROUPE EQ value OR item OF GROUPE LT value AND item OR GROUPE NE value OR item OF GROUPD GT value...

## Lambda Structure Retrieval

As was mentioned previously a database may contain implicit relationships between group occurrences on different branches of the lambda structure. As these relationships are implicit, IDP would normally require explicit directives from the user as to how the branches of the lambda structure are to be traversed. The retrieval module of IDP however, contains logic to separate retrieval from the lambda structure into nine classes based upon the constraint and display items contained in the structure. Following is a description of the nine classes and the retrieval action of each. The examples given are based upon the above sample database diagram.

### Class 1

- Criteria: There are no constraint items in any branch of the lambda.
- Action: All group occurrences of all branches are retrieved once for display.
- Example: DISPLAY item of GROUPB item OF GROUPE SERIAL ON GROUPA.

### Class 2

- Criteria: Only one branch contains constraint items and that branch does not contain any display items.
- Action: The occurrences on the constraint branch are examined for a match to the constraints. If a match is found then all occurrences on other branches are retrieved once for display.
- Example: DISPLAY item OF GROUPE WHEN item OF GROUPB relation value SERIAL ON GROUPA.

### Class 3

- Criteria: Only one branch contains constraint items. The constraint branch also contains items to be displayed.
- Action: The occurrences on the constraint branch are examined for a match to the constraints. If a match is found then all occurrences on other branches are retrieved once for display. All occurrences on the constraint branch that match the constraint are used once for display.
- Example: DISPLAY item OF GROUPE item OF GROUPB WHEN item OF GROUPE relation value SERIAL ON GROUPA.

### Class 4

- Criteria: More than one branch contains constraint items which are connected by an 'AND' condition however, NO constraint branch contains a display item.
- Action: Occurrences on constraint branches are permuted. Each combination is examined for a match to the constraints. When a match is found the permuting is terminated and all occurrences containing display items are retrieved once.
- Example: DISPLAY item OF GROUPA item OF GROUPD WHEN item of GROUPB relation value 'AND' item of GROUPE relation value SERIAL ON GROUPA.

### Class 5

- Criteria: All branches contain constraint items connected by an 'AND' condition and all branches contain display items.
- Action: All occurrences on all branches are permuted. Each combination is examined for a match to the constraints. Matching occurrences are used in a display.
- Example: DISPLAY item OF GROUPB item OF GROUPE WHEN item of GROUPB relation value 'AND' item OF GROUPE relation value SERIAL ON GROUPA.

### Class 6

- Criteria: All branches contain constraint items connected by an 'AND' condition and some but not all contain display items.
- Action: Occurrences from the branches containing display items are permuted with those from the branches not containing display items. Each combination is tested for a match to the constraints. When a match is found the requested items are displayed. Permuting is then restarted with the next occurrences on the display branches and the first occurrences from the non-display branches.
- Example: DISPLAY item OF GROUPD WHEN item OF GROUPD relation value 'AND' item OF GROUPB relation value SERIAL ON GROUPA.

### Class 7

- Criteria: More than one branch contains constraint items connected by an 'AND' condition. All constraint branches also contain display items. In addition, some branch contains display items only.
- Action: Permutation and display of occurrences from constraint branches is carried out as in Class 5. In addition, if a set of qualifying occurrences is found, all occurrences on nonconstraint branches are retrieved once for display.
- Example: DISPLAY item of GROUPD item OF GROUPC item of GROUPB WHEN item of GROUPB relation value 'AND' item OF GROUPC relation value SERIAL ON GROUPA.

### Class 8

- Criteria: More than one branch contains constraint items connected by an 'AND' condition. Some but not all constraint branches also contain display items. Some branch contains display items only.
- Action: Permutation and display of occurrences from constraint branches is carried out as in Class 6. In addition, if a qualifying set of occurrences is found, all occurrences on nonconstraint branches are retrieved once for display.
- Example: DISPLAY item OF GROUPD item OF GROUPC WHEN item of GROUPD relation value 'AND' item OF GROUPB relation value SERIAL ON GROUPA.

### Class 9

- Criteria: More than one branch contains constraint items which are connected by an 'OR' condition.
- Action: The occurrences on all constraint branches are tested for a match to the constraints. If a match is found, then all occurrences containing display items are retrieved once for display, that is the display reverts to Class 1.
- Example: DISPLAY item OF GROUPF item OF GROUPG WHEN item OF GROUPB relation value OR item OF GROUPC relation value OR item OF GROUPD relation value SERIAL ON GROUPA.

DECEMBER, 1975

CORRECTIONS TO IDP REFERENCE MANUAL

PUBLICATION NO. 90 30 66B-1(12/75)

The attached pages contain information on the retrieval strategy of IDP. All of the attached pages should be inserted into your manual as Appendix C and the table of contents in your manual should be replaced with page iii.

These changes will be incorporated into the next edition of the manual.

Revision bars in the margins of replacement pages identify changes.

## Reader Comment Form

We would appreciate your comments and suggestions for improving this publication			
Publication No.	Rev. Letter	Title	Current Date
How did you use this publication? <input type="checkbox"/> Learning <input type="checkbox"/> Installing <input type="checkbox"/> Sales <input type="checkbox"/> Reference <input type="checkbox"/> Maintaining <input type="checkbox"/> Operating		Is the material presented effectively? <input type="checkbox"/> Fully Covered <input type="checkbox"/> Well Illustrated <input type="checkbox"/> Well organized <input type="checkbox"/> Clear	
What is your overall rating of this publication? <input type="checkbox"/> Very Good <input type="checkbox"/> Fair <input type="checkbox"/> Very Poor <input type="checkbox"/> Good <input type="checkbox"/> Poor		What is your occupation?  	
Your other comments may be entered here. Please be specific and give page, column, and line number references where applicable. To report errors, please use the Xerox Software Improvement or Difficulty Report (1188) instead of this form.			
		Your name & Return Address  	

PLEASE FOLD AND TAPE—  
NOTE: U. S. Postal Service will not deliver stapled forms

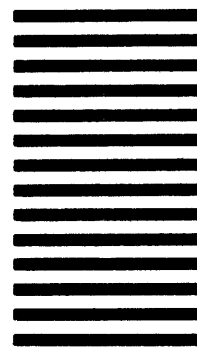


NO POSTAGE  
NECESSARY  
IF MAILED  
IN THE  
UNITED STATES

**BUSINESS REPLY MAIL**  
FIRST CLASS PERMIT NO. 59153 LOS ANGELES, CA 90045

POSTAGE WILL BE PAID BY ADDRESSEE

HONEYWELL INFORMATION SYSTEMS  
5250 W. CENTURY BOULEVARD  
LOS ANGELES, CA 90045



ATTN: PROGRAMMING PUBLICATIONS

**Honeywell**

CUT ALONG LINE  
FOLD ALONG LINE  
FOLD ALONG LINE

**Honeywell Information Systems**

In the U.S.A. 200 Smith Street, MS 486, Waltham, Massachusetts 02154

In Canada 2025 Sheppard Avenue East, Willowdale, Ontario M2J 1W5

In Mexico Avenida Nuevo Leon 250, Mexico 11, D.F.

18487, 2.5C677, Printed in U.S.A.

XQ27, Rev. 0