

An overview of MultiValue commands

Contents

Command Prompt	3
Querying files and records	3
Operators	4
Select Lists	4
TCL Commands	4
Command Stack	5
Editing Records	6
Creating a file on UniVerse	7
Creating a file on UniData	8
Maintaining a file	8
Creating a dynamic file	8
Copying a file	8
Dictionaries: A description of common fields for a dictionary record	9
Creating a data (D-type) field	9
Creating an interactive (I-type) field	9
Creating a phrase field	9
Creating a paragraph (script)	10
Creating a keyword (parameter/modifier applied to verbs)	10
Creating a file pointer	10
Creating a qpointer	10
Creating a synonym	10
Editor Macros: Creating a new AE macro	11
Indexing Fields	11

Command Prompt

Action	Command
Change to a different account	LOGTO accountname
Quitting the session	QUIT
Logging off a session	LO
Breaking out into a shell in Linux	SH
Running command processor in Windows	DOS
Start online help	HELP commandname
Echo a string out to screen	DISPLAY string

Querying files and records

Action	Command
A basic list	LIST filename
Listing fields	LIST filename field1 field2 field3
Sorting by fields	LIST filename BY field3 field1 field2
	LIST filename BY field2 BY field3 field1
	LIST filename BY.DSND field3 field1 field2
	SORT filename BY field1
Sampling a list	LIST filename SAMPLE n
Listing with criteria	LIST filename WITH field3 = "data" field1 field2
Totaling a field	LIST filename field1 field2 TOTAL field3
Setting a breakpoint	LIST filename BREAK.ON field1 field2 field3
Summarizing a report	LIST filename BREAK.ON field1 field2 DET.SUP
Providing headers and footers	LIST filename field1 field2 field3 HEADING "report heading" FOOTING "report footing"
Overriding existing dictionary definitions	LIST filename field1 FMT "override format"
	LIST filename field1 CONV "new conversion"
	LIST filename field1 COL.HDG "new heading"
	LIST filename field1 SINGLE.VALUE
	LIST filename field1 MULTI.VALUE
Columns with EVAL	LIST filename EVAL "formula"

Operators

Criteria can be reversed with NOT and concatenated with AND or OR.

Symbol	Synonyms	Description
=	EQ	Equal to
#	<>, >; <, NE	Not equal to
>	GT, AFTER	Greater than
<	LT, BEFORE	Lesser than
>=	=>, GE	Greater than or equal to
<=	=<, LE	Lesser than or equal to
LIKE	MATCHES, MATCHING	Finds a matching word (... is a wildcard)
UNLIKE	NOT.MATCHING	Finds not matching word (... is a wildcard)
SPOKEN	SAID, ~	Phonetic matching

Select Lists

Action	Command
Creating a select list	SELECT filename
Numbered select lists	SELECT filename TO n
Saving a select list	SELECT filename
	SAVE.LIST listname FROM n
Getting a saved list	GET.LIST listname TO n
Clearing a list	CLEARSELECT n

TCL Commands

Action	Command
Count the records in a file	COUNT filename
Copy a record to the terminal (display a record)	CT filename record
Copy a record to the printer	CP filename record
Copy records to another file	COPY FROM sourcefile TO destfile record1 record2 record3
	COPY FROM sourcefile TO destfile ALL
Change the name of a file	CNAME oldname TO newname
Change the key of a record in a file	CNAME filename oldrecord, newrecord
Swap the contents of two records	EXCHANGE filename record1 record2
Delete records from a file	DELETE filename record1 record2
Remove all records from a file	CLEAR.FILE filename
Clear the terminal	CS

Command Stack

All the items with an “n” in their execution can either be removed (to operate on the most recent item) or replaced with a number indicating the item in the stack the operation is to effect.

Action	Command
Viewing your command history	.L
Executing a command from the stack	.Xn
Delete an item from the stack	.Dn
Recall an older command to the head of the stack	.Rn
Change a string in a command on the stack	.Cn/oldstring/newstring
Change all occurrences of a string in a command on the stack	.Cn/oldstring/newstring/G
Appending something to the end of a command on the stack	.An extra
Save commands to a sentence VOC	.S name from_line to_line

Editing Records

Action	Command
Invoking the Editor	AE filename recordkey
Display a pageful of the record	P
Quitting a record	Q
Inserting lines	I
Moving between lines	n (move to line n)
	+n (move forward n lines)
	-n (move backward n lines)
	t (move to top)
	b (move to bottom)
Changing lines (/G for global)	C/oldstring/newstring/G
Undo	00PS
Appending to a line	A extra-string
Breaking a line	B word-to-break-on
Concatenating lines	CAT
Deleting lines	D
Duplicating lines	DUP
Locating and finding	L something
	F something
Setting editing blocks	< (sets the starting block)
	> (sets the ending block)
Copy, move or drop blocks	COPY
	MOVE
	DROP
Saving your record	SAVE
Run a &AE& macro	.X macroname

Creating a file on UniVerse

CREATE.FILE filename filetype other

Filename is used to refer to the data file. There are few limitations to the name, do not include spaces.

File type	Type	Description
1 & 19	directory	
2	static	Keys end with numbers.
3	static	Keys end mainly with numbers.
4	static	Keys end with letters.
5	static	Keys end with full range of ASCII characters.
6	static	Keys begin with numbers.
7	static	Keys begin mainly with numbers.
8	static	Keys begin with letters.
9	static	Keys begin with full range of ASCII characters.
10	static	Keys are numbers.
11	static	Keys are mainly numbers.
12	static	Keys are letters.
13	static	Keys are full range of ASCII characters.
14	static	Entire keys are numbers.
15	static	Entire keys are mainly numbers.
16	static	Entire keys are letters.
17	static	Entire keys are full range of ASCII characters.
18	static	Entire keys are of arbitrary form.
30	dynamic	



Creating a file on UniData

```
CREATE.FILE filename modulo[,separation] TYPE hashtype
```

Maintaining a file

```
RESIZE filename newtype newmodulo newseparation
```

Creating a dynamic file

```
CREATE.FILE filename 30
```

Copying a file

```
COPY FROM sourcefile TO destfile ALL OVERWRITING
```


Dictionaries: A description of common fields for a dictionary record

Field	Description
Conversion	Blank unless a conversion is required. e.g. D DMY[2,A3,4] would store 1 for 01 JAN 1968
Column Header	Title that appears at the head of the column
Format	The number of characters to display and alignment. 10R - 10 characters, right aligned. 10T - 10 characters, text aligned. 10L - 10 characters, left aligned.
Single or Multi	S for single value, M for multi-value

Creating a data (D-type) field

```
> AE DICT filename fieldname
----: I
0001= D Description of the field
0002= field number
0003= Conversion
0004= Column Header
0005= Format
0006= Single or Multi
```

Creating an interactive (I-type) field

```
> AE DICT filename fieldname
----: I
0001= I Description of the field
0002= formula
0003= Conversion
0004= Column Header
0005= Format
0006= Single or Multi
```

Creating a phrase field

```
> AE DICT filename fieldname
----: I
0001= PH Description of the field
0002= fieldnames
```

Creating a paragraph (script)

Paragraphs start with PA, then continue to define the script itself. For variables that are unknown at runtime the «...» syntax can be used to prompt the user inline for the value.

```
> AE VOC paragraphname
----: I
0001= PA
0002= script ....
0003= script ....
```

Creating a keyword (parameter/modifier applied to verbs)

Keywords start with a K. Their second parameter is the value that gets substituted for the word.

```
> AE VOC keywordname
----: I
0001= K
0002= value
0003= verbcommand (optional)
```

Creating a file pointer

```
> AE VOC filename
----: I
0001= F
0002= directoryname
0003= filename
```

Creating a qpointer

```
> AE VOC synonymname
----: I
0001= Q
0002= account OR (space for current account)
0003= filename
```

Creating a synonym

```
> AE VOC remotecommandname
----: I
0001= R
0002= filename that contains the command record
0003= key of the command record
```

Editor Macros: Creating a new AE macro

These macros can be run from the AE editor with **.X macroname.**

@FILE variable can be used to be the existing file being edited. **@ID** variable can be used to be the primary key of the current record being edited. Example usage of **@FILE** and **@ID** within a macro which could then be run from AE with **.X SBR**

> AE &AE& SBR (save, BASIC and RUN)

----: I

0001= E

0002= SAVE

0003= XEQ BASIC @FILE @ID

0004= XEQ RUN @FILE @ID

0005=

Indexing Fields

Action	Command
Creating an index	CREATE.INDEX filename fieldname
Bringing an index up to date	BUILD.INDEX filename fieldname
Checking indexes on a file	LIST.INDEX filename fieldname/ALL
Removing an index from a file	DELETE.INDEX filename fieldname