DATATRIEVE-11 Pocket Guide



DATATRIEVE-11 Pocket Guide

Order No. AA-X103A-TK

September 1983

This guide contains useful information about DATATRIEVE-11.

| Operating System and Version: | RSTS/E | |
|-------------------------------|---------------|----|
| | RSX-11M-PLUS | |
| | RSX-11M | |
| Software Version: | DATATRIEVE-11 | VS |

digital equipment corporation, maynard, massachusetts

The information in this document is subject to change without notice and should not be construed as a commitment by Digital Equipment Corporation. Digital Equipment Corporation assumes no responsibility for any errors that may appear in this document.

The software described in this document is furnished under a license and may be used or copied only in accordance with the terms of such license.

No responsibility is assumed for the use or reliability of software on equipment that is not supplied by DIGITAL or its affiliated companies.

Copyright © 1983 by Digital Equipment Corporation. All Rights Reserved.

The following are trademarks of Digital Equipment Corporation:

MASSBUS PDP P/OS Professional DEC DECwriter DIBOL UNIBUS VAX VMS VT Rainbow RSTS RSX DECmate DECsystem-10 DECSYSTEM-20 DECUS Work Processor

Contents

Page

| Edit String Characters | | | .1 |
|--|---|--|----|
| Arithmetic, Date, and String Expressions | • | | .5 |
| Statistical Expressions | | | .7 |
| Relational Operators and Boolean Expressions | | | .8 |
| Record Selection Expressions | | | 10 |
| Keywords | | | 12 |
| Access Control Privileges | | | 58 |
| Calls | | | 59 |

Syntax Format Conventions

| UPPERCASE WORDS | are DATATRIEVE keywords. |
|--------------------|---|
| lowercase words | are entries that you provide. |
| [](brackets) | enclose optional entries of a syntax format. When they enclose vertically stacked entries, brackets indicate that you can select one of the enclosed entries. |
| { } (braces) | enclose mandatory entries of a syntax format. When they enclose vertically stacked entries, braces indicate that you must choose one of the enclosed entries. |
| (ellipsis) | indicates that you can repeat the preceding syntax element. |
| <>(angle brackets) | enclose an argument that is an ASCII string. |

Edit String Characters

| Edit String Character | Description | |
|--------------------------|--|--|
| Replacem | ent | |
| Х | Replaced by one alphanumeric character; left justified; trailing character positions blank filled. | |
| Т | Replaced by text. Length of lines indicated by number of Ts. | |
| 9 | Replaced by one digit. Digits right justified; leading character positions zero filled. | |
| Z | Replaced by a space if it matches a leading zero. Otherwise, replaced by a digit. | |
| * | Replaced by $*$ if it matches a leading zero. Otherwise, replaced by a digit. | |
| Alphanumeric Insertion | | |
| + | Plus sign inserted. | |
| - | Hyphen inserted. | |
| • | Period inserted. | |
| , | Comma inserted. | |

1

| Edit String Character | Description | |
|------------------------------------|--|--|
| Numeric I | nsertion | |
| + | If only one specified, replaced by a plus sign if field content is positive; minus sign if negative. | |
| _ | If only one specified, replaced by blank if field content is positive; minus sign if negative. | |
| | Decimal point inserted. | |
| , | Comma inserted if digits left of comma are not suppressed zeros. | |
| CR | CR inserted if field content is negative. | |
| DB | DB inserted if field content is negative. | |
| Alphanumeric and Numeric Insertion | | |
| В | Blank inserted. | |
| 0 | Zero inserted. | |
| \$ | Dollar sign inserted if only one specified. | |
| % | Percent sign inserted. | |
| 1 | Slash inserted. | |
| | | |

Numeric Floating Insertion

- \$ If more than one specified left of other edit string characters, leading zeros suppressed, and one \$ displayed at left.
- + If more than one specified left of other edit string characters, leading zeros suppressed, and sign of field's value displayed at left.
- If more than one specified left of other edit string characters, leading zeros suppressed. If value of field is negative, minus sign displayed at left.

Date Replacement

- D Replaced by the corresponding digit of the day of the month.
- M Replaced by the corresponding letter of the name of the month.
- N Replaced by a digit of the number of the month.
- Y Replaced by a significant digit of the numeric year (for example, YY = 83).
- J Replaced by the corresponding digit of the Julian date.

Edit String Characters (Cont.)

| Edit String Character | Description |
|--------------------------|--|
| W | Replaced by the corresponding letter from the name of the day of the week. |
| В | Replaced by a blank. |
| / | Slash inserted. |
| - | Hyphen inserted. |
| | Period inserted. |

Arithmetic, Date, and String Expressions

| Arithmetic Operators | | |
|-------------------------|--|--|
| Operation | | |
| Addition | | |
| Subtraction or negation | | |
| Multiplication | | |
| Division | | |
| | | |

| Date Value Expressions | |
|------------------------|---------------------|
| Expression | Value Returned |
| "TODAY" | Current system date |
| | |
| | |
| | |

| Prompting Value Expressions | | |
|-----------------------------|--|--|
| Expression | Purpose | |
| *."prompt-name" | Prompts for a value. Once for each evaluation of the expression in a loop. | |
| **."prompt-name" | Prompts for a value. Once for execution of entire statement. | |

| Concatenated Expressions | | |
|--------------------------|---|-------------------------|
| Expression | Purpose | Example |
| 1 | Leaves literals as they are | "ABC "¦"DEF" ABC DEF |
| 11 | Suppresses trailing spaces of first literal; does nothing to leading spaces of second literal | "ABC "!!"DEF" ABCDEF |

Statistical Expressions

Statistical functions compute values based on a value expression evaluated for each record in a record stream.

| Values Derived with Statistical Functions | |
|---|---|
| Function | Value |
| MAX | Largest value of value expression |
| MIN | Smallest value of value expression |
| AVERAGE | Average value of value expression |
| TOTAL | Total value of value expression |
| COUNT | Number of records in CURRENT collection, named collection, or record stream |

DTR> FIND FIRST 10 YACHTS; PRINT AVERAGE LOA, TOTAL LOA DTR> PRINT FAMILIES WITH NUMBER_KIDS = MAX NUMBER_KIDS OF FAMILIES

Relational Operators and Boolean Expressions

| Relationship of Values in Boolean | Relational Operator | Boolean Expression RIG = "YAWL" RIG = "MS","YAWL" | | | |
|---|----------------------------|---|--|--|--|
| Exact match (case sensitive) | = EQUAL EQ | | | | |
| No match (case sensitive) | NE NOT_EQUAL | RIG NE "YAWL" RIG NE "MS", "YAWL" | | | |
| Substring matches (not case sensitive) | CONT CONTAINING | RIG CONT "awl" RIG CONT "ms", "awl" | | | |
| Substring does not match (not case sensitive) | NOT CONT NOT CONTAINING | RIG NOT CONT "ms" RIG NOT CONT "ms", "ya" | | | |
| Beginning substring matches (case sensitive) | STARTING WITH | RIG STARTING WITH "M" RIG STARTING WITH "M", "Y" | | | |
| First value is greater | > or GT GREATER_THAN | PRICE > 50000 | | | |

| First value is greater than or equal | GE GREATER_EQUAL | PRICE GE 50000 | | | |
|--|----------------------|-------------------------------|--|--|--|
| First value is less than or equal | LE LESS_EQUAL | PRICE LE 20000 | | | |
| First value is less | < or LT LESS_THAN | PRICE < 20000 | | | |
| First value is between the two values or equal to either | BT BETWEEN | PRICE BETWEEN 30000 AND 54000 | | | |
| Field value is in the table | IN table-name | RIG IN RIG_TABLE | | | |
| Field value is not in the table | NOT IN table-name | RIG NOT IN RIG_TABLE | | | |
| Record stream is not empty | ANY rse | FAMILIES WITH ANY KIDS | | | |
| Record stream is empty | NOT ANY rse | FAMILIES WITH NOT ANY KIDS | | | |

Record Selection Expressions

Form a record stream according to the conditions specified.

[FIRST n ALL [context-var IN] domain-name collection-time list-name

[WITH boolean-expression]

```
[SORTED BY sort-key [,...]]
```

DTR> FIND PRICY-BOATS IN YACHTS WITH PRICE GT 45000 DTR> PRINT FIRST 2 CURRENT WITH BEAM = 13

DTR> PRINT FAMILIES WITH ANY KIDS WITH AGE > 30

Keywords

:Command

Invokes a procedure.

:procedure-name

DTR> :SALARY_REPORT

$@ \operatorname{\mathbf{Command}} \\$

Invokes a command file.

@file-spec

DTR> @DB0:E200,2003REPORT.CMD

ABORT Statement

Stops execution of a statement, procedure, or command file.

ABORT [value-expression]

DTR> ABORT "Data entry terminated."

ADT Command

Invokes the Application Design Tool.

ADT

ALLOCATION Clause

Specifies the type of word boundary alignment used in a data file. Part of the DEFINE RECORD command.

| ALLOCATION IS | MAJOR_MINOR ALIGNED_MAJOR_MINOR LEFT_RIGHT |
|---------------|--|
|---------------|--|

fields

MAJOR_MINOR: Word boundary alignment according to data types for elementary fields defined with the SYNC clause

ALIGNED_MAJOR_MINOR:Word boundary alignment according to data types for all elementary fieldsLEFT_RIGHT:Word boundary alignment for COMP, COMP_1, COMP_2, and DATE elementary

Assignment Statement

Assigns a value to an elementary field, group field, or variable.

Elementary Field:

field-name = value-expression

DTR> STORE YACHTS USING BUILDER = *. "MANUFACTURER"

Group Field:

```
group-field-name-1 = group-field-name-2
```

DTR> FOR YACHTS STORE NEW_YACHTS USING NEW_TYPE = TYPE

Variable:

```
variable-name = value-expression
```

DTR> DECLARE T USAGE DATE.; T = "15-May-1983"

BEGIN-END Statement

Groups statements into a single compound statement.

BEGIN

statement-1 [statement-2]

END

.

DTR> FOR YACHTS WITH PRICE EQ O MODIFY USING CON> BEGIN CON> PRINT; PRICE = *."price"; PRINT CON> END

CLOSE Command

Closes an RMS trace file created with the OPEN command.

CLOSE

COMPUTED BY Clause

Describes a virtual field. The value is calculated when the field is used in a statement.

COMPUTED BY value-expression

DTR> DECLARE OVERHEAD COMPUTED BY CON> TOTAL SALARY OF PERSONNEL WITH DEPT = "C82".

DECLARE Statement

Defines a global or local variable.

DECLARE variable-name variable-definition.

DTR> DECLARE X USAGE REAL EDIT_STRING IS ZZZ99V9(5).

DECLARE PORT Statement

.

;

Creates a temporary port and readies it for write access. Used to pass values between a calling program and DATATRIEVE.

DECLARE PORT port-name USING

level-number-1 field-definition-1.

[level-number-2 field-definition-2.]

[level-number-n field definition-n.]

DECLARE PORT PT1 USING 01 NUM PIC 9(4) USAGE COMP.;

DEFINE DICTIONARY Command

Creates a new DATATRIEVE-11 data dictionary file.

DEFINE DICTIONARY file-spec

DTR> DEFINE DICTIONARY DB0:[200;200]SADLER,DIC

DEFINE DOMAIN Command

Stores a domain definition in a DTR-11 data dictionary.

RMS Domain:

DEFINE DOMAIN domain-name USING record-name

ON file-spec;

DTR> DEFINE DOMAIN PERSONNEL USING PERSONNEL_REC ON DB0:[200;200]PERSON,DAT;

(passwd)

(*)

DEFINE DOMAIN Command (Cont.)

View Domain:

;

DEFINE DOMAIN view-name OF domain-name [,...] USING

level-number-1 field-name-1 OCCURS FOR rse-1 .

level-number-2 field-name-2

OCCURS FOR rse-2 FROM domain-name-2

| DTR> | DEFINE | DOMAIN S | SATI BOATS | DE Y | ACHTS . | NUNER | S UST | NG | |
|------|---------|----------|------------|-------|---------|--------|-------|--------|--------|
| DEN> | 01 SAIL | BOAT OC | CURS FOR | YACHT | S. | Uniten | 0 001 | | |
| DEN> | 05 | BOAT FI | ROM YACHT | в. | | | | | |
| DEN> | 05 | SKIPPE | S OCCURS | FOR | OWNERS | WITH | TYPE | = BOAT | .TYPE. |
| DFN> | | 10 N | AME FROM | DWNER | 5.; | | | | |

DEFINE FILE Command

Creates an RMS sequential or indexed data file for an RMS domain.

Sequential file:

```
DEFINE FILE [FOR] domain-name [,]
```

```
\left[\begin{array}{c} ALLOCATION = n\\ SUPERSEDE\\ MAX \end{array}\right] \quad [,...]
```

Indexed sequential file:

DEFINE FILE [FOR] domain-name [,]

```
 \left[ \begin{array}{c} ALLOCATION = n \\ SUPERSEDE \\ MAX \end{array} \right] [,...] ( \\ \left\{ KEY = field-name-1 \ \left[ \ ( [NO] CHANGE [,] \ [NO] DUP ) \right] \right\} [,...] \right\}
```

DTR> DEFINE FILE FOR YACHTS KEY = TYPE (DUP), KEY = MODEL (CHANGE

DEFINE PORT Command

Enters the definition of a port into a DATATRIEVE-11 data dictionary.

DEFINE PORT port-name [USING] record-name ;

DTR> DEFINE PORT YPORT USING YACHT;

DEFINE PROCEDURE Command

Enters a procedure definition into a DATATRIEVE-11 data dictionary.

DEFINE PROCEDURE procedure-name

END_PROCEDURE

DTR> DEFINE PROCEDURE DEMO DFN> SET DICTIONARY SY:DEMO.DIC; READY YACHTS DFN> END_PROCEDURE

DEFINE RECORD Command

,

Enters a record definition into a DATATRIEVE-11 data dictionary.

DEFINE RECORD record-name [USING]

ALLOCATION IS

MAJOR-MINOR ALIGNED-MAJOR-MINOR LEFT-RIGHT

level-number-1 field-name-1 [field-definition-1] .

[level-number-2 field-name-2 field-definition-2 .]

```
DTR> DEFINE RECORD PHONE_REC USING
DFN> 01 PHONE.
DFN> 02 NAME PIC X(20).
DFN> 02 NUMBER PIC 9(7) EDIT_STRING IS XXX-XXXX.
DFN> 02 LOCATION PIC X(9).;
```

DEFINE TABLE Command

Enters a dictionary table in a DATATRIEVE-11 data dictionary.





```
DTR> DEFINE TABLE DEPT_TABLE
DFN> CE : "Commercial Ensineerins",
DFN> CS : "Customer Support",
DFN> SD : "Sales"
DFN> ELSE "Unknown Department"
DFN> END_TABLE
```

DEFINEP Command

Adds an entry to the access control list for a dictionary object. See Access Control Privileges for a list of privileges.



DELETE Command

Deletes dictionary objects from a DATATRIEVE-11 data dictionary.

DELETE object-name ;

DELETEP Command

Deletes an entry from the access control list of a DATATRIEVE-11 dictionary object.

DELETEP object-name sequence-number

DTR> DELETEP YACHTS 2

DISPLAY Statement

Displays the value of a value expression.

DISPLAY value-expression

DTR> FOR FIRST 5 PERSONNEL DISPLAY SALARY

DROP Statement

Removes a selected record from a collection.

DROP [collection-name]

DTR> FIND PERSONNEL WITH DEPT = "F11"; SELECT LAST DTR> DROP

EDIT Command

Invokes the DATATRIEVE Editor to edit the specified dictionary object.

EDIT object-name $\begin{bmatrix} (passwd) \\ (*) \end{bmatrix}$ [ADVANCED]

EDIT_STRING Clause

Specifies the output format of a field value. See Edit String Characters for full listing.

EDIT_STRING [IS] edit-string

DTR> DECLARE X PIC 9(7)V99 EDIT_STRING IS \$\$,\$\$\$,\$\$\$,99.

ERASE Statement

Permanently removes data records from an indexed sequential file or a relative data file.

ERASE [ALL [OF rse]]

DTR> ERASE ALL OF PERSONNEL WITH DEPT = "F11"

EXIT Command

Ends a DATATRIEVE session.

EXIT

EXTRACT Command

Copies the definitions of dictionary objects to the specified file. Default file extension is .CMD.

EXTRACT [ON] file-spec object-name [,...]

DTR> EXTRACT ON TEMP, CMD FAMILIES

FIND Statement

Establishes a collection of records from a domain, view, collection, or list.

FIND rse

DTR> FIND BIG_ONES IN YACHTS WITH LOA > 30 DTR> FIND FIRST 3 BIG_ONES

FINISH Command

Ends control over domains and releases collections associated with the domains. FINISH [domain-name] [....]

FOR Statement

Executes one or more statements once for each record in the record stream formed by an RSE.

FOR rse statement

DTR> FOR FIRST 2 FAMILIES CON> FOR KIDS CON> MODIFY USING AGE = AGE + 1
HELP Command

Provides on-line information about DATATRIEVE.

```
HELP [ADVANCED] [topic] [,...]
```

IF-THEN-ELSE Statement

Causes DATATRIEVE to execute one of two statements depending on the evaluation of a Boolean expression.

IF boolean-expression [THEN] statement-1 [ELSE statement-2]

```
DTR> FOR YACHTS WITH BUILDER = "PEARSON"
CON> BEGIN
CON> IF *."Y to modify price: N to skip" CONT "Y"
CON> THEN MODIFY PRICE ELSE PRINT "No chanse"
CON> IF *."Y to continue" NOT CONT "Y" THEN
CON> ABORT "END OF PRICE CHANGES"
CON> END
```

MODIFY Statement

Changes the values of fields in records.

MODIFY [ALL] [field-name [,...] USING statement-1] [VERIFY [USING] statement-2]

[OF rse]

DTR> FOR PERSONNEL WITH DEPT = "F11" MODIFY ALL USING DEPT = "F12"

OCCURS Clause

Defines multiple occurrences of a field or group field.

Fixed number of occurrences:

OCCURS n TIMES

```
03 KIDS_NAMES OCCURS 5 TIMES.
05 FIRST_NAME PIC X(10).
05 MIDDLE_INIT PIC X.
```

Variable number of occurrences:

OCCURS min TO max TIMES DEPENDING ON field-name

03 NUMBER_KIDS PIC 99 EDIT_STRING Z9, 03 KIDS OCCURS 0 TO 10 TIMES DEPENDING ON NUMBER_KIDS, 06 EACH_KID, 09 KID_NAME PIC X(10) QUERY_NAME IS KID,

OPEN Command

Opens an RMS file to log an interactive dialogue with DATATRIEVE.

OPEN file-spec

PICTURE Clause

Specifies the format of a field value as it is stored.

PIC[TURE] [IS] picture-string

| Picture String Characters | | |
|---------------------------|---|--|
| Character | Description | |
| Х | One character | |
| 9 | One digit; 1 to 18 digits for a numeric field | |
| S | Sign (+ or -); only one S leftmost | |
| V | Implied decimal point | |
| Р | Decimal scaling position; "distance" in digits from implied decimal point | |

PRINT Statement

Writes the specified value expressions to a terminal, file, or device.

Retrieving from selected records and target record streams formed by FOR loops:

 PRINT [print-list]
 ON
 {file-spec *.prompt }

Retrieving from the current collection:

Retrieving from record streams formed by the PRINT statement:

1. One RSE PRINT [print-list OF] rse $\left[\begin{array}{c} ON & \left\{ \begin{array}{c} file-spec \\ *.prompt \end{array} \right\} \right]$

2. Two RSE's (inner print list follows another print list)

PRINT print-list, ALL print-list OF rse-1 [,print-list] OF rse-2

ON { file-spec }

3. Two RSEs (Inner print list precedes any other print list)

PRINT ALL ALL print-list OF rse-1 [,print-list] OF rse-2

DTR> PRINT BUILDER ("Vendor"), LOA (-) OF FIRST 10 YACHTS ON REPORT.LIS

| Print List Elements and Modifiers | | |
|--|--|--|
| Print-list Element | Description | |
| field-name [modifier] group-field-name list-field-name [modifier] | Specifies the field whose contents are to be output. | |
| literal [modifier] *.prompt-name [modifier] **.prompt-name [modifier] arithmetic-exp [modifier] statistical-exp [modifier] | Specifies a value expression to be evaluated and output. | |
| SPACE [n] | Inserts n horizontal spaces before next print list element. | |
| TAB [n] | Inserts space of n tab characters before next print list element. | |
| COL n | Specifies column n as start on next print list element. | |
| SKIP[n] | Begins output of next print list element at start of nth line from current line. | |
| NEW_PAGE | Moves cursor to top of a new print page. Column headers suppressed. | |
| ALL printlist OF rse | Specifies an inner print list evaluated once for each record specified by outer RSE. | |

| Print List Modifier | Description |
|-----------------------|---|
| ("header-segment"[/]) | Specifies character string literal displayed as column header above first line of output. |
| (-) | Suppresses display of query header. |
| USING edit-string | Specifies edit string for preceding field or value expression. |

QUERY_HEADER Clause

Specifies the column header used when a field value is displayed by the PRINT statement.

```
QUERY_HEADER [IS] {"header-segment"} [/...]
```

```
OG DISPLACEMENT PIC 9(5)
QUERY_HEADER IS "WEIGHT"/"OF"/"BOAT".
```

QUERY_NAME Clause

Specifies an alternate name for a field.

QUERY_NAME [IS] query-name

03 SPECIFICATIONS QUERY_NAME IS SPECS.

READY Command



REDEFINES Clause

Provides an alternate field definition.

level-no field-name-1 REDEFINES field-name-2

```
05 PART_NUMBER PIC 9(10),
05 PART_NUMBER_PARTS REDEFINES PART_NUMBER,
07 PRODUCT_GROUP PIC 99,
07 PRODUCT_YEAR PIC 99,
07 ASSEMBLY_CODE PIC 9(6).
```

RELEASE Command

Ends control over collections, tables, and variables, freeing the workspace they occupy.

RELEASE

collection-name table-name [,...] variable-name

REPEAT Statement

Executes a statement a specified number of times.

REPEAT value-expression statement

DTR> REPEAT 5 DTR> BEGIN DTR> STORE PERSONNEL DTR> PRINT "There are now "{COUNT OF PERSONNEL!!"employees." DTR> END

REPORT Statement

Invokes the Report Writer and is the first statement in a report specification.

REPORT [rse] ON { file-spec *.prompt }

Statements used in a report specification: Naming the report:

SET REPORT_NAME = { "string"[/...] *.prompt

Specifying date or string at upper right of each page:

SET DATE = "string"

Controlling display of date or page number:

REPORT Statement (Cont.)

Specifying page width or length and report length:

SET
$$\begin{cases} COLUMNS_PAGE = \\ LINES_PAGE = \\ MAX_LINES = \\ MAX_PAGES = \end{cases} \end{cases} \begin{cases} n \\ *.prompt \end{cases}$$
 [....]

Printing detail lines:

```
PRINT print-list-element [,...]
```

```
Displaying header and summary lines:
```

AT BOTTOM OF REPORT PAGE field-name PRINT summary-element [,...]

Ending the report:

END_REPORT

| Report Writer Header and Summary Elements | | |
|---|--|--|
| Element | Description | |
| AVERAGE value-expression | Displays average of value expressions in record stream or collection. | |
| COL n | Specifies position of next output element. | |
| COLUMN_HEADER | Displays column headers specified in PRINT and AT statements. | |
| COUNT | Displays number of records in record stream or collection. | |
| field-name [modifier] | In AT TOP OF field-name, prints common field value at top of control groups. | |
| MAX value-expression | Displays maximum value of value expressions in record stream or collection. | |
| MIN [value-expression] | Displays minimum value of value expressions in record stream or collection. | |
| NEW_PAGE | Starts new page. | |
| NEW_SECTION | Starts new page and new section numbered page 1. | |
| REPORT_HEADER | Displays report header. | |
| SKIP[n] | Displays next header or summary element n lines from current line. | |

| SPACE [n] | Inserts n spaces. |
|------------------------|---|
| TAB [n] | Inserts space of n tabs. |
| TOTAL value-expression | Displays total of value expressions in record stream or collection. |
| value expression | Displays value in header or summary line. |

SELECT Statement

Establishes a target record in a collection.

SELECT FIRST NEXT LAST value-expression [collection-name]

DTR> FIND YACHTS; SELECT LAST; MODIFY PRICE

SET Command

Enables or disables the specified options. Specifies default dictionary.

SET set-option [,...]

| Option | Default | Purpose |
|---------------------|--|---|
| [NO] ABORT | NO ABORT | Controls whether procedures and command files are aborted on ABORT statement. |
| COLUMNS_PAGE = n | la har a ser a | Establishes number of columns per page for output. |
| DICTIONARY dic-name | | Sets current DATATRIEVE–11 data dictionary. |
| GUIDE | | Starts Guide Mode. |
| [NO] PROMPT | PROMPT | Controls syntax prompting. |

SHOW Command

Displays information about DATATRIEVE-11 data dictionary and its contents; shows status of various set options.

| SHOW | ALL collection-name COLLECTIONS CURRENT DICTIONARY DOMAINS FIELDS [FOR domain-name] object-name PROCEDURES READY RECORDS SPACE | | [,,- |
|------|---|---|------|
| | TABLES | / | |

[,...]

SHOWP Command

Displays the access control list for a dictionary object.

SHOWP object-name (passwd)

SIGN Clause

Specifies the location of a sign in a numeric elementary clause for a program that uses the record. Does not affect output format of field

SIGN [IS] LEADING [SEPARATE]

03 NEW_PRICE PIC S999V99 SIGN TRAILING SEPARATE.

SORT Statement

Arranges a collection in the order specified.

SORT [collection-name] [BY] sort-key-1 [,...]

DTR> FIND FIRST 10 YACHTS DTR> SORT BY ASCENDING LOA, DESC PRICE

STORE Statement

Creates a record in a domain and stores values in fields of the record.

```
STORE [context-variable IN] domain-name
```

```
[USING statement-1]
```

```
[VERIFY [USING] statement-2]
```

```
DTR> STORE A IN PERSONNEL USING

CON> BEGIN

CON> ID = *."ID"

CON> LAST_NAME = *."last name"

CON> END VERIFY USING

CON> BEGIN

CON> PRINT A.PERSON, SKIP

CON> IF *."confirmation (Y or N)" CONT "N" THEN

CON> PRINT SKIP THEN ABORT "Bad record"

CON> END
```

```
      SUM Statement

      Summarizes totals for fields in a collection.

      SUM print-list BY sort-list

      ON
      {file-spec 
*.prompt}
```

```
DTR> FIND YACHTS
DTR> SUM 1 ("NUMBER"/"OF YACHTS"), PRICE USING $$,$$$ BY DESC RIG
```

THEN Statement

```
Joins multiple statements into a compound statement.
```

```
statement-1 {THEN statement-2} [...]
```

```
DTR> FOR YACHTS WITH BUILDER EQ "ALBIN"
CON> PRINT THEN MODIFY
```

USAGE Clause

Specifies internal format of a numeric field or specifies a date.

| | (COMP) (INTEGER) |
|------------|---------------------|
| | (COMP-1) (REAL) |
| USAGE [IS] | COMP-2 |
| | COMP-3 PACKED |
| | COMP-5 ZONED |
| | DATE |

| Usage Types | | |
|------------------|--|--|
| Туре | Description | |
| DISPLAY | Each digit occupies one byte. Default usage. | |
| COMP INTEGER | Binary format. Size depends on number of fields specified in PICTURE clause. | |
| COMP_1 REAL | Single-precision real format. | |
| COMP_2 DOUBLE | Double-precision real format. | |
| COMP_3 PACKED | Packed decimal format. | |
| COMP_5 ZONED | Signed decimal format. | |
| DATE | Date field. | |

VALID IF Clause

Validates a field value before storage.

VALID IF boolean-expression

```
OG RIG PIC X(6)
VALID IF RIG EQ "SLOOP", "KETCH", "MS", "YAWL".
```

WHILE Statement

Repeats a statement as long as the condition specified in the Boolean expression is "true."

WHILE boolean-expression statement

```
DTR> DECLARE X PIC 99.; X = 0
DTR> FOR YACHTS WITH LOA < 35 AND BEAM NE 0 SORTED BY BEAM, LOA
CON> WHILE X < BEAM
CON> BEGIN
CON> PRINT TYPE, LOA, BEAM
CON> X = BEAM
CON> END
```

Access Control Privileges

| Privilege | Meaning |
|-----------|---|
| R | Read. User can SHOW or EXTRACT the associated dictionary object. For a domain, user can ready the domain for READ access only. |
| w | Write. User can ready the domain for READ, EXTEND, MODIFY, or WRITE access to retrieve, modify, store, or erase records. |
| Е | Extend or Execute. For a domain, user can ready the domain for EXTEND access only to store records. For a procedure, user can execute the procedure. For a table, user can refer to the table using VIA or IN. You must have E access to a record to ready the associated domain. |
| М | Modify. User can ready the domain for READ or MODIFY access to read or change records in the domain, but not to add or delete. |
| С | Control. User can issue the commands DEFINEP, DELETE, DELETEP, EDIT, and SHOWP. |
| Space(s) | No access. User cannot access the dictionary object. |

Calls

For all calls, dab is the DATATRIEVE Access Block, passed by reference. In BASIC and COBOL, strings are passed by descriptor; in FORTRAN, strings are passed by an address and length.

DTCMD (dab, <command-str> [, <arg-str> ...])

Passes a command line to DATATRIEVE.

command-str

A DATATRIEVE command string passed by descriptor. May contain !CMD substitution directive.

arg-str

A substitution string passed by descriptor.

DTCONT (dab)

Causes DATATRIEVE to continue execution until it enters the next appropriate state.

DTFINI (dab)

Ends a program's interaction with DATATRIEVE.

DTGETP (dab, <record-buf>, record-len)

Retrieves a record from a port and passes it to a program.

record-buf

The buffer that contains the port record. Passed by descriptor.

record-len

Length of the record in the record-buffer. Passed by reference.

DTINIT (db, str-len, buff-len, <node>, options)

Initializes the DATATRIEVE Call Interface.

str-len

The length of the string buffer. Passed by reference.

buff-len

The length of the internal buffer DAB\$V_BUFFER in bytes. Passed by reference.

node

The DECnet node specification that your program uses. Passed by descriptor.

options

Options value. Passed by reference.

| DTINIT OPTIONS | |
|----------------|---|
| Options = | Meaning |
| 0 | Semicolons required. DATATRIEVE banner is not displayed. |
| 1 | Semicolons not required. DATATRIEVE banner is not displayed. |
| 2 | Semicolons required. DATATRIEVE banner is displayed. |
| 3 | Semicolons not required. DATATRIEVE banner is displayed. |

DTLINE (dab, <pline-buf>, pline-len)

Gets a print line from DATATRIEVE and places it in a buffer.

pline-buf

A buffer that contains the print line text. Passed by descriptor.

pline-len

The length of the print line without trailing spaces.

DTMSG (dab, <msg-buf>, msg-len)

Gets the text of a DATATRIEVE message.

msg-buff

A buffer that contains the message. Passed by descriptor.

msg-len

The length of the message text. Passed by reference.

DTPEOF (dab)

Sends an end-of-file marker to DATATRIEVE, terminating the passing of records.

DTPUTP (dab, <record-buf>)

Passes a record from a port to DATATRIEVE.

record-buf

The record buffer that stores the record to be passed to DATATRIEVE. Passed by descriptor.

DTPVAL (dab, <value>)

Passes a value from a program to DATATRIEVE.

value

An ASCII string specifying the value to be passed in response to the DATATRIEVE prompt. Passed by descriptor.

DTUNWD (dab)

Stops execution of a command passed to DATATRIEVE.



DIGITAL EQUIPMENT CORPORATION

Printed in U.S.A.