

FLEX INQUIRY

Reference Manual

Relative to version 610.04

Contents

| | |
|---|-----------|
| Recent Changes | 5 |
| March 2019 600.03 Add option suffix for loads..... | 5 |
| Oct 2015 580.02 Enhanced Log command | 5 |
| Feb 2015 580.01 Allow for page numbers bigger than 999 | 5 |
| Jul 2013 560.07 :ACT,BR | 5 |
| May 2013 560.05 Add 'other reels' column to LISTVL HTML report..... | 5 |
| Feb 2013 560.02 Respect granulated privileges | 6 |
| Feb 2013 560.01 Correct Usage layout problem on large screens | 6 |
| Nov 2011 540.42 Find and DFH | 6 |
| Oct 2011 540.39 Search Sorts..... | 6 |
| Jul 2011 540.37 Resolve RULES confusion | 7 |
| Mar 2011 540.32 Implement FLEX logging..... | 7 |
| Jan 2011 540.28 Inhibit batch compile listings for :PRINT runs..... | 7 |
| Jul 2010 540.23 Zip Commands..... | 7 |
| Jul 2010 540.22 Deimp :NOZIP | 9 |
| Apr 2010 540.17 Remove options TTY and TD830 | 9 |
| Feb 2010 540.13 Allow a DO <opal string> in a DO file | 9 |
| Feb 2010 540.05 New WHERE modifier | 9 |
| Jan 2010 540.03 Support new Myself(INPUTPARAM) attribute..... | 9 |
| Introduction..... | 10 |
| Getting Started - The Guided Tour | 12 |
| Setting options for your terminal | 22 |
| Further Tips..... | 22 |
| Filenames, Familynames, Directories, Patterns, and Wildcards | 24 |
| Familynames..... | 24 |
| Hostnames..... | 24 |
| Usercode..... | 24 |
| Filenames | 25 |
| Directories..... | 25 |
| Patterns and wildcards..... | 26 |
| Usercodes, search order, and the <pattern prefix> | 29 |
| Using Quotes in FLEX | 30 |
| Using quotes in OPAL expressions..... | 30 |
| Using quotes in <filename>s and <pattern>s | 30 |
| Commands | 33 |
| Command Groupings..... | 33 |
| Description of the command groupings | 33 |
| Command Summary | 34 |

| | |
|--|----|
| Command Descriptions | 36 |
| ACCESS command | 36 |
| BACKUp command | 37 |
| Options | 37 |
| Examples | 38 |
| BRK command | 38 |
| DEFault command | 38 |
| DO command | 40 |
| Parameters | 40 |
| Multi line commands | 41 |
| Typical Use | 41 |
| Examples | 41 |
| Opal Compiler | 42 |
| FAMily command | 43 |
| Files command | 44 |
| Options | 45 |
| Output | 45 |
| Examples | 46 |
| FIND command | 47 |
| <find target> | 48 |
| FORGet command | 48 |
| HELP command | 49 |
| INSTALL command (Full Flex only) | 49 |
| LC command | 50 |
| LFiles command | 50 |
| List command | 52 |
| LISTVL command | 53 |
| Variants | 54 |
| LOAD command | 54 |
| From Source | 55 |
| Via Catalog or Archive | 55 |
| Load TO | 56 |
| Non zero generations | 56 |
| Load Via | 56 |
| Compare two generations | 57 |
| Lost Family Index | 57 |
| Examples | 57 |
| LOG Command | 57 |
| <DateTimeRange> | 59 |
| <Units> | 59 |
| MZIP Commands | 60 |
| MAINT command | 61 |
| NExt command | 62 |
| ODT command | 62 |
| OPtion command | 63 |
| Available options & descriptions | 64 |
| Option groupings | 65 |
| Alphabetical listing of options & descriptions | 66 |
| <ACTion option> | 66 |
| <APpend option> | 66 |
| <ARChive option> | 66 |
| <ATT option> | 67 |
| <AUDit option> | 67 |
| <AUTOrm option> | 67 |
| <BRief option> | 67 |
| <Byte option> | 67 |
| <CANDE option> | 68 |
| <CATalog option> | 68 |
| <COPYWrite option> | 68 |
| <DEBUG option> | 68 |

| | |
|---|------------|
| <FAst option> | 68 |
| <File option> | 69 |
| <FINd option> | 69 |
| <FLat option> | 70 |
| <INTER option> | 70 |
| <LEVEL option> | 70 |
| <LINear option> | 71 |
| <MEMSZ option> | 71 |
| <MONitor option> | 71 |
| <NOMSGs option> | 71 |
| <NR option> | 71 |
| <Print option> | 72 |
| <REdirect option> | 72 |
| <REO option> | 73 |
| <Resident option> | 73 |
| <SAve option> | 74 |
| <SETCode option> | 74 |
| <SETList option> | 74 |
| <SYNtax option> | 74 |
| <TERMinal option> | 74 |
| <TRuncate option> | 74 |
| <TWobackups option> | 74 |
| <UNSEquence option> | 75 |
| <UNSOrt option> | 75 |
| <USdates option> | 75 |
| <WAIT option> | 75 |
| <Xlate option> | 75 |
| <ZIPMORE option> | 75 |
| The ACTION option | 76 |
| The FLAT and LINEAR options | 78 |
| PDT command | 79 |
| <device> ::= MT CD PK | 79 |
| PURGE command | 80 |
| PV command (Cataloging Only) | 81 |
| Quit command | 82 |
| RELease command | 83 |
| REMOve command | 84 |
| REPOrt command | 85 |
| REPORT SORT | 87 |
| REPORT DO | 87 |
| <opal string> | 88 |
| <opal block> | 89 |
| Practical Usage | 91 |
| Complex examples | 91 |
| RULEs command (Full Flex Only) | 93 |
| SAMe command | 93 |
| SAVE command | 93 |
| SECure command | 94 |
| SELection command | 95 |
| SPLit command | 96 |
| STart command | 96 |
| STD command | 96 |
| USAGE command | 98 |
| USER command | 99 |
| WFL command | 100 |
| WHAT command | 100 |
| Write command | 101 |
| DESTROY command | 101 |
| Cataloging | 103 |
| Extra features for cataloging users | 103 |
| LOAD | 103 |
| FILES :LINEAR | 103 |

| | |
|--|------------|
| "UN-PURGE" or oops! function | 103 |
| PDT BU MT <serialno list>..... | 104 |
| DELETE | 104 |
| RELEASE & DESTROY..... | 104 |
| LISTVL | 104 |
| MAINT & Diagnostic functions | 104 |
| OPAL | 106 |
| An Overview of the OPAL Language | 106 |
| SUMMARY of some OPAL Language components | 107 |
| FILECOPY | 109 |
| Replacing FILECOPY and Snap Backups | 109 |
| Copyright..... | 112 |
| ACKNOWLEDGEMENTS | 112 |

Recent Changes

March 2019 600.03 Add option suffix for loads

A new option has been added to improve the loading of old generations.

LOAD <filename> GENERATION= <n> :Suffix will load the requested file adding a suffix of the form yyyyymmdd_hhmmss_SV.

yyyyymmdd_hhmmss is formed from the timestamp of the file.

Since LOAD VIA <filename> does not allow the specification of generation, The USERINFO of the file is used to specify generation.

Ex

```
WFL MODIFY LOAD/LIST(USERINFO=6) followed by  
LOAD VIA LOAD/LIST:SUFFIX
```

Would load generation 6 of each file in LOAD/LIST adding a timestamp based suffix

Oct 2015 580.02 Enhanced Log command

If the response to a LOG command is truncated, due to the record limit, the record limit is now reported along with the date and time of the last record scanned.

A new 'time spec' option MORE has been added which allows a new log command to continue from where the previous one stopped.

Ex.

```
LOG MSG + will stop after 500 entries have been found.  
LOG MSG MORE will continue from the minute where the previous  
command stopped.
```

Feb 2015 580.01 Allow for page numbers bigger than 999

Page numbers on printed reports which were show in 3 digit scientific notation.

Pages 1000 to 14999 were show as 1E3. Large page numbers are now reported correctly.

Jul 2013 560.07 :ACT,BR

:ACT,BR puts an action field in front of the file in a normal files report, hence allowing more than twice the files per screen.

May 2013 560.05 Add 'other reels' column to LISTVL HTML report

Adds an 'Other Reels' column to the LISTVL HTML report which reports the serial

numbers of all reels of a multi reel volume.

Feb 2013 560.02 Respect granulated privileges

If Flex is run from a usercode with GSDIRECTORY privilege then :Flat , :Fast and :Find are allowed on any directory.

If the READ privilege is set then any file may be listed.

If The REMOVE privilege is set then any file may be removed.

If the CHANGESEC privilege is set then any file may have its security changed.

Feb 2013 560.01 Correct Usage layout problem on large screens

The usage report was badly formatted on screens with more than 38 lines. Usage now reports megabytes (1,000,000 bytes) instead of segments and uses thousands separators. This change also contains some preparatory code to support an upcoming feature to use granulated privilege.

Nov 2011 540.42 Find and DFH

When a FIND uses FLAT, the UNSORTED option is set, to avoid sorting large disk file headers. The results of the FIND can be sorted using the SORT modifier or REPORT SORT.

Oct 2011 540.39 Search Sorts

The results from a directory search may now be sorted using an OPAL report. This can be specified in-line using a SORT modifier or can be specified like a header using the new variant command, REPORT SORT.

The string segments in the report are truncated to 128 chars if no length is explicitly given. The total length is truncated to 512 chars. The report is executed once per eligible file and the result is used as a string key. That key determines the order the files are returned. E.g.

```
F =CAT= WHERE SEGMENTS > 99 SORT - SEGMENTS:FL REP SORT TITLEID  
64,,TIMESTAMP
```

The syntax of the SORT clause is:

```
SORT ----- <OPAL string> -----|
```

where - denotes sorting down. To turn off a REP SORT, use REP SORT OFF.

Note that if there is a WHERE clause in a command, it must precede the SORT clause.

Jul 2011 540.37 Resolve RULES confusion

To maintain rules for the system directory (i.e. no usercode) Flex uses the 'special' usercode SYSTEM-NODE.

Ex. RULES LIST FAM DEV USER SYSTEM-NODE.

The use of USER * was not specifically prevented but would never have the intended effect.

USER * is now a synonym for user SYSTEM-NODE in Rules commands.

Mar 2011 540.32 Implement FLEX logging

As with other Metalogic software, logging has been implemented for the FLEX suite. Most messages, errors, commands and warnings are now unconditionally logged into a file called *METALOGIC/FLEX/LOG and all FLEX software uses this system.

Logging is performed for all FLEX runs though only non-interactive batch (DO) runs are more complete. Errors and warnings for the file system in particular are now retained in these logs. Where files are removed, released or destroyed by a run of FLEX Utility then these entries will be logged too. It is likely that the FLEX content that is logged will be changed over the next few FLEX changes.

```
11:23:20 Job:Flex(27000):BOJ Flex UTILITY 54.999.32 running
11:22:17 Job:Flex(26990):EOJ Flex UTILITY run complete
11:22:15 Wf1:Flex(26990):RUN OBJECT/XFER/LIST/SORT
11:22:11 Wf1:Flex(26990):COPY XFER/LIST/SAVE AS XFER/LIST/OLD
11:22:11 Cmd:Flex(26990):DO XFER/FILECOPY
11:22:11 Msg:Flex(26990):User=IPP,Stn=174,Cmd=DO XFER/FILECOPY
11:22:11 Job:Flex(26990):BOJ Flex UTILITY 54.999.32 running
```

The LOG command syntax is identical to that provided for SUPERVISOR, MAILLIB and TRIM; full details on the implementation and syntax can be seen by using the HELP LOG command.

Jan 2011 540.28 Inhibit batch compile listings for :PRINT runs

Batch DO files that use the :PRINT output option and are executed from a non-remote origin will now not include any compiled script listings unless a script fails with syntax errors. Previously, the listings were unconditionally generated before the print output.

Jul 2010 540.23 Zip Commands

MZIP Commands

```
+-----<-----+-- , --+<-----+
```


Jul 2010 540.22 Deimp :NOZIP

The option NOZIP has been deimplemented. It had a small effect on certain jobs in the INSTALL module. These now act as if NOZIP is set. It was used in BACKUP to mean SYNTAX. Using :SYNTAX now has the same function. In LOAD, it was used as a debugging tool to save the LOADJOB generated. This can now be done with :SAVE.

Apr 2010 540.17 Remove options TTY and TD830

As far as we are aware flex will only be run on terminal emulators. The options used with old 'hard wired' terminals have been removed.

Flex is now more rigorous in checking for spurious input between a valid command and the : options.

Flex Help has been updated to include the WHERE clause.

Feb 2010 540.13 Allow a DO <opal string> in a DO file

A DO <file name> command is not permitted with a DO File. This restriction has been partly relaxed by a new feature allowing DO <opal string> within a DO file. The DO <opal string> is only permitted within a DO file. The expected use is to generate flex commands within a REPORT and then be able to execute them.

Feb 2010 540.05 New WHERE modifier

The WHERE modifier can be used to invoke an 'inline selection. The text following WHERE should be valid SELECT code. A WHERE clause will be used in preference to an active SELECT. The where clause can be used after a directory list and before any colon options. Since currently the SECURE command does not accept a directory list the WHERE clause cannot be used with SECURE. This will be addressed in a subsequent release.

```
Ex FILES TEST WHERE SYMBOLFILE FILES = WHERE AGE=0:Flat  
REMOVE = WHERE CODEFILE and AGE >365 :FL
```

Jan 2010 540.03 Support new Myself(INPUTPARAM) attribute

INPUTPARAM Returns the text of the most recent command.

Ex REP HEAD Myself(INPUTPARAM) Followed by FILES =BoB= ON DEV:FL
Would return FILES =BoB= ON DEV:FL followed by the list of files

Introduction

The FLEX Inquiry utility was written to allow fast and easy maintenance of files by the user or site manager. Most users of Unisys' Large Systems can recall many times when they wished that they could remove or list a file from a menu, or send a list of files to the printer, or select a group of files to be copied by certain file attributes. FLEX Inquiry is designed to allow complete maintenance of files within one program, while appearing to be an extension of CANDE or MARC. Some of its highlights are:

- It can be run as an interactive program from ET terminals or emulators, or from WFL.
- Uses the special features of Unisys screen terminals (TD830s, ET1100s and ET2000s, MT983s and SR100s) or compatible terminals or terminal emulators to great effect.
- It has a comprehensive and easy to use online help facility.
- Greatly improved version of CANDE's FILES, LFILES, SECURITY and REMOVE commands.
- File search methods that are *one thousand times* faster than standard searches
- Flexible and faster replacement for FILECOPY.
- Action mode, a menu-screen of files and their attributes, for browsing through directories, and acting on each file in a different way.
- Commands to print files or list them on a screen.
- Special commands to manage files on a cataloging system (LOAD, RELEASE,).
- Commands to manage the CATALOG and VOLUME LIBRARY on cataloging systems.
- Simple wildcard facilities similar to those found on microcomputers, including searching for files with specified content.
- The ultimate wildcard facility is provided through ERGO-like selection expressions. This feature is complemented by a report generator, which can build screens, disk files or printer reports.
- It is the interface to the full FLEX package and allows the user to define, test and refine FLEX rules.

This Manual is intended to complement the HELP feature of FLEX. As with any highly interactive program, the best way to learn about it is to use it. Instructions on how to get started are given in the next section. Next is a summary of the available commands, with the commands, their syntax, usage and examples presented in alphabetical order. Installation and special features are described. Note that the

various options available in FLEX inquiry are described under the OPTION command.

Getting Started - The Guided Tour

This section will guide you through a quick tour of some of the features of FLEX Inquiry. After working through this chapter, you will be in a good position to explore its other capabilities. We assume that your site manager has installed the package with the META/INSTALL program. If you are a CANDE user, you should log on to your terminal and enter:

U FLEX

If you are the first user to use FLEX since the last Halt Load or if the FLEX Library is not initialised, there will be a delay of one or two minutes while the library starts up. Normally, the library will be set up already and FLEX will announce itself after a short delay. To run FLEX from MARC, go to the "RUN" menu. Enter "OBJECT/FLEX" in the "Task name" field and " : INTER" in the "Parameter" field, then transmit.

After FLEX comes up, commands may be entered one at a time. FLEX will wait at page boundaries for your input telling it to continue, to break, or to process a new command. To terminate the program, enter the command "QUIT" or "END". On hardcopy or non-Unisys terminals, FLEX will behave as if the terminal were a teletype. Some of the more powerful features of FLEX are not available in this mode.

You will notice as you work with FLEX that the user interface resembles CANDE to a high degree. This is intentional as it allows some users to benefit from many of the features without training. Some features are logical extensions to CANDE or MARC; others give the user completely new capabilities.

We start this guided tour with a command that is familiar to all CANDE users — the FILES command. To see this command in FLEX Inquiry, run FLEX as detailed above, then enter "FILES" (or "FIL" for the lazy typist or F for the very lazy). This will show you a list of files in your directory:

```
DO/TEST : JOB & DIRECTORY (14)
DO/TEST/EXP : JOB (14)
IC/INSTALL : JOB & DIRECTORY (15)
IC/INSTALL/DO : SEQ & DIRECTORY (14)
IC/INSTALL/DO/OLD : SEQ (14)
IC/INSTALL/DO/SETUP : JOB (15)
IC/INSTALL/WFL/MAKE : JOB (30)
IC/INSTALL/WFL/SETUP : JOB (15)
IC/INSTALL/HELP : SEQ (14)
TEST/DATA : SEQ (14)
EDITOR/OPTIONS : DATA >10<
PENDING/META/INSTALL : DCALGOL & DIRECTORY (3010)
PENDING/META/INSTALL/PD/HYPER : DCALGOL (154)
PENDING/META/INSTALL/PD/REFIT : DCALGOL (42)
PENDING/META/INSTALL/PD/REORG : DCALGOL (3430)
PENDING/META/INSTALL/PD/EARLYMAGUS : DCALGOL (112)
PENDING/META/INSTALL/PD/NOWORKPACK : DCALGOL (854)
PENDING/META/INSTALL/NOTES : SEQ & DIRECTORY (324)
PENDING/META/INSTALL/NOTES/PD/REFIT : SEQ (14)
PENDING/META/INSTALL/NOTES/PD/REORG : SEQ (14)
PENDING/META/INSTALL/NOTES/PD/EARLYMAGUS : SEQ (14)
PENDING/META/INSTALL/NOTES/PATCHLIST : JOB (15)
PENDING/META/INSTALL/NEWTAPE : DCALGOL (3556)
7039      Push SPCFY for more, XMT to break or NEXT +/- 12:13:23
```

Some of the information is familiar from the CANDE FILES command, and some of it is new.

The full file title is shown with the Filekind reported after a colon character. Finally

the size of the file in 180 byte segments is reported, in () characters if the file is 'Crunched' and >< characters if not. See the line for EDITOR/OPTIONS above.

At the end of the first page, FLEX will be waiting for input. Hit the Specify key to go to the next page, or hit the Transmit key to break the output. Note that the input is interpreted so unlike CANDE you do not have to stop a FILE listing before entering the next command.

If the Specify key is used to step through the whole directory the last page will look like:

```
files
PENDING/META/INSTALL/PATCHLIST : JOB (15)
PENDING/META/INSTALL/SCREENINC/INSTALL : DCALGOL (1176)
PENDING/META/INSTALL/SCREENLIB : DCALGOL & DIRECTORY (4438)
PENDING/META/INSTALL/SCREENLIB/INSTALL : DCALGOL (4522)
PENDING/SYSDEF/META/INSTALL/SPECIFICATIONS : DATA >672<
PENDING/SYSDEF/CONVERTED/META/INSTALL/SPECIFICATIONS : ALGOL (1568)
PENDING/SUPERVISOR/NOTES/PATCHLIST : JOB (180)
PENDING/SUPERVISOR/PATCHLIST : JOB (90)
#Found 31 files [from 31 files]
#Directory: (TEST)= ON DEV
#
```

7039 Utility File Handler Version 53.530.104 12:21:29

Note that a summary is added after the last file and the original command has now been restored to line 1.

Most commands to FLEX can be modified by adding keywords at the end of a command, after a colon (":"). These keywords are called <colon option>s, and are very similar to <colon options> in CANDE. As you may have noticed, the FILES display above shows one file per line. This is done to speed up the display of your file directory. The <CANDE option> can be used with the FILES command to display files in the hierarchical format used by CANDE or MARC.

Try entering "FILES : CANDE" in FLEX. The display will now be something like:

```
. DO
. . TEST : JOB & DIRECTORY (14)
. . . EXP : JOB (14)
. IC
. . INSTALL : JOB & DIRECTORY (15)
. . . DO : SEQDATA & DIRECTORY (14)
. . . . OLD : SEQDATA (14)
. . . . SETUP : JOB (15)
. . . WFL
. . . . MAKE : JOB (30)
. . . . SETUP : JOB (15)
. . . HELP : SEQDATA (14)
. TEST
. . DATA : SEQDATA (14)
. EDITOR
. . OPTIONS : DATA >10<
. PENDING
. . META
. . . INSTALL : DCALGOL & DIRECTORY (3010)
. . . . PD
. . . . . HYPER : DCALGOL (154)
. . . . . REFIT : DCALGOL (42)
. . . . . REORG : DCALGOL (3430)
```

7039 Push SPCFY for more, XMT to break or NEXT +/-4 12:24:29

Note that this display does show more than CANDE or MARC — namely, segments,

crunched status, and resident status. In general, this is how FLEX works — the user interface is very much like CANDE or MARC, but the end result contains significant improvements in function.

Next, suppose that you would like to see the attributes of one of these files, for example **TEST/DATA**. In CANDE, you would enter "**LFILES TEST/DATA**". This would run SYSTEM/FILEDATA, and the information would be shown on the screen. In FLEX, a similar display is obtained by entering "**FILE TEST/DATA : ATT**".

This is shown below:

```
FILES TEST/DATA :ATT
TEST/DATA 1:0 SEQ (14)
Created 12:08:37,27/08/08 BST Security PRIVATE IO Extmode EBCDIC
Altered 12:08:37,27/08/08 BST FilStr ALIGNED180 EXP Rec/Blksz 14/420 W
Used 12:08:37,27/08/08 BST BlkStr FIXED Length 14 W
BackedUp ** No BACKUP ** LastRecord 0 Rows 1*504 S
#Found 1 file [from 1 file]
#Directory: TEST/DATA ON DEV
#
7039 Utility File Handler Version 53.530.104 12:28:53
```

This has been aligned for easier reading, but there is an even more comprehensive form that is given by the LFILES command.

Entering "**LFILES TEST/DATA**" yields:

```
LFILES TEST/DATA
TEST/DATA 1:0 SEQDATA (14 Segs 1 Recs 14+406 Wds)
TimeStamp 12:08:43,27/08/08 Extmode EBCDIC Security PRIVATE IO
Created 12:08:37,27/08/08 BST UserInfo 0 Owner TEST:RWX
Copied On 12:08:37,27/08/08 BST BlkStr FIXED Group <NONE>:NO
Att Mod. 12:08:43,27/08/08 BST Mode/FrameSz 8/48 b Other :NO
Altered 12:08:37,27/08/08 BST Recsz 14 W FilStr ALIGNED180 EXP
Used 12:08:37,27/08/08 BST Blksz 420 W FilOrg NOTRESTRICTED
Read 12:08:37,27/08/08 BST Rowsz 504 S Opt <NONE>
Backed Up 12:08:37,27/08/08 BST Rows 1
Copied To 12:08:37,27/08/08 BST Rows 1
Saved To 30 < 26/09/08
#Found 1 file [from 1 file]
#Directory: TEST/DATA ON DEV
#
7039 Utility File Handler Version 53.530.104 12:30:29
```

The attributes are grouped in a more logical order, and hence are easier to read than the LFILES report produced by SYSTEM/FILEDATA through the CANDE LFILES command. (For a way to see special codefile attributes, see the CODEATTS report example in the [STD command](#)).

All of this is very nice on the screen, but how about sending the results of these commands to the printer ? The only way to do this from CANDE is to put the command in a schedule file, then write out the SCHOUT file produced. There is an

easier way in FLEX Inquiry — by using the [<PRINT option>](#). For example, "LFILES TEST/DATA : PRINT" will send the preceding output to the printer. These <colon option>s can be used in combination, too. To send the other attribute listing to the printer, one could enter "FILES TEST/DATA : ATT,PRINT".

Often, you would like to take action based on the examination of a printout. All of the above results can be sent to a JOBSYMBOL file ,instead of the screen, by using the <FILE option>.

For example

```
FILES : FILE XYZ
```

would send the list of files in your directory to the JOBSYMBOL file XYZ. This could then be edited to produce a job to REMOVE the file, or copy them to a tape. See [Section FileCopy](#).

Another form of output which may be interest uses the BYTE option. This option direct the output to a 'byte stream' file which can be read on PCs.

Example

```
FILES : BYTE "MYFILES.TXT"
```

All of the file related commands (except LIST) will accept a list of filenames and/or directories, e.g. "FILES ABC", "FILES A/=,B/=", etc.

Wildcards may also be used in the file specification, e.g.

```
FILES =/A filenames ending in "/A"  
FILES AC? three-character filenames starting with "AC"  
FILES ER/=DOC= files in ER/= which contain "DOC"
```

See the section: [Filenames, directories and patterns](#) for a full description of wildcard patterns.

As mentioned before, there is an interactive "HELP" facility. On Unisys TD830 terminals, enter "HELP" to place FLEX in HELP mode. The "HELP:PRINT" command is also available.

Most of us have wished for a quick way to clean up files without having to type in long REMOVE lists. The "ACTION screen" of FLEX Inquiry lets us do just that. First, a word is in order about the <syntax option>. This option may be set for the session by entering "OP + SYNTAX", or applied to a command by appending ":SYNTAX" to the command. When this option is set, nothing will be done that permanently changes your file directory. Before we start on the action screen, enter "OP + SYNTAX".

Users without Unisys' TD-compatible terminals will have to skip this part of the guided tour, since this feature uses the special capabilities of these terminals. Now enter "FILES :ACT". After a short time, you will be presented with a screen that will look similar to the following:

| Action | Title & filekind | Segments | Altered | Bkups | Accessed |
|------------------|-------------------------------|-------------------|----------------|----------|-----------------|
| ◀ | A/..... ON DEV | | | | |
| | SEQ | (14) | 17/03/10 | 0 | 07/08/08 public |
| ◀ | A/..... ON DEV | | | | |
| | SEQ | (14) | 17/03/10 | 0 | 07/08/08 public |
| ◀ | A/BB ON DEV | | | | |
| | SEQ | (14) | 17/03/10 | 0 | 09/02/09 public |
| ◀ | A/"B/C" ON DEV | | | | |
| | SEQ | (14) | 17/03/10 | 0 | 29/01/09 public |
| ◀ | J/UT/BLRELEASER/MODIFY ON DEV | | | | |
| | JOB | (15) | 17/03/10 | 0 | 05/03/10 public |
| ◀ | T ON DEV | | | | |
| | JOB & DIRECTORY | (14) | 17/03/10 | 0 | 14/09/09 public |
| ◀ | T/7 ON DEV | | | | |
| | ALGOL | (14) | 17/03/10 | 0 | 20/06/12 public |
| ◀ | X ON DEV | | | | |
| | JOB & DIRECTORY | (42) | 17/03/10 | 0 | 26/02/10 public |
| ◀ | X/A ON DEV | | | | |
| | ALGOL & DIRECTORY | (14) | 17/03/10 | 0 | 01/07/04 public |
| ◀ | X/A/B ON DEV | | | | |
| | SEQ | (14) | 17/03/10 | 0 | 29/01/09 public |
| Familyname : DEU | | | Usercode : BOB | | |
| 2598 | Metalogic Flex | Version 58.580.01 | | 11:16:07 | |

We can enter a number of different actions in the forms boxes, including REMOVE, LIST, DESTROY, LFILES, BACKUP, MZIP, GZIP and LOAD ¹. Several pertinent attributes are shown for each file to help you decide whether to remove the file or not. Try entering "LIST" in one of the boxes, then home the cursor and transmit. To exit this listing, enter transmit again. To exit the action screen, enter "QUIT" in one of the boxes, then transmit. As you can see, the action screen provides a quick way to look through and clean up your files. Note that all of these commands may be entered outside the action screen as well. For a complete discussion of the <action option>, see [Action Option](#).

:ACT,BR gives fewer attributes but shows more files per page

¹The RELEASE, and DESTROY commands only work on Cataloging systems

2598 Metalogic Flex Version 58.580.01 11:16:07

One question that comes to mind when all of this file searching is going on is 'Why can't I select just some of these files ?'. This problem has been solved by the `SELECT` command. For example, one selection of files that we might want to make is the set of files that are larger than 5000 segments, and which have the letter 'Q' in the title. We could construct a `<boolean expression>` that would return true for each file that satisfied the criteria. Such a boolean expression is: `SEGMENTS > 5000 AND "Q" ISIN TITLE`. To use this with the `files` command, we enter:

This set of commands shows a list of exactly the files that we wanted. In general, the SELECT command is used by entering it first, then entering another FLEX command to search for the files. A <selection expression> may be applied to almost all commands.

```
SELECT SEGMENTS > 5000
FILES =0=
```

Page 17

Familynames, Directories, Patterns, and Wildcards.

If the format of the output does not suit your needs — have no fear ! The same language that allows flexible selection of files also allows for the creation of reports. To change the output of the above set of commands, we first enter a report using the REPORT command. As a simple example, suppose we want to make a list of files. We would need a report that would have a comma (",") followed by the file name. This is easy to do; enter

```
REPORT " , " , TITLE
```

This will change the output as desired. For more examples, see the [REPORT command](#), the files in the directory *METALOGIC/FLEX/DEFAULTREPORTS/=, and the section [Replacing FILECOPY](#). All of the commands discussed previously, except for USAGE, have their output created by using a report. Look back at the output produced by the LFILES command; this really points out the power of the OPAL language, since this is also produced by an OPAL report.

To avoid odd results in the rest of this guided tour, we should mention a way to turn off a selection expression or report. Enter "SELECT -" to disable the selection expression, then enter "REPORT -". Both of these are retained; entering "REPORT +" or "SELECT +" will re-enable them if you would like to use them in the remainder of examples.

Often, you would like to perform a series of actions in sequence, perhaps in batch mode. FLEX has the ability to read commands from a file and execute them, just like CANDE. These files are called "DO files". To create a "DO" file, create a file in CANDE with commands to FLEX one to a line. If a command goes over one line, each partial line may be ended with a backslash ("\"). In most cases the backslash is not necessary as Flex is usually able to work out if a new line is a continuation of a previous command , or a new command.

Here is a simple example, for CANDE users only. If you are still in FLEX Inquiry, enter "QUIT" to return to CANDE. Now enter the following (CANDE's responses are in bold type):

```
MAKE T TEXT
#WORKFILE T: TEXT
100 WHAT
200 SELECT SEGMENTS GEQ 500
300 REPORT TITLE, " (", SEGMENTS, " ) "
400 FILES
U FLEX DO
#UPDATING
#RUNNING 3222
#? ... (here FLEX Inquiry starts running)
```

These commands can also be done in batch mode by running FLEX using WFL. See the [DO command](#) for more information.

Combining a simple Report with the Byte output option allows the creation of a 'CSV' file which could be imported into an Excel spreadsheet.

Example

```
REP HEAD "Filename, Age, Size"  
REP Filename, " ", Days (Creationday, Today) , " ", Segments*180  
FILES :Byte "MYFILES.CSV"
```

Sample output file MYFILES.CSV

```
Filename, Age, Size  
(BOB) TEST, 2, 1300320  
(BOB) TEST/1, 196, 2520  
(BOB) TEST/2, 2294, 2520  
(BOB) TEST/3, 2294, 47880  
(BOB) TEST/4, 2218, 2520  
(BOB) TEST/DO, 2613, 2520  
(BOB) TEST/DO/1, 2613, 2520  
(BOB) TEST/DO/PARAMLEN, 1545, 2520  
..  
..  
..  
..  
(BOB) TEST/REDIRECTION, 169, 17640  
(BOB) TEST/PRINTSUPPORT, 126, 7560  
(BOB) TEST/ATTRIBINFORMER, 1590, 20160  
(BOB) TEST/CHECKPOINTARRAY, 1617, 17640  
(BOB) TEST/ATTRIBUTE MODIFIED, 2694, 2520
```

We mentioned that there are several examples of reports and "DO" files in the *METALOGIC/FLEX/DEFAULTREPORTS/= directory. These are good examples to study before making your own reports. If there is a report or "DO" file that you use often, or that would be of wide interest, it may be put into the *METALOGIC/FLEX/DEFAULTREPORTS/= directory.

These files can be accessed by number by using the [STD command](#). In FLEX, enter "STD". If the files are on line, you will see several names. Any of these "DO" files or reports can be "done" by entering the number associated with the name on the screen.

Below is a possible response to the STD command.

```
The following default reports are available:
 1. *DU          2. *ARC          3. *HELP
 4. *CLEAR       5. *CRUNCH       6. *COMPGEN
 7. *PACKDIR     8. *CLEANSTD      9. *CODEATTS
10. *BACKUPPRINT 11. *COSTJUSTIFY
Enter a number from 1 to 11 or 0 to exit
An asterisk (*) indicates that the report is a system file

7095 Utility File Handler Version 53.530.104 14:39:46
```

The site manager can enter STD reports into the "" directory that will be seen by all users².

Enter 9 to get the CODEATTS report. This is activated by doing a FILES command on a code file, for example:

FILE *OBJECT/FLEX

This produces a display with one page per file:

```
Title : *OBJECT/FLEX
-----
General attributes      Codefile info attributes      Boolean attributes
-----
Filekind :             DCALGOLCODE      Executions:             7      CPed : FALSE
Segments :             5040             Priority :             0      PPed : FALSE
Codeversion:           53.180            Parameters:            1      PPedtransprt: FALSE
Cycle/version:         530:0010          Restarts :             0      RPed : FALSE
Created 26/08/08 @ 16:46:55              Nonexecutable:         FALSE
Accessd 27/08/08 @ 14:24:55              Systemfile:             FALSE
Altered 26/08/08 @ 16:49:22              Language :             DCALGOL
Changed 26/08/08 @ 16:49:22              Sharing :             LIBDONTCARE
Unsafe : FALSE
Suppressed : FALSE
Inuse : TRUE

Security Information    Capable?      Limits      Stack size info
-----
Security type :        PUBLIC      DMS: N      MaxIO: 0      D1stack: 537
Security use :          IO      IPC: Y      MaxPROC: 0      D1WSavg: 13861
Protection :          SAVE      SORT: N      WSlimit: 0      D2stack: 2471
Guardfileset :         FALSE      LIBR: N      WSavg : 39967
Security admin :        FALSE
Secadmin transp:        FALSE

7095 Push SPCFY for more, XMT to break or NEXT +/-4 14:42:12
```

After using a standard report you will probably want to do:

SEL -
REP-
or
STD
and then select 4 for Clear.

This will restore Flex to normal operation.

² subject to normal system security on the files

Possibly the most informative of all FLEX Inquiry commands is the USAGE command when used to summarise across directories. For example, suppose that you wished to see a report of all the files under the PENDING directory that have not been altered for 30 days - summarised by directory. Enter :

```
SELECT DAYS (ALTERDAY,TODAY) GEQ 90
USAGE PENDING:1,BRIEF
```

The response may look like this:

| DIRECTORY | FILES | SEGMENTS | LARGEST |
|----------------------|-------|----------|---------|
| PENDING/META/= | 16 | 21690 | 4522 |
| PENDING/SYSDEF/= | 2 | 2240 | 1568 |
| PENDING/SUPERVISOR/= | 2 | 270 | 180 |
| TOTALS | 20 | 24200 | 4522 |

For any command which involves a SELECT or a wildcard in the directory part, it is usually best to add the :Flat option. When the Flat option is specified Flex scans the system (Flat) directory directly and then sorts the results back into directory order. This is generally much quicker than using the normal MCP access method (GetStatus).

There is an additional option which is very useful to search for files on multiple families.

```
Files =BOB= ON = :Fast
```

Would search for any file in my directory with BOB in the title.

If you are privileged or your usercode has the Granulated privilege GSDIRECTORY set, Files (*)=BOB= ON =:Fast would search all non-usercoded and all usercoded files.

Another very useful option for searching is the :[FIND option](#).

```
SEL FILEKIND=DCALGOLSYMBOL
FILES (*)=:F1,FIND 'GETSTATUS'
```

Would find all of the DCALGOL symbol files which contain "GETSTATUS". The access date of the scanned files will not be updated.

The Fast option could not be used in the above example as neither the FILEKIND attribute nor the :Find option is compatible with a Fast search.

To end the guided tour, enter "QUIT". This will terminate this run of FLEX Inquiry,

and return you to CANDE or MARC. For more information on any of the commands, see the full description of the command.

Setting options for your terminal

There are several <colon option>s that affect how FLEX Inquiry operates with your terminal. These options should be set once to "customise" FLEX Inquiry for your terminal environment and the way that you work.

These options are stored on a terminal by terminal basis in a file under your usercode, on your default family. This file is called "FLEXOPTIONSFORME". If this file is lost or removed, you may have to re-enter your saved options. Options may be set and saved with a single command. For example, to set and save the option USDATES, enter the following in FLEX:

```
OP USDATES : SAVE
```

To see what your options are, enter "OP ?" in FLEX. Options that are **off** do not need to be set or saved. If you wish to turn **off** an option, prefix it in the list with a minus sign ("-"). For example,

```
OP -USDATES : SAVE
```

will reset the USDATES option, all of these settings will be saved. The significance of these options will be explained below.

For users in the United States, the option USDATES should be set. This will cause all dates to be displayed in the usual MM/DD/YY format used in the U.S.. If it is not set, dates will be shown in the European format of DD/MM/YY. In general, your site manager will have set this option by default, so you will not need to.

For a further discussion of options, see the OPTION command.

Further Tips

If you only need to do a single command in FLEX, you may start the inquiry by entering the command together with 'U FLEX', i.e.

```
U FLEX <command>
```

which will cause FLEX to execute the command, and then exit. To remain in FLEX, append the <INTER option> to the above.

```
U FLEX <command> :Inter
```

An option or option list may appear after the command, i.e.

```
U FLEX <command> : <option 1>, <option 2>, ...
```

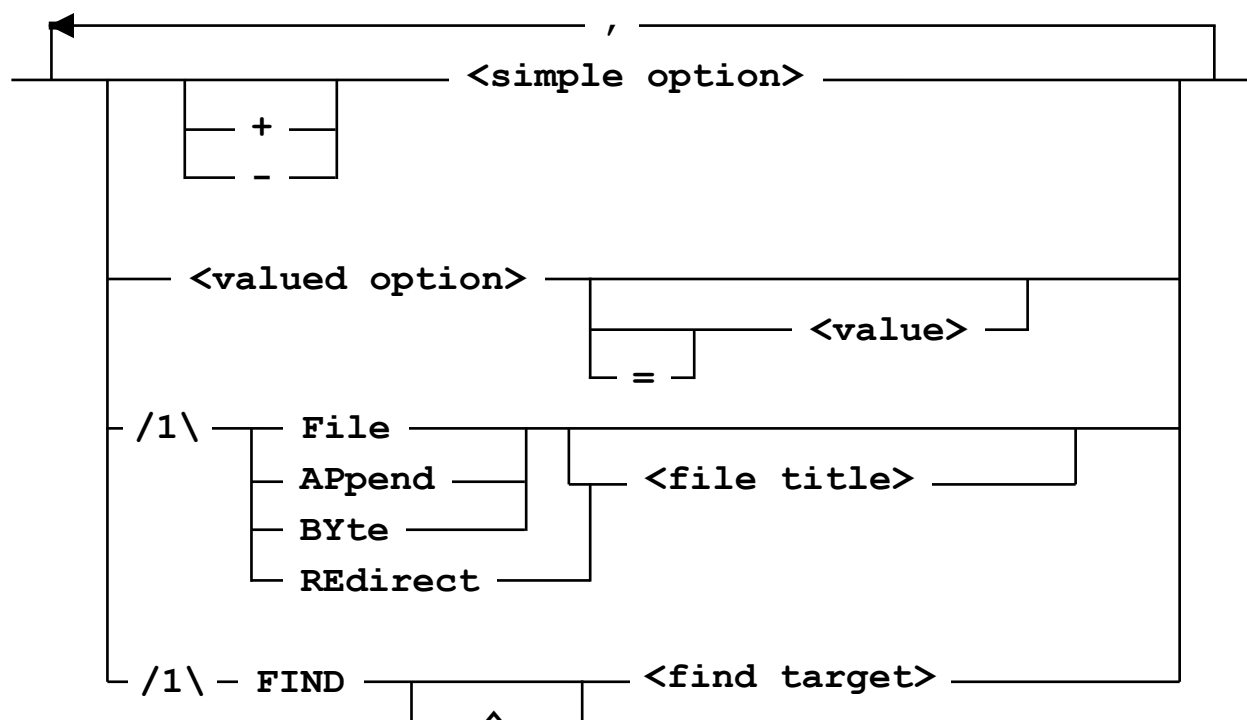
These options will be applied to the command. Most of the commands to FLEX may be followed by a list of options, separated by commas, and found at the end of the command after a colon, just like CANDE. For a list of valid options, and their current setting, enter 'OP ?'. The minimum abbreviation is indicated in capital letters; also, only the first six characters are examined.

Options that appear after a colon at the end of a command are considered temporary, and will return to their default value at the termination of that command. To set an option permanently, use the 'OP' command. If you wish to have your current options saved, so that they will be used for all of your subsequent runs of FLEX, enter

OP :SAVE

To restore your options to these saved values at any point during your FLEX session, enter 'OP *'.

The various options available in FLEX may be strung together in lists, which may appear in the 'OP' command, or at the end of a command, after a colon (":"). The syntax diagram for an <option list> is as follows:



A <simple option> is one of : CAPS, CATALOG, RESIDENT, NONRES, ATT, REONLY, CANDE, PRINT, TERM, UNSEQ, TRUNCATE, HEX, BRIEF, AUDIT, MONITOR, LINEAR, FLAT, USDATES, NOMSGS, OLDWFL, SYNTAX, TWOBACKUPS, SETLIST, SETCODE, or INTER.

A <valued option> is one that may take a value or can be used as a <simple option>; these are DEBUG and WAIT. The <file option> allows the output to be directed to a file, given by <file name>. The file specified must not already exist. If no <file name> is given, an appropriate default will be selected.

The use of "DO files" can improve productivity for repetitive tasks. These may be placed in the STD directory to make for even easier access. See the STD command for information.

The <flat option> and the <linear option> can be used to speed up file searches by a factor of one thousand. They are usable by any user that has access to the *SYSTEM/DIRECTORY/001 (flat), or *SYSTEM/CATALOG/001 (linear) using normal system security.

The <action option> allows an easy, menu-oriented way to cleanup and review files.

Filenames, Familynames, Directories, Patterns, and Wildcards

Familynames

<familyname>s are used to specify which pack or disk (or tape) a particular file may be found on. A <familyname> consists of up to 17 alphanumeric characters, i.e. A-Z and 0-9. Examples are SYSTEM, BATCH, 34, and DATABASES.

If the familyname is omitted it defaults to DISK, and from there to the 'FAMILY DISK=' substitution. When used with lists of files or directories an omitted family defaults first to the family of a subsequent member of the list. Only if there is no explicit family afterwards, will it default to DISK.

Example

Files X, Y ON ZPK == Files X ON ZPK, Y ON ZPK
This also applies to containers, units, etc.:
Files X, Y MT 27 == Files X MT 27, Y MT 27
Files X, Y IN ZZ == Files X IN ZZ, Y IN ZZ
But
Files X ON ZPK, Y == Files X ON ZPK, Y ON DISK

Hostnames

On sites with a CopyWrite license Privileged users may search on other hosts. The syntax requires "AT <hostname>" be placed after the normal title/pattern specifier. This syntax is valid with FILES ,USAGE and LOAD.

Examples.

FILES = CD 50 AT HISHOST
USAGE (FRED)= AT SOURCES, (JIM)= ON BKMT(MT) AT TAPESYS:BR
LOAD OLD/BONES IN (SHAFT)CDIMAGE ON IMAGEPK AT DRHOST, NEWR/=

If the CopyWrite option is used on a LOAD AT, a special form of COPY is used so that if there is a conflict, the file is only overwritten if the incoming one has a later timestamp. This makes it easy to keep a host up-to-date with file changes.

Note that ISO/JOLIET CDs, and non-CopyWrite containers, are currently not accessible using "AT", and that wildcards are disallowed on the family name.

Usercode

A <usercode> is a name consisting up to 17 EBCDIC characters. These need not be alphanumeric; if they are not, they must be enclosed in ""'s. When used as part of a filename, it is surrounded by parentheses. Your usercode is the one that you

use to log on with.

Filenames

A <filename> is an identifier that you and the system use to keep track of your files.

A <filename> may consist of one or two parts : a <usercode> part, a <filenameid> part, or both.

Examples of <filename>s are:

```
*OBJECT/FLEX
(SIMONS)MEMO/FRED
(SITE)JOB/EXAMPLES
MYFILE/ABC
SHORTMEMO
A/VERY/LONG/FILE/NAME/WITH/MANY/MANY/LEVELS
```

Directories

A <directory> is a group of files. It may be either a partial <filename>, a <filename> followed by '/=', a usercode in () followed by =, *=, (=)=, or (*)=. For the meaning of the last three, see the section on <pattern prefix>.

Examples of <directory>s are:

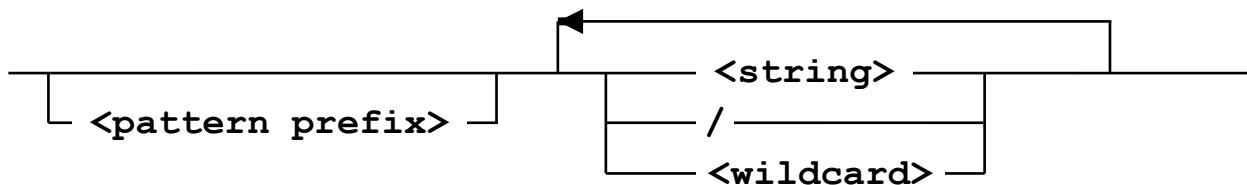
```
*OBJECT/=
MYFILES/ABC/=
(SITE)=
(SIMONS)MEMO/=
```

Patterns and wildcards

The commands FILES, LFILES, LOAD, BACKUP, RELEASE, REMOVE, SECURE, USAGE and DESTROY can use a <pattern> to select files. The <pattern> can be a <filename>, a <directory>, or either of these with embedded wildcard characters.

Note that the <familyname> is not part of the <pattern> but wildcards may be used after the ON part

A <pattern> is



<string> is an unquoted sequence of letters, digits, hyphens and underscores, or a quoted string without embedded quotes.

<pattern prefix> -- see next section.

Two consecutive <string>s are not allowed. Two consecutive slashes are not allowed.

Files are selected which match the <pattern> under the rules:

1. A <char> or slash must match exactly.
2. A <wildcard> will match as follows:

| Wildcard | Action |
|----------|--|
| "=" | match any string, including an empty string |
| "?" | match any single character |
| "&" | match any alphanumeric character (upper and lower case letters and digits) |
| "@" | match any upper or lower case alphabetic character |
| "#" | match any numeric digit |
| "~" | match any string not containing "/" |
| "\" | Escape character use before any other wild card to use the literal value of the character not its wildcard action. Ex. =A\=B= would match any string containing A=B |

Often your <pattern> will be just a <filename> or <directory>, for example:

| <pattern> | sample <filename> | matches with FILE LFILE USAGE | matches with Other commands |
|------------|----------------------|---|--------------------------------------|
| MY/FILE | MY/FILE | Y | Y |
| OP-ED/AL/= | OP-ED/AL/STEW | Y | Y |
| | OP-ED/AL/ST/JO | Y | Y |
| | OP-ED/ALJOE | N | N |

In some commands a <filename> is equivalent to the <directory> formed by adding / = to the <filename>. In all commands, a directory <pattern> will select the file formed by deleting the /=. Examples:

| <pattern> | sample <filename> | matches with FILE LFILE USAGE | matches with Other commands |
|------------|----------------------|--|-----------------------------------|
| MY/FILE | MY/FILE/TOO | Y | N |
| OP-ED/AL/= | OP-ED/AL | Y | Y |

Wildcards generalise the idea of a <directory>. Where a <directory> matches file titles with a constant head and variable tail, wildcards can be embedded or leading parts of the <pattern>:

| <pattern> | sample <filename> | matches with FILE LFILE USAGE | matches with Other commands |
|-----------|----------------------|--|-----------------------------------|
| =/FILE | X/FILE | Y | Y |
| | MY/FILE | Y | Y |
| | FILE/MINE | N | N |
| ?/FILE | X/FILE | Y | Y |
| | MY/FILE | N | N |

| <pattern> | sample <filename> | matches with FILE LFILE USAGE | matches with Other commands |
|-----------|----------------------|--|-----------------------------------|
| =THE= | THE/FIRST/ FILE | Y | Y |
| | MY/OTHER/ ONE | Y | Y |
| A=E | ANY/OLD/FILE | Y | Y |
| | ANY/OTHERS | N | N |
| | AE | Y | Y |
| A?E | ABE | Y | Y |
| | AE | N | N |
| | ANY/OLD/FILE | N | N |

When any wildcard is used other than a trailing /=, the equivalence between <filename>s and <directory>s is "not operative". To be selected, the file title must match the <pattern> precisely, and all commands will select the same files for a given <pattern>.

| <pattern> | <filename> | matches with FILE LFILE USAGE | matches with Other commands |
|------------|--------------|--|-----------------------------------|
| =/FILE | MY/FILE/TOO | N | N |
| OP?ED/AL/= | OP-ED/AL | N | N |
| | OP-ED/AL/JOE | Y | Y |
| =THE=/= | THE/FIRST/EX | Y | Y |
| | THE/OTHERS | N | N |

? matches one, = matches zero or many. To match (for example) two or three, use ?? or ???. To match one to many, use ?=, etc.

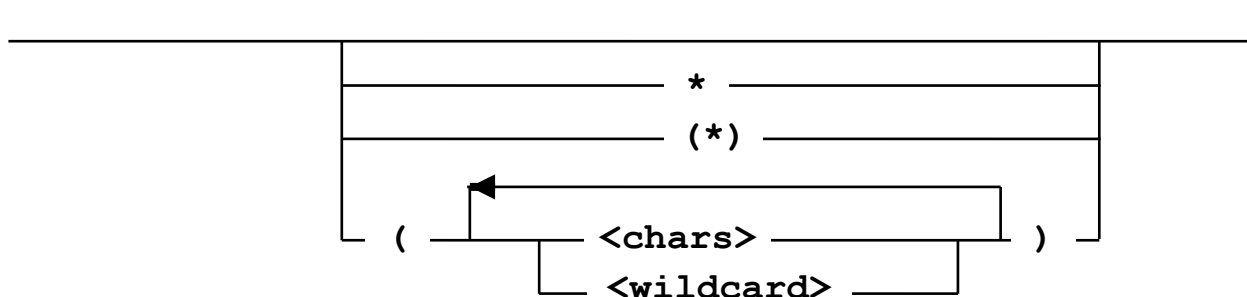
| Matches? | <pattern> | | | | |
|--------------|-----------|------|------|---------|--------|
| <filename> | A=E | A?E= | A?=E | A???=E= | A=E?E= |
| ANY/OLD/FILE | Y | N | Y | Y | N |

| Matches? | <pattern> | | | | |
|-------------|-----------|---|---|---|---|
| ANY/OTHERS | N | N | N | Y | Y |
| AE | Y | N | N | N | N |
| ABE | Y | Y | Y | N | N |
| ABLE/BODIED | N | N | N | Y | Y |

For information on the use of quotes in patterns, see [Using Quotes in FLEX](#).

Usercodes, search order, and the <pattern prefix>

A <pattern prefix> is:



An omitted <pattern>, or a lone =, selects just the files in your usercode directory, unless there are no files in your usercode directory, in which case it selects all non-usercoded (* files) files on the family. A command with any other unprefixed <pattern>, with or without wildcards, first searches your usercode directory; if no matches are found, it then searches the system directory (* files). However, if some files in your usercode directory match the <pattern> but are not selected due to a SELECT expression in effect, the system directory is not searched.

For example:

```
You:    SELECT FALSE
FLEX:    #Selection Expression compiled OK
You:    FILES ER
FLEX:    #Selected 0 files [from 132 files]
FLEX:    #Directory: ER ON SYSTEM
```

You can force FLEX to search outside your usercode by using a <pattern prefix>. However, you are bound, as always, by standard MCP security restrictions, and most files outside your usercode may be invisible to you. Your visible universe may be restricted by the privilege status of your usercode, the SHOWFILES setting of other usercodes, and the security attributes of individual files.

For example:

```
*           : searches system files only
(=)         : searches files under all usercodes
(*)         : searches system files and all usercoded files
```

```
(...)      : if ... is a usercode, searches that usercode.  
          : if ... contains wildcard characters, then it acts  
          as a pattern for the usercode.
```

Granulated privileges are respected.

If Flex is run from a usercode with GSDIRECTORY privilege then :Flat , :Fast and :Find are allowed on any directory.

If the READ privilege is set then any file may be listed.

If The REMOVE privilege is set then any file may be removed.

If the CHANGESEC privilege is set then any file may have its security changed.

Neither = nor ? will match the () delimiting a usercode. Usercode matching must be handled separately by the <pattern prefix>. Thus (=)/FILE is not a valid pattern; (=)=/FILE is valid and will match files under all usercodes whose names end with /FILE.

Using Quotes in FLEX

The quote character (") is used in two distinct ways in FLEX: in OPAL expressions (for REPORT and SELECT), and in <filename>s and <pattern>s.

Using quotes in OPAL expressions

Include a quote in a REPORT by using two quote characters to represent a single quote in the output. These two quotes must be inside a quoted string. To represent the empty string, use two non-embedded quote characters. Either single or double quotes may be used to define a string so 'x"y' would represent x"y.

Examples of REPORTS with embedded quotes:

| REPORT | sample output |
|-----------------------------|---------------------------------------|
| ""test " | "test |
| ""abc ",TITLE," xyz "" k "" | "abc ATITLE xyz " k " |
| "" | <null string> |
| '("test")' | ("test") |
| "abc ',Title,' xyz " k " | "abc (MYUSER)MY/FILE ON DEV xyz " k " |

Using quotes in <filename>s and <pattern>s

Occasionally <filename>s contain special characters. These are the titles which are displayed using quotes by FLEX and by Unisys software. The individual identifiers in the <filename> must be quoted on input when they contain special characters.

Embedded quotes are not meaningful, because Unisys software does not allow quotes to be embedded in the <filename> identifiers.

File names requiring quotes are inconvenient and sometimes confusing. Metalogic recommends that you avoid such names.

Examples of valid quoted <filename> :

```
ABCD/" abc deff77"/UUU
"a lower case"/FILENAME/EXAMPLE
"other #@$!"/CHARACTERS/CAN/" appear"
```

Special characters must be quoted when they appear in a <pattern>, but the quotes themselves need not match precisely the quotes in the target <filename>:

| <pattern> | Matches | Also matches | Doesn't match |
|---|------------|---------------------------|-------------------|
| "A.C"= | "A.C" | "A.C.E" | |
| =". "= | "A.C" | MY/ NEW/"FILE.OBJ " | |
| "["="]" | "[WHAT]" | "[ONE"/ OF/"THREE]" | WHERE/"[EVER" |
| SYS/="#"= | SYS/"#ONE" | SYS/ FIL/"COPY#01" | "SYS#" |
| When an = or a ? is quoted, it loses its wildcard status: | | | |
| "=. =" | "=. =" | | FILE.SCR |
| "FILE.???" | "FILE.???" | | FILE.OBJ |
| "?"="."= | "?F"/"X.O" | "?=. =" | AB.C |

Slashes have special significance in A-Series file names, to narrow the search of directories which can potentially contain entries for many thousands of files. A <pattern> with an unquoted slash will occasionally fail to report a matching file title with a quoted slash. Quoting the slash in the <pattern> will force the match, but possibly at the cost of much slower response.

Metalogic strongly recommends that you never use file names containing quoted slashes, and that you do not quote slashes in <pattern>s.

| <pattern> | Match | No Match |
|-----------|---------|----------|
| "A/B" | "A/B" | A/B |
| ="/A/B" | X/A/B | |
| | X/"A/B" | |

| <pattern> | Match | No Match |
|-----------|---------|----------|
| "A/B"= | "A/B/C" | |
| | A/B/C | |
| | A/B | |
| | "A/B" | |
| | A/"B/C" | |
| A/B/= | A/B/C | A/"B/C" |
| | A/B | |
| A/B= | A/B/C | |
| | A/"B/C" | |
| | A/B | |
| =A/B/= | A/B/C | |
| | "A/B/C" | |
| | A/"B/C" | |

Commands

Command Groupings

The commands to the FLEX inquiry may be grouped according to function, as shown below. For full descriptions of the available options, see the OPTION command.

1. Directory search & reporting commands

| | | | |
|---------------------|-------------------|------------------|-------------------|
| FILES | FILES:CAT | FILES:ATT | FILES:FLAT |
| FILES:LINEAR | FILES:FAST | | |
| LFILES | NEXT | SAME | SELECT |
| REPORT | STD | USAGE | |

2. File maintenance commands

| | | | |
|---------------|-----------------|------------------|--------------|
| BACKUP | LOAD | FILES:ACT | PURGE |
| FORGET | REMOVE | DESTROY | |
| | SECURITY | | |

3. Session handling commands

| | | | |
|---------------|---------------|-------------|-------------|
| ACCESS | FAMILY | QUIT | USER |
| WHAT | | | |

4. Rules base management commands [Full FLEX only – see FLEX Site Management Manual]

| |
|--------------|
| RULES |
|--------------|

5. Volume Library and Catalog maintenance commands

| | | | |
|---------------|--------------|------------|-----------|
| LISTVL | MAINT | PDT | PV |
|---------------|--------------|------------|-----------|

6. Miscellaneous commands

| | | | |
|-------------|------------|-------------|----------------|
| BRK | DO | HELP | INSTALL |
| LIST | ODT | WFL | WRITE |
| FIND | | | |

7. The OPTION command and <colon options>

| |
|---------------|
| OPTION |
|---------------|

The commands to FLEX and the options are described below with a railroad diagram, syntax, practical usage, and examples. They are listed in alphabetical order, except for individual <colon option>s, which are in alphabetical order under the OPTION command.

Description of the command groupings

The basic command for searching file directories is the FILES command. The FILES, FILES:CAT, FILES:ATTR and LFILES commands are improved versions of the CANDE FILES and LFILES commands. Many of FLEX's options act together with these commands.

FILES:FLAT and FILES:LINEAR provide methods for searching the SYSTEMDIRECTORY and the CATALOG, respectively, at speeds that are up to *1300 times faster* than GETSTATUS. USAGE gives a summary of all files under the usercode under which is running.

SELECT and REPORT provide very powerful ways of scanning directories and reporting in a completely user-specifiable way. None of these commands alter files or their attributes in any way.

File maintenance commands are used to remove, load, change security, purge, and other actions which may destroy a file or change its attributes. They are available to all users, with the restriction that only a privileged user or one with appropriate Granulated privilege may act on another user's files.

If the READ privilege is set then any file may be listed.

If The REMOVE privilege is set then any file may be removed.

If the CHANGESEC privilege is set then any file may have its security changed.

Session handling commands allow you to establish a running environment for FLEX — <usercode>, <accesscode>, default <familyname>; or to end a FLEX session.

Rules base management commands allow interrogation and updating of the FLEX system's RULES base. This contains the rules used by the FLEX system to backup, remove, and report on files.

Volume library and catalog maintenance commands provide an easy interface to the VOLUME LIBRARY and the SYSTEM/CATALOG. These are used primarily by the site manager and the tape librarian.

The remaining commands provide additional functions and are used in conjunction with the other commands.

Command Summary

| | |
|----------------------------------|---|
| ACCESS | change <accesscode> for FLEX session |
| BACKUP | make copies of resident files to other media |
| BRK | stop current command & go to command mode |
| DEFAULT | establish settings for the global options of FLEX |
| DELETE or FORGET | delete a file entry & its backup information |
| DO | accept FLEX commands from a batch input file |
| FAMILY | change default <familyname> for FLEX session |
| FILES | show file(s) or directories |
| FILES: ATT | show file(s) and their attributes |

| | |
|--|--|
| FILES: ARC | show file(s) with archive information |
| FILES: CAT | show file(s) and their catalog information |
| FILES: FAST | very fast method of searching file directory but with few attributes returned. |
| FILES: FIND | show files which contain specified text |
| FILES: FLAT | fast method for searching file directory |
| FILES: LINEAR | fast method for searching file directory when cataloging option is set |
| FILES: ACT | menu screen for file maintenance |
| FORGET or DELETE | delete a file entry & its backup information |
| FIND | find and report line within a file which match a specified target |
| HELP or TEACH | interactive or printed help on FLEX inquiry |
| INSTALL | automatic installation of the FLEX package |
| LFILES | show file(s) and their attributes |
| LIST | show a listing of a symbolic file |
| LISTVL | report on tape & pack volumes in VOLUME LIBRARY |
| LOAD | retrieve file(s) automatically from tape backup |
| LOAD :SUFFIX | Appends a timestamp suffix to each loaded file |
| LOG | view the Flex Log file |
| MAINT | report & diagnostic function for SYSTEM/CATALOG |
| NEWS | synonym for HELP NEWS, shows latest changes |
| NEXT | page forwards and backwards in FILES command |
| MZIP | compress files |
| ODT | pass ODT commands to CONTROLLER |
| OPTION & SO | set/reset options for the FLEX inquiry |
| PDT | show tape directory of LIBRARY/MAINTENANCE tape |
| PURGE | Forgets all backups of a file, but does not remove it |
| PV | show VOLUME LIBRARY info for a tape/pack volume |

| | |
|-------------------------------|--|
| QUIT | exit FLEX inquiry |
| RELEASE | REMOVE a file if safely backed up |
| REMOVE | REMOVE a file without checking backup status |
| REPORT | design customised reports for FLEX inquiry |
| RULES | for management of the rules and rules base (FLEX Site Mgmt.) |
| SAME | show the same page in a FILES command (see NEXT) |
| SAVE | saves state (Selection and Reports) |
| SECURITY | change the security of a file or directory |
| SELECT | set a criterion to select certain files |
| SO&OPTION | set/reset options for the FLEX inquiry |
| SPLIT | release current LINE print file for PrintS processing |
| START | start a WFI job, may prompt for parameters |
| STD | show a menu of available standard reports |
| USAGE | show a summary of files for a user or directory |
| USER or HELLO | change <usercode> for current FLEX session |
| WFL | pass commands to the WFL compiler |
| WHAT | show the current running environment for FLEX |
| WRITE | same as LIST but output is routed to printer |
| PURGE | completely eliminate a file(s) & backup info |
| | |

Command Descriptions

The names of the command are show in mixed case. The upper case part indicates the minimum abbreviation.

ACCESS command



The ACCESS command allows you to change the accesscode under which FLEX Inquiry is running. The <accesscode> specified must be valid for the current <usercode>. If an <accesscode> is required for the <usercode>, it must be supplied; if not, '.' may be specified as the <accesscode>. If an incorrect <accesscode>/<password> is given, the program will be DS'ed. An example of the command is:

ACCESS MONGOL/HORDES

To run without an <accesscode>, enter 'ACCESS .' . This may not be used if an <accesscode> is required for the <usercode>. If an <accesscode> is allowed for a <usercode>, it may be supplied with the USER command immediately after a semicolon (";") character, e.g.

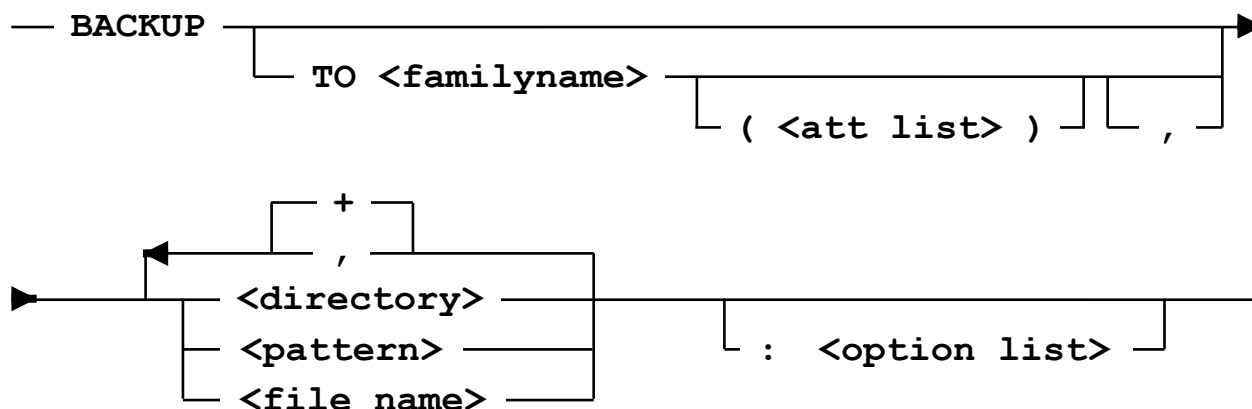
USER X/Y ; ACCESS = AC/APASS

There is no method currently for changing to a new usercode if ACCESSCODEREQUIRED is set in the USERDATAFILE.

See also:

USER command

BACKup command



BACKUP is used to make copies of files resident on PACK families to other destinations. After it has selected the files it generates a LIBRARY/MAINTENANCE COPY command to move the files. If no TO clause is specified, FLEXBACKYYDDD is assumed, where YYDDD is the current 5-digit Julian day.

Options

If the option TWObackups is set, two destinations are used, by adding Y and Z to the usual volume name.

If the option CATalog is set, an COPY&BACKUP is done, unless the system option Cataloging is reset, in which case an ARCHIVE FULL is used.

If option COPYWrite is used, the COPY will be sent to CopyWrite. If no TO clause is used, the destination will be a CD image file.

Examples

BACKUP XYZ/=

Copy the resident files in XYZ/= to a tape named, say, FLEXBACK06123.

BACK TO C(CD),Y/=:TWO,CAT

Copy Y/= with COPY&BACKUP to 2 CDs CY & CZ

See also:

[<filename>](#) [<directory>](#) [<pattern>](#) [<copywrite option>](#)
[<twobackups option>](#) [<syntax option>](#) [<catalog option>](#) [<action option>](#)

BRK command

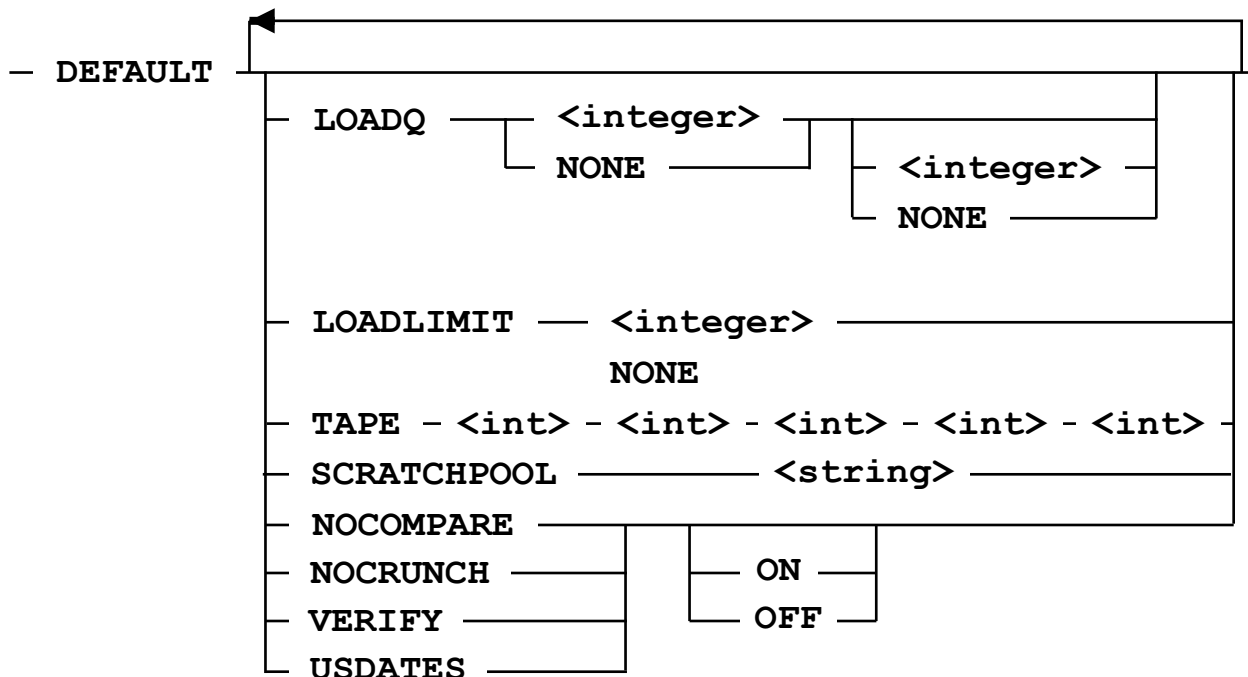
———— BRK —————

The BRK (break) command may be used when FLEX is waiting at a page break to terminate the current command, and return to command mode. To completely exit from FLEX, use the QUIT command.

An ordinary "?BRK" may be entered to stop output when FLEX Inquiry is sending to the terminal, but it will not work when it is waiting for user input or waiting at a page break.

Flex Inquiry will not check for user input (including the BRK) command unless the [<monitor option>](#) is set.

DEFAULT command



The DEFAULT command is used to specify certain parameters that will be used as defaults by the FLEX system. Update may only be used by the rules manager. i.e. Flex must be running with both the usercode and accesscode of the rules

manager.

DEFAULT on its own shows a screen with the current values. These can be updated by a further screen. Adding the values to the command, as described below, is normally only used for batch input.

| | |
|-------------|---|
| LOADQ | <p>The <integer> parameter indicates which system QUEUE should be used by the FLEX utility for LOADING of files.</p> <p>If a second queue is specified, that queue is used for large LOADs, i.e. ones needing more than LOADLIM tapes.</p> |
| LOADLIMIT | <p>Specifies the number of tape volumes in a LOAD which if exceeded, will cause FLEX to treat it as a large LOAD. If there is a large LOADQ, the LOADs will be placed in the large LOADQ. If there is no large LOADQ, FLEX will trim the LOAD to that number of tapes, unless the LOAD is done from a PU.</p> |
| TAPE | <p>The five integers specify defaults used when FAMILYMANAGER or MERGETAPE create tapes. The integers represent DENSITY, LENGTH (Mbytes), SAVEFACTOR, Retry logic, and LIBMAINTDIR file creation.</p> <p>The values for retry logic are as follows.</p> <p>0 -> NONE Copies will not be restarted.</p> <p>1 -> ERROR Copies will be restarted if the copy task terminates abnormally. (i.e. DSes)</p> <p>2 -> FULL As ERROR but will also restart if any files are not copied. (i.e. TASKVALUE=1)</p> <p>The values for LIBMAINTDIR are:</p> <p>0 -> Auto Makes LIBMAINTDIR files if the TAPE density can have LOCATECAPABLE drives.</p> <p>1 -> Always make LIBMAINTDIR files.</p> <p>2 -> Never make LIBMAINTDIR files.</p> |
| | |
| SCRATCHPOOL | <p>This string specifies the default scratch pool used for FAMILYMANAGER or MERGETAPE tapes.</p> |

| | |
|-----------|--|
| NOCOMPARE | Do not use 'COPY&COMPARE' in the LIBRARY/ MAINTENANCE jobs run by FLEX. NOT recommended, see VERIFY. For this and the following options, if an 'ON/OFF' is omitted, 'ON' is assumed. |
| VERIFY | Forces 'COPY&VERIFY' in all LIBRARY/ MAINTENANCE jobs that FLEX runs. Recommended. |
| NOCRUNCH | Do not CRUNCH any files through FAMILYMANAGER. |
| USDATES | All dates will be shown in the American date format MM/DD/YY. If reset, dates will appear as DD/MM/YY. |

DO command

```
— DO — <file name> ( <parameter list> ) —
```

The DO command allows you to input FLEX commands from a file. FLEX Inquiry will read the commands from the file given by <filename>, and accept them as batch input, processing them one by one until it gets to the end of the file. If you are running in interactive mode, it will ask you for a new command when it is done; if run through WFL or a job, FLEX will terminate when it gets to the end of the file.

If <filename> is omitted, and you have a CANDE workfile, FLEX will use that as the input file. The file may be of any source or data kind recognised by CANDE.

Parameters

Parameters may be passed to FLEX 'DO' files. The implementation is similar to that provided by CANDE and is illustrated by the following example DO file:-

```
PARAMETER (DIRNAM, SOURCEFAM, TAPENAME)
REPORT FOOT "FROM #SOURCEFAM# (PACK) TO #TAPENAME#", /, "END JOB"
REPORT HEAD "BEGIN JOB SAVE/#TAPENAME#; ", /, "COPY"
REPORT IF #FCNT.Sum(1) > 1 THEN ", " ELSE EMPTY, FILENAME
FILE #DIRNAM# ON #SOURCEFAM#: FLAT, FILE WFL/#TAPENAME#, AUTORM
```

The parameter names are defined within parentheses on the first line and in this example are DIRNAM, SOURCEFAM and TAPENAME. The parameters are referenced by enclosing the name within #'s eg. #SOURCEFAM#. When the DO file is invoked references to parameter names will be replaced by the parameter values supplied.

A DO file that expects parameters is invoked by following the DO file name by a list of parameter values enclosed in parentheses. For example if the above example was called MY/DO/FILE it could be invoked as follows:-

```
DO MY/DO/FILE (METALOGIC/=,PACK,METADUMP)
```

This would result in METALOGIC/= being assigned to the parameter DIRNAM, PACK to SOURCEFAM and METADUMP to TAPENAME.

Multi line commands

FLEX attempts to group multi-line REPORT or SELECT scripts by searching for command words at the beginning of each new line. The '\' (backslash) character can be used to indicate that a command line is continued on the next line. Some ambiguities where FLEX keywords collide with OPAL functions and attributes have been addressed but other collisions are possible. In such cases, using the '\' terminator on a previous line in the DO file will always force FLEX to regard the following line to be part of the SELECT or REPORT. Comments may appear in the file after a percent ("%") sign, since FLEX ignores any text on the input record after a "%" sign.

Typical Use

FLEX DO files can be useful to do a long string of operations in FLEX, without typing them in each time. For example, loading a REPORT that requires a HEAD and FOOT can be put into a "DO" file. (See the STD command for an automatic way to find these files). Another application is to store commands for batch runs of FLEX, allowing FLEX to be run through WFL. An example of this is:

```
RUN *OBJECT/FLEX("DO MY/BATCH/COMMANDS ON USR");
```

where FLEX will "DO" the commands stored in the file MY/BATCH/COMMANDS. The FLEX inquiry will abort if there is an error in a "DO" file.

Examples

The 'DO' files of the form *METALOGIC/FLEX/DEFAULTREPORTS/<name>/DO are examples of how complicated reports and series of operations can be simplified by use of the DO command. These "DO" files allow replacement of the PACKDIR report of SYSTEM/FILEDATA, and others. After one of these "DO" files has been DOne, the FLEX inquiry will wait for your command, for example 'FILES :PRINT'. For a quick way of accessing these files, see the STD command.

The next example shows a way to put some of the files in a directory into production. The lines shown could be put into a DO file, which could be used as is or modified if necessary.

```
SELECT GENERATION=0 AND NOT ("INCLUDE" ISIN TITLE OR
    "GENERATOR" ISIN TITLE)
REPORT HEAD "BEGIN JOB PROD;USER=PRIV/PASS;COPY DUMMYFILE"
REPORT FOOT "FROM SYSTEM(PACK) TO SYSTEM(PACK) ; DISPLAY ("COPIED") ;"&
    "END JOB ."
REP " ,",TITLEID," AS *METALOGIC/",FILENAMEID
FILES (META)FLEX/LIBRARY/REPORT :F PRODJOB
WFL STARTJOB PRODJOB
REMOVE PRODJOB
QUIT
```

The DO file above would create a copy job called PRODJOB that would copy files in the directory (META)FLEX/LIBRARY/REPORT to the directory *METALOGIC, e.g. the file (META) FLEX/LIBRARY/REPORT/LFILES would be copied as *METALOGIC/ FLEX/LIBRARY/ REPORT/LFILES.

A DO <file name> command is not permitted with a DO File. However DO <opal string> within a DO file is allowed. The DO <opal string> is only permitted within a DO file. The expected use is to generate flex commands within a REPORT and then be able to execute them.

Opal Compiler

A major aid in developing complex DO files is the ability to check the syntax of the file using their OPAL compiler *SYSTEM/OPAL. C WITH OPAL in Cande or in SYSTEM/EDITOR or Compile Special in Programmer's Workbench with perform a syntax check.

To be able to correctly compile Do files with parameters, it needs examples of valid parameters. These examples are provided in the Parameters heading.

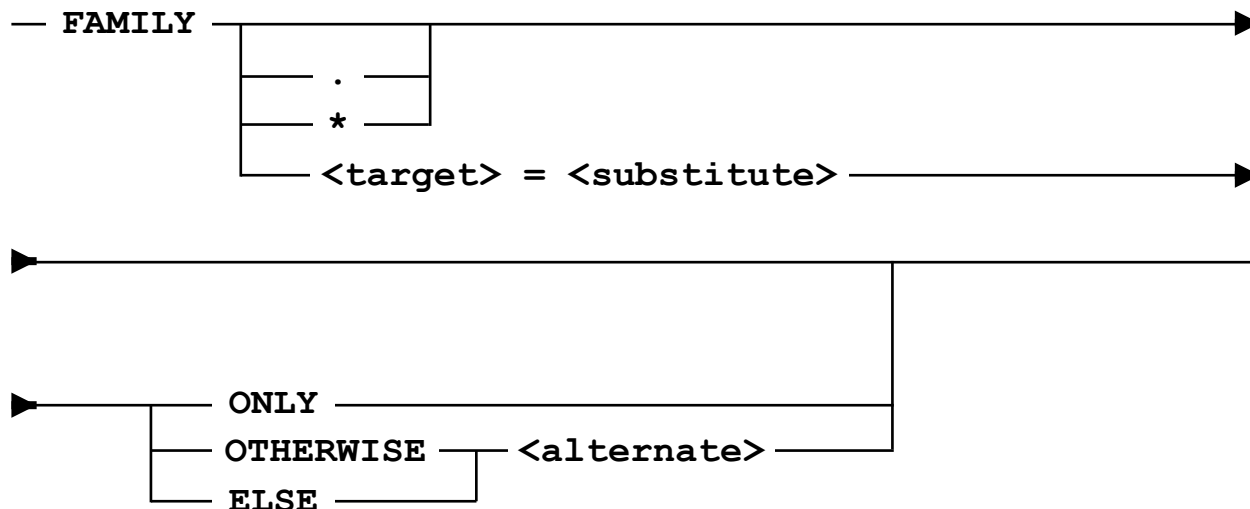
Ex.

```
PARAMETERS (FAM=A,MAXDAYS=10,MAXSEGS)
```

These assigned values will then be used in the compiled script to replace individual parameters. If no default is provided, SYSTEM/OPAL considers the parameter as numeric. Note that, in the current implementation, all default values must be shorter in length than the original parameter name to avoid complex substitution issues.

Flex respects this syntax but note that the supplied values are not used as defaults when the DO file is run.

FAmily command

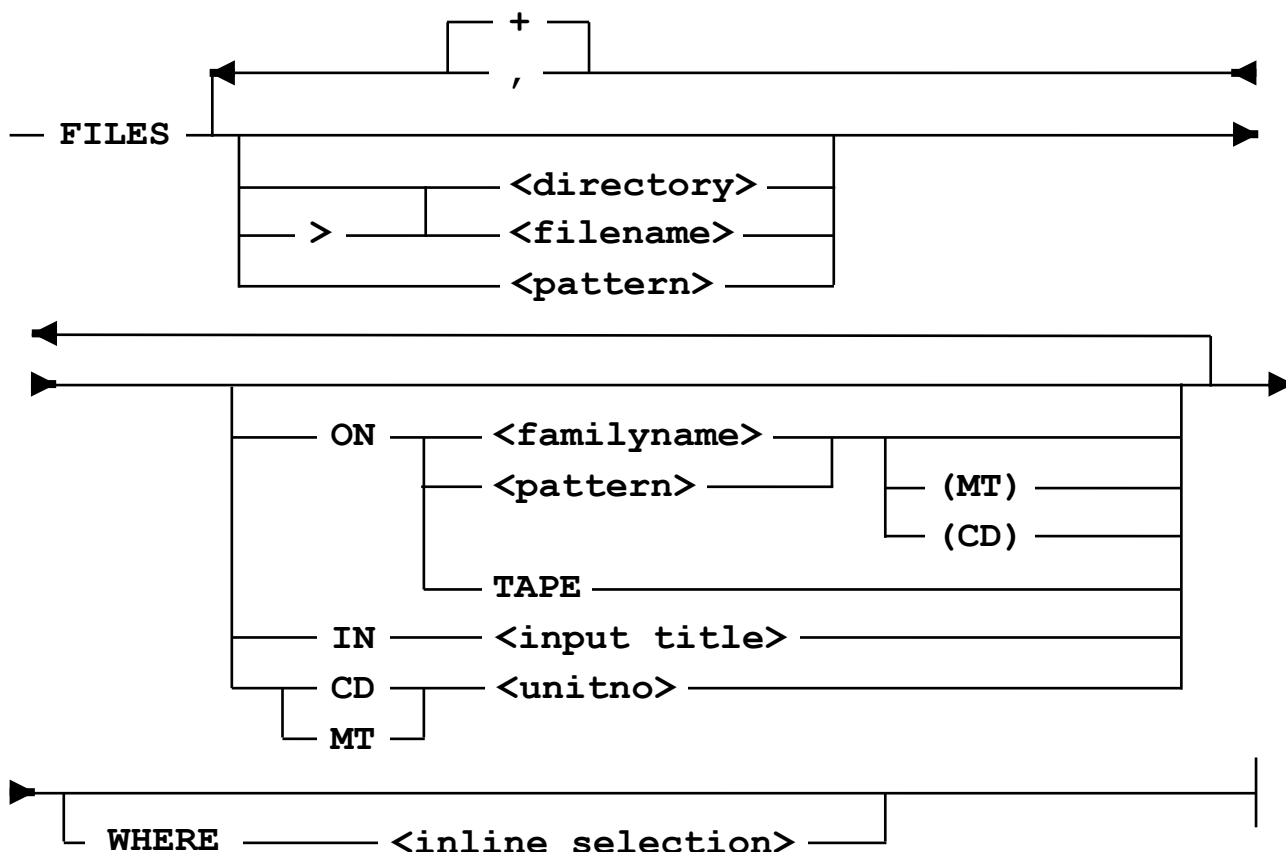


The FAMILY command is similar to the CANDE 'FAMILY' or MARC CHFAM commands. The specified families are used to limit the search for files to these one or two families. Setting FAMILY to '.' means that a family must be supplied with each command. <target family> may be either DISK or PACK. <substitute family> and <alternate family> should be the name of a disk or pack family. An example is:

```
FAMILY DISK=SYSTEM OTHERWISE DATABASES
```

The "*" variant causes the FAMILY to be set to the default FAMILY for the user as specified in the USERDATAFILE. Note that although <target family> cannot be the family TAPE, tape files can be seen by entering "FILES ON TAPE". Changing the current <usercode> with the USER command may change the FAMILY statement.

Files command



<input title> can refer to a LIBMAINTDIR file a Container or a Copywrite disk image.

The FILES command produces a list of files specified by the text following the command. Valid synonyms for FILES are PD, FIL, and FILE.

FILES will list the names of all files in your library on your primary disk family. Use the <pattern> to restrict the search to a subset of the files available to you. For more information, see the section on "[Filenames, Familynames, Directories, Patterns and Wildcards](#)".

If you enter the 'ON <familyname>' part, files on the specified disk <family> will be reported. Otherwise, the report will be for the primary family set by the default for your usercode or by the FAMILY statement of CANDE or FLEX.

SELECT and REPORT specifications previously entered are applied to the FILES command. SELECT restricts what files are selected to report on. REPORT enables you to select the information to report and to format the output as you choose. See the SELECT and REPORT statements for more information.

'FILES >' allows you to specify the starting point in the file directory. The search will start at the first file 'greater than' <filename> or <directory>, where the ordering used is the familiar order in the usual FILES display. The file specified by <filename> does not need to exist.

The WHERE clause can be used to invoke an 'inline selection'. The text following WHERE (<inline selection>) should be valid SELECT code. A WHERE clause will be used in preference to an active SELECT. The where clause can be used after a directory list and before any colon options. Since currently the SECURE command

does not accept a directory list the WHERE clause cannot be used with SECURE. This may be addressed in a subsequent release.

```
Ex FILES TEST WHERE SYMBOLFILE FILES = WHERE AGE=0:Flat  
REMOVE = WHERE CODEFILE and AGE >365 :FL
```

Options

Many of the available <colon option>s may appear after the FILES command. Some of these produce other reports of interest. These are documented here:

- :ATT Show file names and some generally important attributes of the files. (For complete attribute list, use the LFILES command and, for codefiles, the STD CODEATTS report.)
- :CAT Show files and the catalog information including the generations of the files and their backup information
- :ACT Available for Unisys terminals only, this option presents the files in a menu format, and allows the following actions to be entered, file by file : REMOVE, RELEASE, DESTROY, LOAD, MZIP LIST, HELP. See the complete discussion at the end of the OPTION command. The BR may be added to the :ACT option to provide more files per page but with reduced attributes.
- :LINEAR Does a very rapid linear search of the CATALOG. Depending on site restrictions, the LINEAR option may require a privileged usercode. See the complete discussion of LINEAR under the OPTION command. For searching moderate-sized to large directories, this method is approximately 200-1000 times faster than the normal file search method used by the MCP.
- :FLAT Does a rapid linear search of the SYSTEMDIRECTORY. Depending on site restrictions, the FLAT option may require a privileged usercode. See the complete discussion of FLAT under the OPTION command. For searching moderate-sized to large directories, this method is approximately 100-800 times faster than the normal file search method used by the MCP.
- :FAST Allows a very fast search of all families, with the restriction that the only attribute available is the title.
- :FIND Return only files which have a specified string in their content. Access date and time of the searched files are not changed.

Note that many of the options may be used in combination with each other; for example FILES <parameter> :CAT,ATT would show both the catalog information and the attributes of each file found.

Output

The output is directed by default to the requesting station if FLEX is running

interactively, otherwise to the printer file. An interactive user can direct the output to a printer file with the FLEX <print option>. If output is wanted at both the station and the printer file, the option TERM may be set. For example:

```
FILE = : PRINT
FIL OBJECT :PRINT,TERM
```

FLEX can also direct its output to a disk file, by means of the <file option>, for example:

```
F A,B/= :FILE MY/FILES
```

The file written to disk has a filekind of JOBSYMBOL. If no filename is specified, a default title of "JOBOUT" will be used.

Output can be directed to a byte stream file using :BYTE and to a byte stream file on a PC using :REDIRECT.

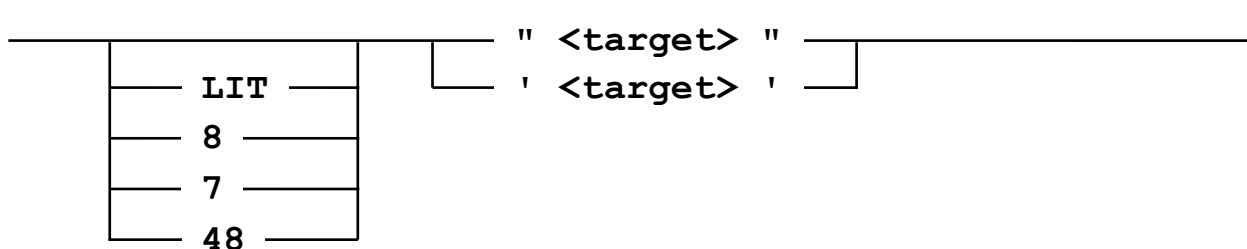
Several options are used to restrict the output of a FILES command. The RESIDENT option will output only those files, which have a resident entry. NONRESIDENT or NR outputs only those files, which have no resident entry. REO (only relevant on cataloging systems) outputs information about only the resident entry – the information returned is a subset of that returned by the RESIDENT option.

A number of options supply additional information to the basic file command. The ATTRIBUTE option shows the attributes of resident file entries. NOTE: attributes of files where the resident entry is not generation zero cannot be obtained from the MCP. The CATALOG option shows information obtained from the catalog block for the file.

Examples

In the examples, we assume that 'SYS' and 'BATCH' are pack or disk families, and that FLEX is running under usercode 'US', with a family statement of FAMILY DISK=SYS ONLY.

| Command | Result |
|--------------------------|--|
| FILES or FILES = | All files for user (US) on SYS (or all files on SYS, if there are none in (US)) |
| FILES X/Y or FILES X/Y/= | All files in <directory> (US)X/Y ON SYS (or if none, then in *X/Y ON SYS) |
| FILES =THE= | All files whose name contains "THE" |
| FILES ?A | Two-character filenames beginning with A |
| FIL (THEM) | All files for user (THEM) ON SYS |
| FILES *= or FILES * | All system files on SYS |

See Also:[<find option>](#)**<find target>**

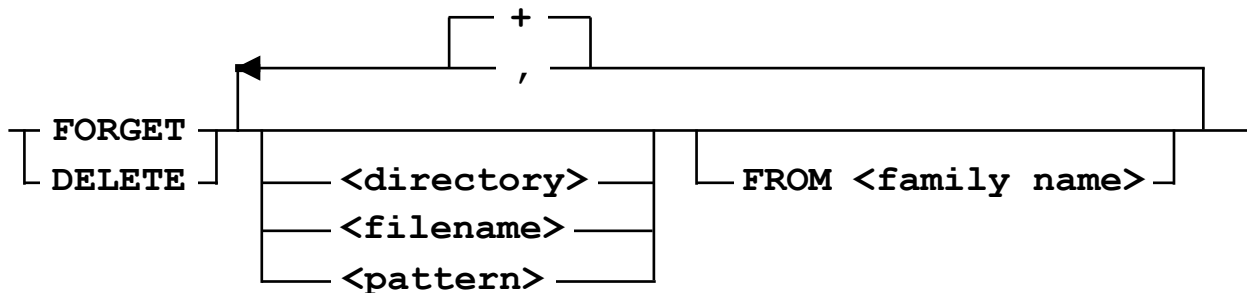
"LIT" and "8" are equivalent, and indicate a literal FIND.

"7" means search for the literal target in ASCII.

"48" means search for the literal target in Hex. Note that the search is still a byte search, so there must be an even number of hexadecimal characters.

If no literal specifier is given, the Find becomes a Whole Word search.

The use of the single quote as the delimiter indicates that the FIND should treat case differently. For ASCII it means search for upper case only. For other searches it means ignore case in making a match.

FORGet command

The FORGET command allows you to CATALOG DELETE generations of a file by offering a screen showing the generations of a file and its backups. To delete a generation, go to the box containing the information on the generation you wish to DELETE, and press SPCFY. The corresponding entry of the file will be CATALOG DELETEed, and the screen will then be shown again to allow further deletions. To go to the next file, press SPCFY on the top line. To end this command before all the files have been processed, enter 'BRK' on the top line, then transmit.

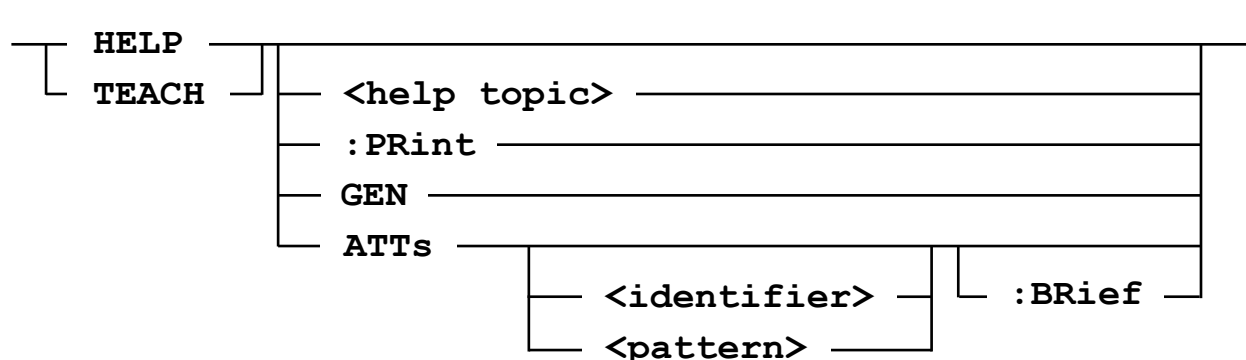
Selection expressions, reports, report headings, and report footings are ignored by the FORGET command.

When a generation of a file is Deleted, all references to its backups are lost. As with the DESTROY command, there is no effect on the backup tapes themselves; only the catalog information is lost. The resident entry of a file may be Forgotten, thereby forgetting backup information, without removing the resident entry or affecting the other entries.

See also:

DESTROY command [WFL command](#)

HELP command



The HELP command provides an interactive way for the user to get information on how to run the FLEX inquiry. If a <help topic> is specified, the HELP page for that topic will be shown; if no topic is given, FLEX will show the index page. To get a printout of the textbook, enter 'HELP :PRINT'.

For a full printout of the attributes available in FLEX, enter 'HELP ATTS' with no qualifier. If HELP ATTS is followed by an <identifier> FLEX will look for the identifier in the attribute list and return the information about it. If a pattern is given, all attributes which match the pattern will be returned. For some cases, e.g. HELP ATT P=, this can be a long list. In that case using the option :BRIEF to return the attributes names and contexts in a more concise table format.

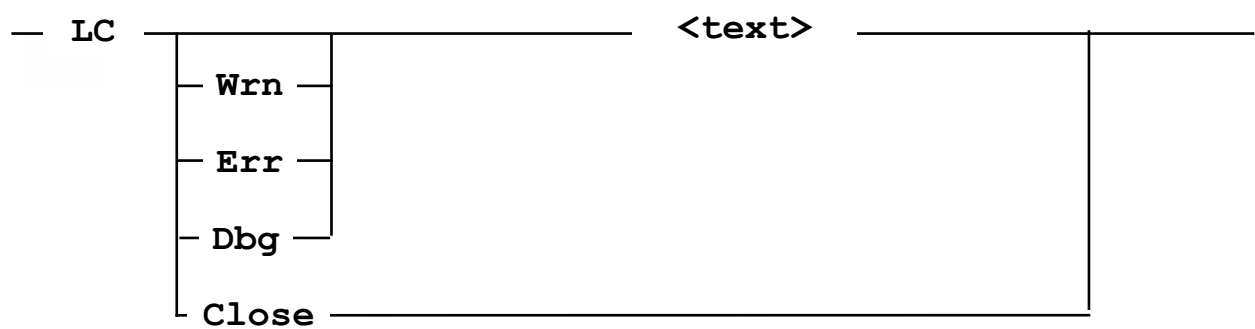
'HELP NEWS' can be used to inform users of new features or enhancements in the FLEX package. Entries appear most recent to oldest.

The 'HELP GEN' syntax is used only to generate a look up table for the HELP textbook. It is normally used only by METALOGIC personnel.

INSTALL command (Full Flex only)

This command is described in the FULL FLEX reference manual

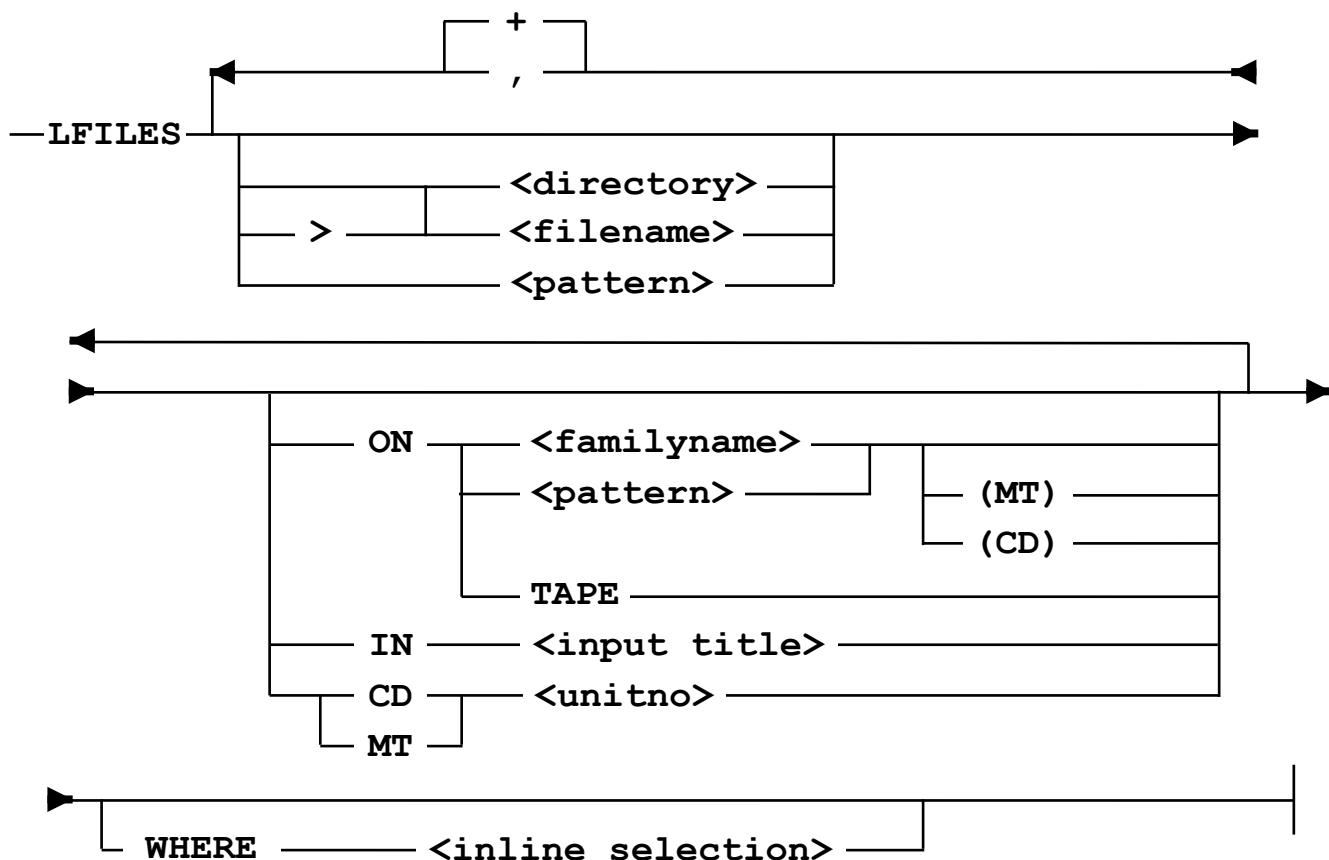
LC command



Adds an entry to the Flex Log. The optional modifiers , apart from Close, are case sensitive and will create a log entry with that category. If omitted the category will be Lgc.

The special modifier Close will close the current Flex log and start a new one.

LFiles command



<input title> can refer to a LIBMAINTDIR file a Container or a Copywrite disk image.

LFILES shows the <file attributes> meaningful for the the resident files specified in the parameter. The report is formatted to group together certain classes of attributes. These give the size, blocking and physical structure, general catalog information, and security attributes.

The WHERE clause can be used to invoke an 'inline selection. The text following WHERE (<inline selection>) should be valid SELECT code. A WHERE clause will be used in preference to an active SELECT. The where clause can be used after a directory list and before any colon options. Since currently the SECURE command does not accept a directory list the WHERE clause cannot be used with SECURE. This may be addressed in a subsequent release.

```
Ex FILES TEST WHERE SYMBOLFILE FILES = WHERE AGE=0:Flat
REMOVE = WHERE CODEFILE and AGE >365 :FL
```

A similar report is given by using the FILES:ATT variant of the files command

Example:FILES *metallogic/flex/library:ATT

```
files *metallogic/flex/library:att
*METALOGIC/FLEX/LIBRARY 530:16 DCALGOLCODE (8118)
Created 16:20:43,27/01/09 GMT Security PUBLIC IN Extmode SINGLE
Altered 16:20:43,27/01/09 GMT FilStr ALIGNED180 EXP Rec/Blksz 30/540 W
Used 16:40:09,27/01/09 GMT BlkStr FIXED Length 243330 W
CopiedTo 09:53:08,30/01/09 GMT LastRecord 8110 Rows 17*504 S
ReleaseId 530.16
#Found 1 file [from 1 file]
#Directory: *METALOGIC/FLEX/LIBRARY ON DEV
```

A similar display is available with special attributes through used of the CODEATTS STD report.

Example:Entering "FILES *metallogic/flex/library after STD CODEATTS" yields:

| Title : *METALOGIC/FLEX/LIBRARY | | | |
|---------------------------------|--|--------------------------|------------|
| General attributes | | Codefile info attributes | |
| Filekind : DCALGOLCODE | | Executions: 0 | |
| Segments : 8118 | | Priority : 0 | |
| Codeversion: 53.189 | | Parameters: 0 | |
| Cycle/version: 530:0016 | | Restarts : 0 | |
| Created 27/01/09 @ 16:20:43 | | Nonexecutable: FALSE | |
| Accessd 27/01/09 @ 16:40:09 | | Systemfile: FALSE | |
| Altered 27/01/09 @ 16:20:43 | | Language : DCALGOL | |
| Changed 27/01/09 @ 16:20:54 | | Sharing : SHARED BYALL | |
| Security Information | | Capable? | Limits |
| Security type : PUBLIC | | DMS: N | MaxIO: 0 |
| Security use : IN | | IPC: Y | MaxPROC: 0 |
| Protection : SAVE | | SORT: N | WSlimit: 0 |
| Guardfileset : FALSE | | LIBR: Y | WSavg : 0 |
| Security admin : TRUE | | | |
| Secadmin transp: FALSE | | | |
| | | Stack size info | |
| | | D1stack: | 683 |
| | | D1WSavg: | 0 |
| | | D2stack: | 3072 |

For information on the meaning of the attributes shown, see the Unisys I/O Subsystem Reference manual, or the printout produced by entering 'HELP ATTS' .

Example: Entering "LFILES *metallogic/flex/library" yields:

```

lfiles *metallogic/flex/library
*METALOGIC/FLEX/LIBRARY 530:16 DCALGOLCODE (8118 Segs 8111 Recs 243330+210 Wds)
      Extmode      SINGLE Security      PUBLIC IN
Created   16:20:43,27/01/09 GMT UserInfo      0      Owner      *:RWX
Copied On 16:20:43,27/01/09 GMT BlkStr      FIXED Group      <NONE>:NO
Att Mod. 16:20:54,27/01/09 GMT Mode/FrameSz 48/48 b Other      :RX
Altered   16:20:43,27/01/09 GMT Recsz      30 W FilStr      ALIGNED180 EXP
Used      16:40:09,27/01/09 GMT Blksz      540 W FilOrg      NOTRESTRICTED
Read      16:40:09,27/01/09 GMT      Opt      <NONE>
Backed Up 16:20:43,27/01/09 GMT Rowsz      504 S
Copied To 09:53:08,30/01/09 GMT Rows      17      Flag      LIB(BYALL)
$ Version 53.530.0048      CU Est      115922
MP +PU,+SECADMIN,+TASKING,      D1/D2 Stk 683/3072
MP +IDENTITY FLEXLIB
ReleaseId 530.16      Product METALOGIC 2295
Saved To   30 < 26/02/09
#Found 1 file [from 1 file]
#Directory: *METALOGIC/FLEX/LIBRARY ON DEV

```

```

lfiles description/acdb
DESCRIPTION/ACDB 1:0 DASDLDATA (108 Segs 12 Recs 3072+168 Wds)
TimeStamp 08:13:01,28/03/08 Extmode EBCDIC Security PRIVATE IO
Created 08:13:00,28/03/08 BST UserInfo 0 Owner BOB:RWX
Copied On 08:13:00,28/03/08 BST BlkStr FIXED Group <NONE>:NO
Att Mod. 08:13:00,28/03/08 BST Mode/FrameSz 8/48 b Other :NO
Altered 08:13:00,28/03/08 BST Recsz 256 W FilStr ALIGNED180 IMP
Used 13:58:15,14/10/08 BST Blksz 256+14 W FilOrg NOTRESTRICTED
Read 13:58:15,14/10/08 BST Opt <NONE>
Backed Up 08:13:00,28/03/08 BST Rowsz 720 S
Copied To 12:28:58,06/02/09 GMT Rows 1
Saved To 999 < 22/12/10
#Selected 1 file [from 1 file]
#Directory: DESCRIPTION/ACDB ON DEV

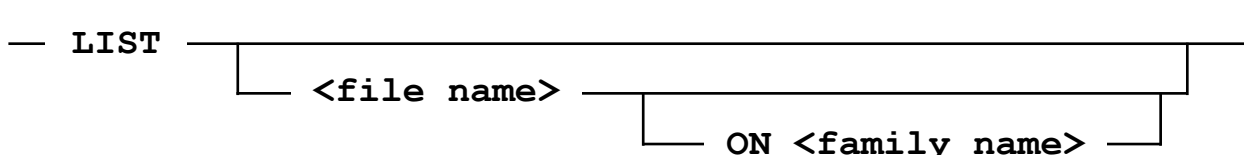
```

Total wastage do to a file not being crunched, badly aligned blocks or the last block not being full are reported after the + sign on the second line. (168 words in the file above). Wastage per block due to badly aligned blocks is reported after a + sign with the BlkSz entry. (14 Words for the file above)

See also:

```
<print option> FILES command <filename> <directory>  
quotes <usercode>
```

List command



LIST provides a basic way to list a symbolic file through FLEX. If no <filename> is given, FLEX will list the current CANDE workfile, if there is one.

Several output options affect the output of LIST, including the <unseq option>, <print option>, and the <hex option>. To send the output to a printer, use the <print option>.

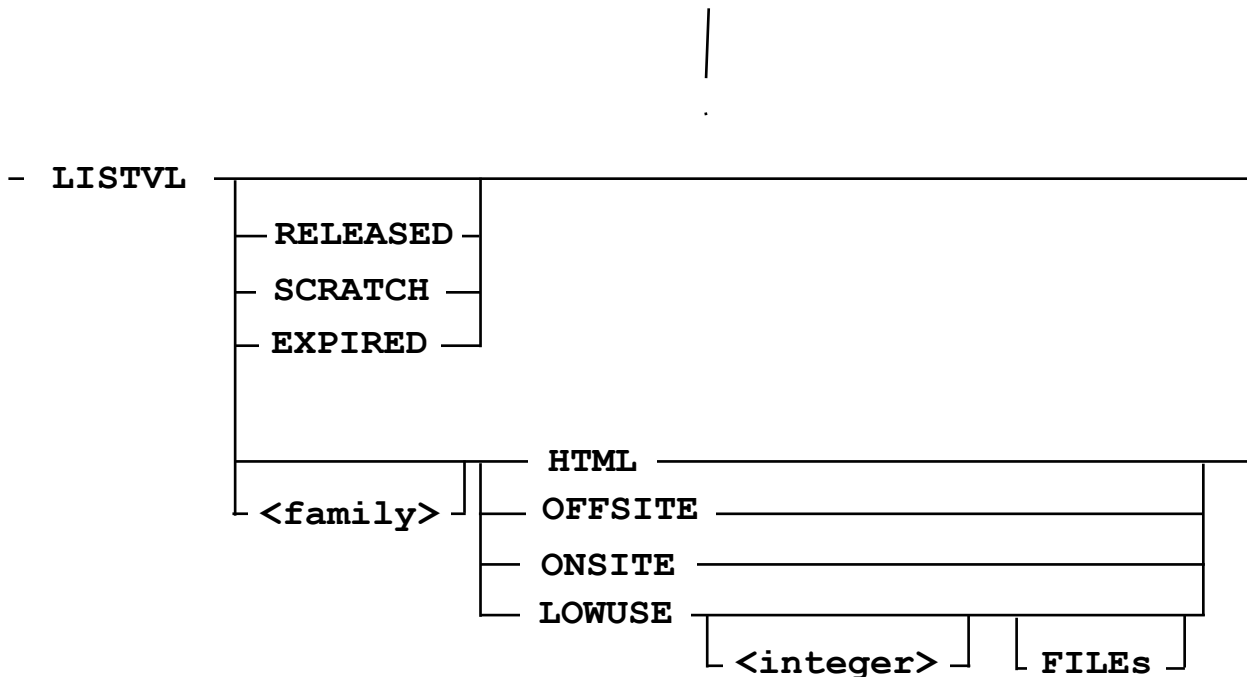
The LIST command may also be entered from the action screen.

See the <action option> for a discussion of the action screen.

See also:

[<unseq option>](#) [<print option>](#) [<hex option>](#) [<action option>](#)
[WRITE command](#)

LISTVL command



The LISTVL command which is only available on cataloging systems -- generates a variety of reports for printing volume library information. Many additional capabilities are provided above and beyond those of SYSTEM/LISTVOLUMELIB. The first list shows all of the volumes in the Volume Library, sorted by serial number, together with all of the information obtainable by PV requests on a single line. In addition, the number of files for which each volume is in use as backup is given. The second report shows them sorted by name. A list of scratch entries is given. Volumes that are in use as backup, but not in the volume library, are listed. A list of expired volumes that are not in use as backup is given, and a page of statistics on the catalog is given. In addition, short reports may be generated that show only expired volumes, or only tapes that are used as backup for a small (specifiable) number of files. All of these reports can be run against an old CATALOG.

The LISTVL command prints a report of all of the volumes in the VOLUME LIBRARY, and all of the volumes in use as backup on the information is reported for each volume: There is also a list sorted by volume name. The following information is reported for each volume :

- The SERIALNO, volume name, KIND, creation date, savefactor, and the structure of the volume family.
- Whether or not the volume is in use as a backup.

A summary is provided at the end showing the total number of files and file entries on the system, the number of volumes in the VOLUME LIBRARY, the number of volumes in use as backup, and run statistics.

Variants

If no modifier appears after LISTVL, all of the volumes in the volume library will be shown. If a <family> appears, the output will be restricted to those volumes which are in use as backup for the given <family>. If LOWUSE <number> appears, only the volumes which are in use as backup for the given <family> which are backup for <number> files or less are shown. Only a privileged user may use this command.

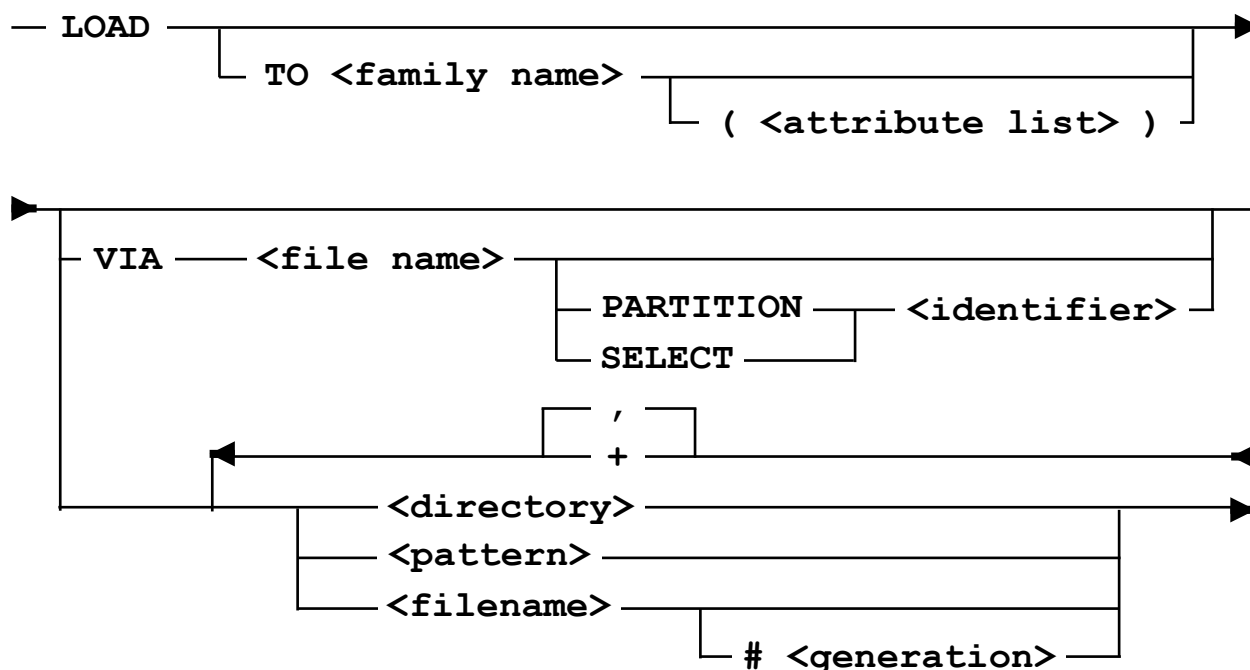
If the HTML modifier is used the report is generated as an HTML table which can be sorted by clicking on a column heading. Only volumes in use as backup are reported.

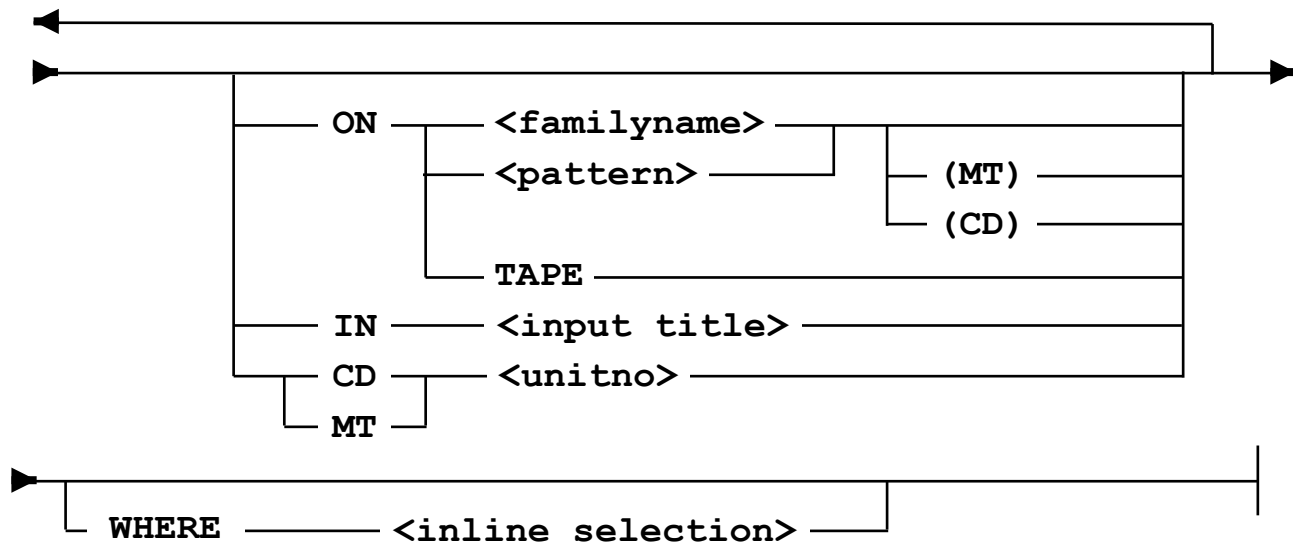
If the Metalogic mail library is available and the UserData entry for the user running flex has an email specification, the report will be emailed as an attachment.

Example

```
LISTVL HTML
LISTVL DEV HTML
```

LOAD command





<input title> can refer to a LIBMAINTDIR file a Container or a Copywrite disk image.

The LOAD command is used to retrieve files from backup.

It behaves in two different ways.

The WHERE clause can be used to invoke an 'inline selection'. The text following WHERE (<inline selection>) should be valid SELECT code. A WHERE clause will be used in preference to an active SELECT. The where clause can be used after a directory list and before any colon options. Since currently the SECURE command does not accept a directory list the WHERE clause cannot be used with SECURE. This may be addressed in a subsequent release.

From Source

If MT,CD, or IN <input title> are specified then files are loaded from the specified source.

Via Catalog or Archive

The (Catalog or Archive) LOAD command will only attempt to load files that are NOT currently resident on disk (i.e. [NR] files). A minimal set of backup volumes will be chosen from the backups for the files specified in the parameter, and one or more load jobs will be passed to the system. If the LOAD cannot retrieve the file using one of the backups, a new job will automatically attempt to LOAD the file from the second backup, if possible.

LOAD will use a LIBMAINTDIR if available. The info in the LIBMAINTDIR will override the FLEX Verify option for the COPY. Also if the tape is not one made by FLEX, or if it has been appended to, FLEX will check where the correct files are, and, if necessary, use the '#<file no.>' syntax to access them.

For maximum efficiency, all of the <file>s and <directories> that need to be loaded should be specified in one LOAD statement, since FLEX can optimise the number of tapes required. Note that if a <directory> is specified, if there is a file of the same name as the <directory>, it will also be loaded, e.g. if doing LOAD ABC/=, both the file ABC and the <directory> ABC will be loaded.

Load TO

The 'TO <familyname>' modifier is used to LOAD files to a family different than the original family from which the file was backed up. An attribute list may appear after the Familyname e.g. LOAD TO A(KIND=CD) X/Y

The list has to be a valid WFL <destination volume attribute list> in a COPY statement. Because of the possibility of ambiguity (e.g. is "KIND=CD" as usercode pattern?), an optional comma can be placed after the ")", e.g. LOAD TO Y(TAPE), X

If there are multiple source volumes, the list will be added to the ADD destination in each job.

Non zero generations

If a generation of a file other than zero needs to be loaded, the # <generation> syntax should be used. Valid values of <generation> are 0-6, where 0 is the default. Only a privileged user may specify <generation> for a <directory>.

The :Suffix option may be used to change the title of the loaded file.

LOAD <filename> GENERATION= <n> :Suffix will load the requested file adding a suffix of the form yyyymmdd_hhmmss_SV.

yyymmdd_hhmmss is formed from the timestamp of the file.

Load Via

The 'LOAD VIA' form of the LOAD command allows the LOADING of a list of files, whose names are stored in a file.

The file used with the 'LOAD VIA' syntax must consist of a list of titles terminated by periods. The file can be grouped into sub lists , prefixed by a record with a '%' in column 1, followed by an identifier. The PARTITION or SELECT option allows these sub lists to be loaded. If no selection option appears, all files (all sub lists) will be LOAD'ed.

Note that lists of this type can be made by using the following report in FLEX inquiry:

```
REPORT TITLE, ". "
```

Although specifying generation is not allowed in the list of file, there is a work around.

The USERINFO of the file may be used to specify generation.

```
Ex
WFL MODIFY LOAD/LIST (USERINFO=6)
followed by
LOAD VIA LOAD/LIST:SUFFIX
```

Would load generation 6 of each file in LOAD/LIST adding a timestamp based suffix

Compare two generations

A practical way to compare is to combine the 'TO <familyname>' and '#<generation>' variants. For example, suppose we wanted to match the current version of a program called PROG ON SYSTEM with generation 2 of the same program. By entering

```
LOAD TO BATCH PROG #2
```

FLEX will load generation 2 of PROG to BATCH pack, and an ordinary CANDE match can be done.

The queue where the load jobs are placed is specified by the site manager using the DEFAULT command.

Lost Family Index.

The Load command may be used to reload any files which have rows on a member of the family which has failed.

To recover all recoverable files on a pack where index 2 has crashed:

```
Load (*)= on FAM:FL #2
```

Unlike other loads, a :Flat Load without a 'TO <dest>' part, will overwrite the resident file, so a confirmation is asked for.

Examples

In the examples below, we assume that SYSTEM and BATCH are diskpack families, and that the family statement for the user is FAMILY DISK=SYSTEM ONLY.

| | |
|----------------------------|--|
| LOAD XYZ/= | load files in <directory> XYZ to SYSTEM pack |
| LOAD PROG #3 | load generation 3 of PROG to SYSTEM |
| LOAD A,B/=,C/= | load A,B,B/=,C,C/= to SYSTEM pack |
| LOAD VIA (PRIV)DISCARDLIST | load files listed in DISCARDLIST file |
| LOAD TO BATCH XYZ | load XYZ to BATCH pack |

See also:

<filename>_____<directory>_____quotes_____<usercode>

LOG Command

As with other Metalogic software, logging has been implemented for the FLEX suite. Most messages, errors, commands and warnings are now unconditionally logged into a file called *METALOGIC/FLEX/LOG on a nominated family configured using the INSTALL utility. By default, this family is set to that of DL LOG. and all FLEX software uses this system.

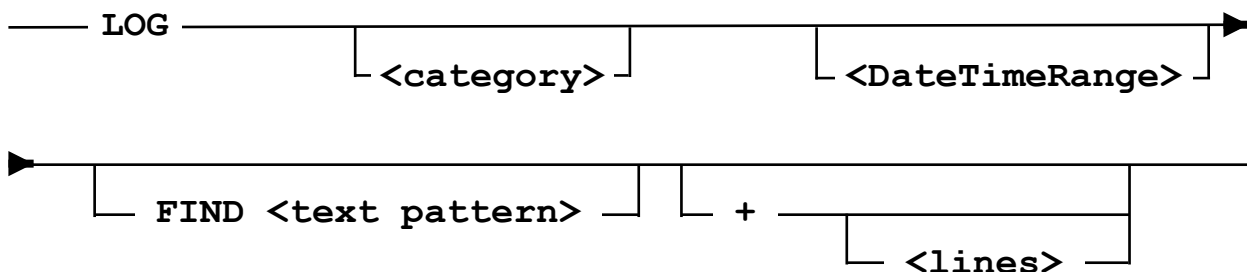
Logging is performed for all FLEX runs though only non-interactive batch (DO) runs are more complete. Errors and warnings for the file system in particular are now

retained in these logs. Where files are removed, released or destroyed by a run of FLEX Utility then these entries will be logged too.

```
11:23:20 Job:Flex(27000):BOJ Flex UTILITY 54.999.32 running
11:22:17 Job:Flex(26990):EOJ Flex UTILITY run complete
11:22:11 Cmd:Flex(26990):DO XFER/FILECOPY
11:22:11 Msg:Flex(26990):User=IPP,Stn=174,Cmd=DO XFER/FILECOPY
11:22:11 Job:Flex(26990):BOJ Flex UTILITY 54.999.32 running
```

The LOG command syntax is identical to that provided for SUPERVISOR, MAILLIB and TRIM; full details on the implementation and syntax can be seen by using the HELP LOG command.

The LOG command syntax is described below.



All Flex log entries are identified by a <Category> field which consists of a 3-character text string and can be used to filter any any log search by message type.

<category>

| | |
|-----|--|
| ALL | returns last 500 entries in one or more logs |
| CMD | selected Flex commands eg WFL, REMOVE, DO... |
| ERR | returns all error messages |
| LGC | returns all LC (Log Comment) entries |
| MSG | returns all solicited and unsolicited messages |
| FLT | returns all internal faults |
| JOB | BOT or EOT of Flex tasks |
| WRN | returns all warning messages |

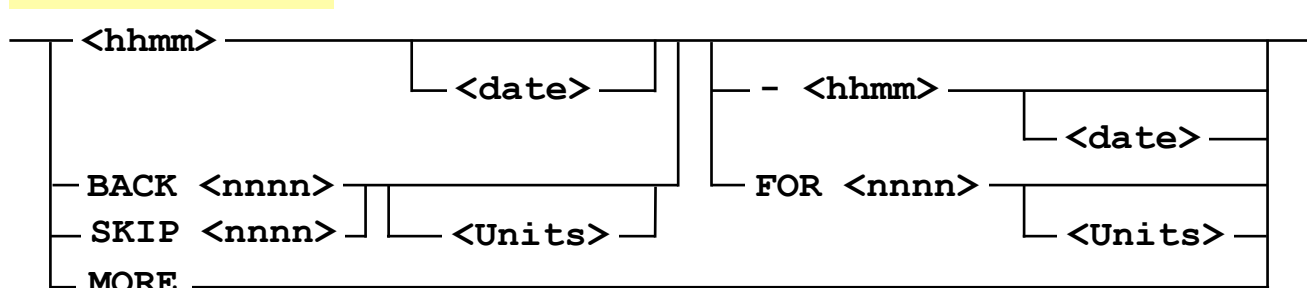
By default, FLEX will only show one page of information for each of the above commands (except for ALL) but note this may not always be complete if FLEX fails to find enough entries. Only the current FLEX log file is searched.

A MAGUS configuration variable called SYS_LOGALL has been implemented to support a customer enhancement. This variable, supported by INSTALL's SYSTEM CONFIG menu, expects a usercode list (separated by commas) that is checked

each time a LOG command is used.

If the logged-on usercode is in the list, Flex appends '+' to all LOG commands (unless '+' is already present) so that each command can return more than one page of information.

<DateTimeRange>



The <date> format depends on the global setting of the Metalogic configuration variable SYS_USDATES which can be set with the INSTALL utility. If SYS_USDATES is set to TRUE then <date> has the form mm/dd/yy, otherwise dd/mm/yy.

The MORE modifier set the Stop date and time to be the date and time of the last log record reported in the most recent log command.

The BACK and SKIP modifiers respectively allow specific start and end time points to be more easily assigned without having to use explicit values. BACK allows a simple log start time to be assigned whereas SKIP assigns a log End time. The FOR specifies an optional duration associated with the Start or End time. If FOR is omitted then both BACK and SKIP will assign default search range periods of 24 hours.

<Units>



By default, MINUTES will be used if no <Units> is given. A minimum of one character may be used to specify any <Units> type. BACK and FOR may be used in any combination with normal date and times.

If the '+' modifier is used, an optional count may be used to control the number of response lines returned to the caller. This modifier has a default of 500 lines and should be used with care if a filter is used with many logs present.

'+' cannot be used as part of a <Text Pattern>

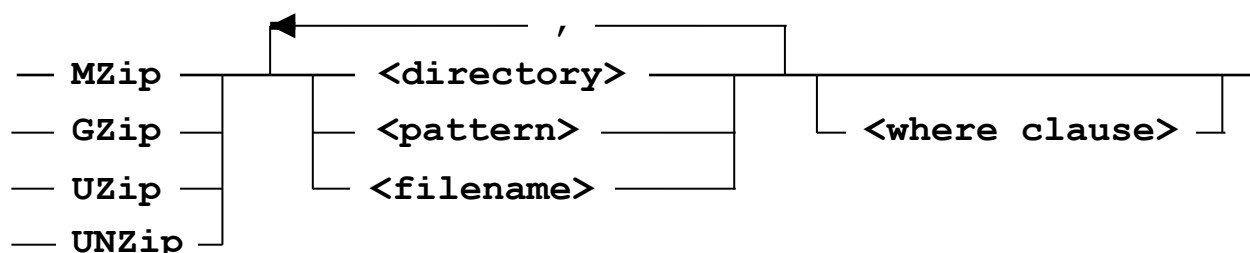
To search for the literal characters # @ & = ~ they should be prefixed by the \ character. Ex. LOG FIND \= to find the literal "=" in a log entry.

If a <DateTimeRange> is specified or the '+' modifier is used in the command text,

then all available log files will be searched until a maximum of 500 lines of text has been returned.

The FIND modifier permits the searching of each log entry for the specified text. The FIND automatically encloses the target with '=' at both ends of the text and is NOT case-sensitive, therefore finding both lower-case and upper-case matches. FIND must be used after all other modifiers as spaces are permitted with in the FIND text. The log files are automatically released when the file size exceeds 15,000 records or the LC CLOSE command is used.

MZIP Commands



These commands access a fast compression function, using the LZ77 algorithm. MZIP compresses files and their attributes in a format suited to the MCP system. GZIP compresses the data in byte stream files (and loses the file attributes), but the resulting file can be expanded using gunzip on almost any other system. UNZIP (or UZ) expands a file compressed by MZIP or GZIP. But, as yet (07/2010), it cannot expand most files created by gzip on other systems.

To get the output file name, add /MZ for MZIP, ".gz" for GZIP, and /UZ for UZIP. However, if the file already ends in ".gz" or /MZ, MZIP and GZIP will not process that file, while UNZIP will strip off the ending. If the output file title already belongs to a resident file, the action is skipped, unless the AutoRm option is set, when the resident file is replaced.

These commands can also be Actions in :ACT commands.

MZIP and GZIP have 2 specific options to control how they work. ZipMore or ZM controls how much effort goes into the compression in MZip and GZip. If it is OFF, the compression is done with as little CPU time as possible. If ON, about 50% more CPU is used to improve the compression, typically ending with 10% smaller compressed file size. ZM ON files normally UNZIP more quickly than ZM OFF ones.

ZM can also take an integer value, though this is not saved between sessions. The only current assigned meaning is for ZM 2 in MZip. This value will often add a further 50% CPU time over ZM 1, and drop the final compressed file size a further 7%.

All these % figures are data dependent and each file will behave somewhat differently. However, a file with ZM 2 compression will normally need about double the time to UNZIP, vs ZM 1.

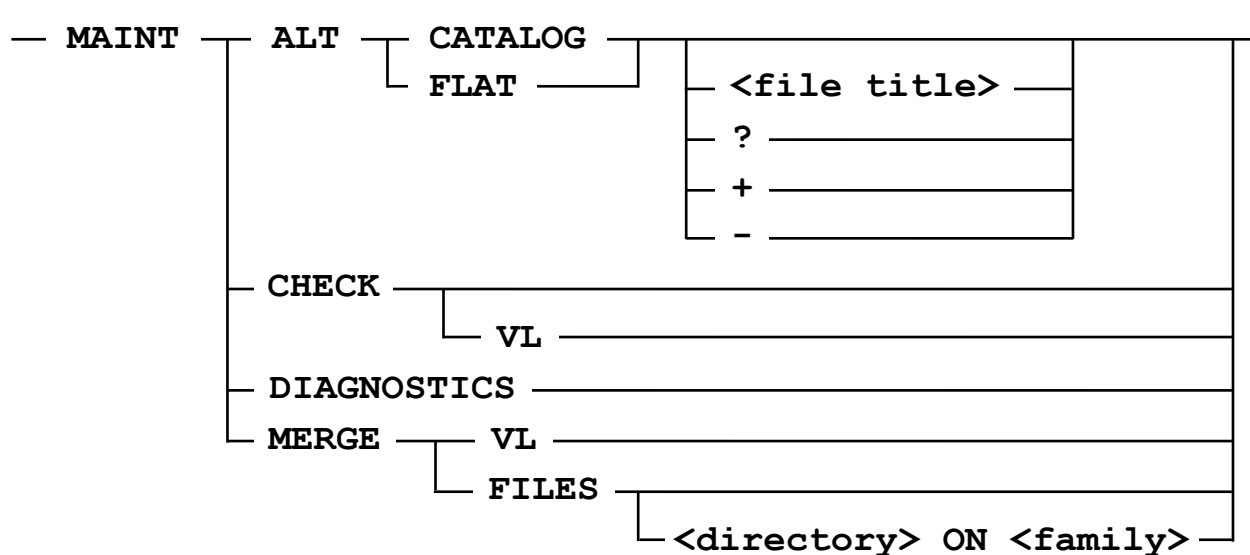
The MemSz option is essentially a debug option to override the memory size used by MZIP and GZIP.

It is normally (Default), in which case MZIP/GZIP look at the system to determine

how much memory to use, but can be set to (Big), (Med), and (Small) corresponding to 16MW, 1 MW, and 64kW respectively. Add at least 25% if ZipMore in set. MemSz is not kept between sessions.

The XLATE option in MZIP and GZIP, it does an EBCDIC to ASCII translation, and in UNZIP, an ASCII to EBCDIC one.

MAINT command



The MAINT command is for configuration and maintenance of the system directories. All variants apart from ALT FLAT require a cataloging system.

The ALT CATALOG syntax is used to change the SYSTEM/CATALOG that is used for file search using the <linear option>, or the PDT BU, MAINT, or LISTVL commands. The file specified by <file title> must be a current or previous SYSTEM/CATALOG. The +/- syntax allows enabling or disabling use of this alternate CATALOG, without changing the <file title> originally specified.

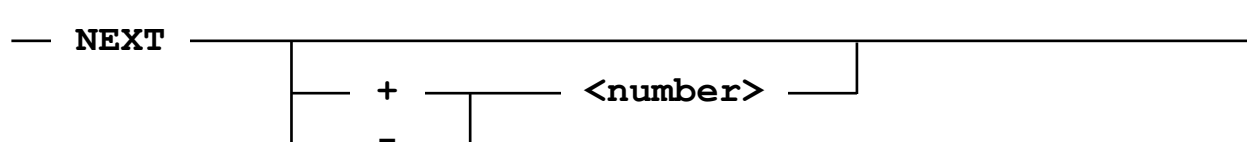
The ALT FLAT syntax is used to change the SYSTEMDIRECTORY that is used for file search using the <FLAT option>. The file specified by <file title> must be a current or previous SYSTEMDIRECTORY, or a file made using the directory copy function of the GETSTATUS DCALGOL intrinsic. As with the Catalog, the +/- syntax allows enabling or disabling use of this alternate SYSTEMDIRECTORY, without changing the <file title> originally specified.

The MERGE VL syntax allows the Volume Library (VL) to be moved from one catalog to another.

MERGE FILES allows files to be merged from one family to another, with their backup references intact. The source family is in the Catalog specified by the ALT CATALOG variant, and the target is the prime family from the FAMILY command.

The DIAGNOSTICS and CHECK variants are to look for problems in the Catalog. For further information on them, and how they and MERGE can be used to enhance recovery strategies, file placements and pack configurations, see the FLEX Site Management Manual.

NExT command



The NEXT command is used in conjunction with the FILES command to allow skipping back and forth among the pages of files. This command may be entered when waiting at a page break during a FILES command. The NEXT command may be used in conjunction with the FILES > syntax.

The + and - versions will skip to succeeding/preceding pages in the FILES display, for example NEXT + will go to the next page of files; NEXT-4 will go back four pages of files. If the number is too large, FLEX will pick an appropriate maximum and use that, e.g. NEXT - 777 will go to the first page of files. Note that the SAME command is equivalent to NEXT+0 and NEXT-0.

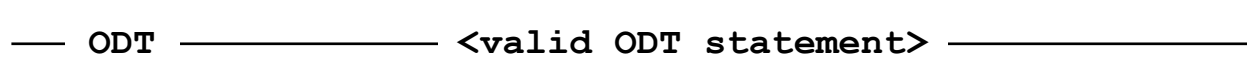
Examples :

| | |
|--------------------------|--------------------------------------|
| NEXT+ or NEXT or <SPCFY> | skip to the next page of files |
| NEXT- or NEXT-1 | go back to the previous page |
| NEXT+ 0 or NEXT- 0 | re-display the current page of files |
| NEXT-2 | go back 2 pages of files |

See also:

[<FILES command>](#) [<SAME command>](#)

ODT command

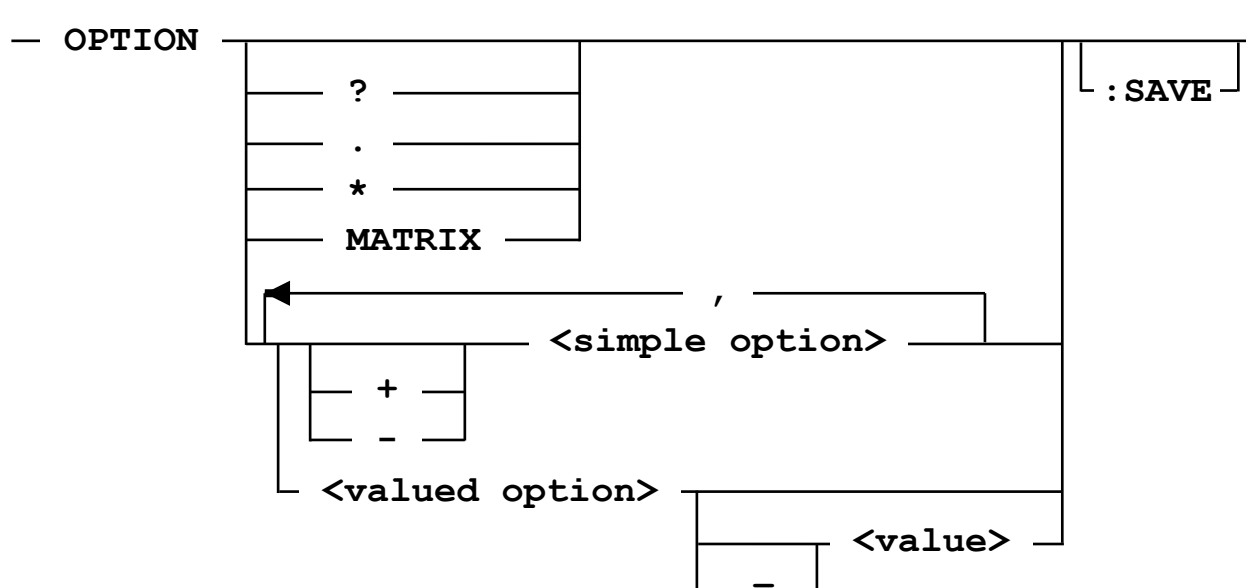


The ODT command allows privileged users the ability to enter ODT commands. It is intended for use in batch mode, where it may be desirable to see and record the running environment during a run. The parameter is passed to the system via DCKEYIN, and system responses are returned.

This command requires a privileged usercode.

This should not be confused with the ODT statement which may be used in an [<opal block>](#).

OPTION command



The `OPTION` command is used to set and interrogate the values of the permanent options of the FLEX inquiry. For a summary, enter 'HELP OPTIONS' in FLEX inquiry. 'OP' or 'OP ?' will show a page of the available permanent options and their current values. These options are those that are meaningful throughout a FLEX session, and may be saved in the file `FLEXOPTIONSFORME` by using the `<save option>`. The minimum abbreviation for an `<option>` is indicated in capital letters; only the first six characters are examined.

'OP *' will return your options to the values saved in your file `FLEXOPTIONSFORME`. 'OP .' sets your options to suitable defaults, chosen for your site by the site manager.

When setting or resetting an `<option>`, if an option in the option list is not preceded by a '+' or a '-', then '+' is assumed.

If the `<save option>` is specified, the current settings of the permanent options will be saved in `FLEXOPTIONSFORME` under your usercode, and are kept for each terminal that runs FLEX. The `<save option>` is the only `<colon option>` that can appear after the 'OPTION' command.

Most of the commands to FLEX may be followed by a list of options, separated by commas, and found at the end of the command after a colon, like `CANDE`.

Options that appear after a colon at the end of a command are considered temporary, and will return to their default value at the termination of that command. To set an option permanently, use the 'OP' command.

A `<simple option>` is one of :CAPS, SPEAK, CATALOG, RESIDENT, NONRES, ATT, REONLY, CANDE, PRINT, TERM, UNSEQ, TRUNCATE, HEX, BRIEF, AUDIT, MONITOR, LINEAR, FLAT, FAST, UNSORT, USDATES, NOMSGS, OLDWFL, SYNTAX, TWOBACKUPS, SETLIST, SETCODE, or INTER. A `<valued option>` is one that may take a value or can be used as a `<simple option>`; these are `DEBUG` and `WAIT`. The `<file option>` allows the output to be directed to a file, given by `<file name>`. The file specified must not already exist. If no `<file name>` is given, an

appropriate default will be selected.

Available options & descriptions

In the following table the part of the action in uppercase is the minimum abbreviation which can be used. So :ACT is equivalent to :ACTI or :ACTIO or:ACTION.

| Option | Brief description |
|------------|---|
| ACTion | Present a menu of files and allow various actions on them |
| APpend | Append output from commands to a file. |
| ARChive | Use Archive information on a Files command. |
| ATtributes | Show file attributes on a FILES command. |
| AUDit | All input and output to FLEX inquiry will also be printed. |
| AUTorm | Allow file output to overwrite an existing file. (Auto Remove) |
| BRIEF | Suppress or shorten certain messages and reports |
| Byte | Send output to a byte stream file. |
| CANDe | REMOVES, DESTROYs, RELEASEs also affect OBJECT file; also changes the FILES display |
| CATalog | Show catalog information on a FILES command |
| COPYWrite | Makes BACKUP use Copywrite instead of Library/Maintenance |
| DEBUG | Produces diagnostic output for debugging FLEX |
| FAst | Fast search of whole File Access Structure. (Restricted) |
| File | Send output from commands to a job symbol file. |
| FLat | Fast search of whole SYSTEMDIRECTORY |
| Xlate | Show EBCDIC and hexadecimal characters on LIST or WRITE |
| INTER | Used as a colon option when passing commands in the Run of FLEX to keep FLEX running after executing the command. |
| LEVEL | Modify a file search or USAGE to N levels of file name. DEPTH is a synonym for LEVEL. |
| LINear | Fast search of whole CATALOG (restricted) |
| MONitor | FLEX will send messages showing progress during a search |
| NOMSGS | Suppress messages from job started by FLEX inquiry |
| NR | Show all entries of files with no RESIDENT entry ([NR]) Listed on the Options screen under SELRES |
| Print | Send output from commands to the line printer |
| REdirect | Send output from commands to a file on a PC server using REDIRECTION. |
| REO | Show only the resident entry of files. Listed on the Options screen under SELRES |
| Resident | Show all entries of files with a RESIDENT entry on disk. Listed on the Options screen under SELRES |
| SAve | Colon option used to save current options in the file FLEXOPTIONSFORME. KEep is a synonym for SAVE |
| SETCode | Show code generated by the RULE compiler (see RULES) |
| SETList | Show text passed to the RULE compiler (see RULES) |
| STRICT | Only applies when Flex is run in batch mode. Will stop with an accept message if any syntax errors are found in a DO file. |

| Option | Brief description |
|--------------|---|
| SYNTAX | If set, actions on files will be checked for syntax only. Jobs started by LOAD will only compile for syntax |
| TERMINAL | Allows output to the terminal if PRINT or FILE are set |
| TOUCH | Colon option used with the Load command to change the access date of reloaded files. |
| TRUNCATE | Truncate output on LIST or WRITE to 80 characters |
| TWOBACKUPS | RELEASE command will require that a file has 2 backups |
| UNSEQUENCED | Do not show sequence numbers when Listing or Writing |
| UNSORT | Do not sort the output from FILE:FLAT. |
| USDATES | Show dates as MM/DD/YY, otherwise DD/MM/YY |
| WAIT | If set, FLEX will wait for your input after each page. If a value appears after WAIT, this will be the number of seconds to wait at a page break. |
| MEMSZ | Used as a colon option when compressing files. |
| ZIPMORE (ZM) | Used as a colon option when compressing files. |

Option groupings

Options fall into 6 groups as follows:

I/O options

APPEND, AUTORM, BYTE, FILE, XLATE, PRINT, REDIRECT, TERMINAL, TRUNCATE, UNSEQ

WFL and file maintenance options

ACTION, NOMSGS, SYNTAX

Terminal options

WAIT

Attribute display

ARC(exclusive to CAT), ATT, CANDE, CAT (Exclusive to ARC), FIND (colon option), LEVEL (colon option), NR, REONLY, RES

Performance options

FAST ,FLAT, LINEAR, UNSORT

Miscellaneous options

AUDIT, BRIEF, MONITOR, SAVE(colon option), USDATES

Special or privileged options

DEBUG, SETCODE, SETLIST, TWOBACKUPS, ZIPMORE(colon option),
MEMSZ(colon option)

Alphabetical listing of options & descriptions

The available options are given below in alphabetical order. Most options may be set for the duration of the FLEX session by using the OPTION command. To set an option or options only for the current command, put the option list after a colon (":") character at the end of the command, e.g.

FILES :ATT,CAT,-USDATES,R

would temporarily set the <attribute option>, the <catalog option>, and the <resident option>, and reset the <USDATES option>. Note that the minimum abbreviations for each option is indicated on the 'OP ?' display and the following descriptions by showing it in capital letters.

<ACTion option>

Used with the FILES command to present a menu of the files found, and allow actions to be taken on the individual files. For a full description, see the [discussion](#) at the end of the OPTION command in this manual.

<APpend option>

— :Append — —

The append option allows output to be appended to an existing file. If the specified file does not exist a file of type JOB will be created. It will shut off output to the terminal unless the <terminal option> is also specified. The name of the output file, if no <filename> is specified, is FLEX/JOBOUT.

<ARChive option>

The <archive option> is used in conjunction with the FILES command to show the archive information for each file found. This includes backup information, generations of the file, and resident status.

Used with LOAD, it uses the Archive Backups on a Cataloging MCP.

It is used in the BACKUP command to make the copy an ARCHIVE FULL, instead of

a plain ADD.

Setting this option resets :CAT.

<ATT option>

Used with the FILES command to display the file attributes of the resident entry of the given file. These attributes include header attributes, blocking and size information, dates and times showing changes and accesses to the file and security information.

Any wastage due to misaligned blocks is indicated by "+nn" after the "BlkSz entry.

```
files description/acdb :att
DESCRIPTION/ACDB 1:0 DASDLDATA (108)
Created 08:13:00,28/03/08 BST Security PRIVATE IO Extmode EBCDIC
Altered 08:13:00,28/03/08 BST FilStr ALIGNED180 IMP Rec/Blksz 256/256+14 W
Used 13:58:15,14/10/08 BST BlkStr FIXED Length 3072 W
CopiedTo 12:28:58,06/02/09 GMT LastRecord 11 Rows 1*720 S
#Selected 1 file [from 1 file]
#Directory: DESCRIPTION/ACDB ON DEV
```

For the

file shown above the wastage is 14 words per block.

Also called the [<attribute option>](#)

<AUDit option>

All input from the user and all output from the FLEX inquiry is also sent to the printer. This option is useful to keep a record of a FLEX session.

<AUTOrm option>

Used in conjunction with the <file option> ,<byte option> and <redirect option> to allow an existing file to be overwritten.

<BRief option>

FLEX will suppress certain messages or present them in a shortened form. As you gain experience with FLEX, you may wish to set this option permanently. It has a special role in the HELP ATT command where it suppresses the full documentation of each attribute in the report. Instead a concise table is returned. It has a similar effect in the USAGE command. Instead of giving the full breakdown by FILEKIND, only one line per directory level is returned giving the grand totals.

<Byte option>

— :Byte —————|
 |
 | <filename> |

The byte option allows output to be sent to a byte stream file. This removes any practical line length restrictions. Such files may be read from PCs via NXSERVICES or could be transferred to PCs for further processing. It will shut off output to the

terminal unless the <terminal option> is also specified.

The name of the output file, if no <file title> is specified, is:FLEX/JOBOUT.

If the [<autorm option>](#) is not set then the file given by <filename> must not be present, or FLEX will respond with "#Output file already exists".

See also:

[<redirect option>](#) [<file option>](#) [<append option>](#)

<CANDE option>

The default report produced by the FILES command is a hierarchical listing, similar to the CANDE FILES command. Whenever a REMOVE, RELEASE, or DESTROY is done for a file, the corresponding object will also be acted upon. For example, if you DESTROY a program called TESTPROG, the object file OBJECT/TESTPROG would also be DESTROYed if the <CANDE option> were set. Use of other output options like ATT to CAT or ARC will override the listing behaviour of the CANDE option.

<CATalog option>

Used with the FILES command to show the catalog information for each file found. This includes backup information, generations of the file, and resident status. It is only meaningful on cataloging systems.

ARC and CAT are mutually exclusive. The most recently set will reset the other.

<COPYWrite option>

For the BACKUP command this forces a call to Library/Maintenance to go via CopyWrite.

<DEBUG option>

This option is for use by METALOGIC personnel only. It may be followed by an optional value to set various debugging flags, which are subject to change.

<Fast option>

The <fast option> (:FA or :FAST) causes the FILES command to search the File Access STructure directly. This is very fast for large searches but only the title attributes are returned.

:FAST can be used in LOAD with only info in the Archive requested. This is implemented by doing a lookup of the Archive info for each file and so is very slow. It is principally used with LOAD:FAST to recover a lost family on non-Cataloging systems.

:FAST can be used with searches on MT/CD or IN containers. This can be much faster than the non-FAST call, but it only looks at the Directory, not at the headers,

which means only the title related attributes are available (and LicenseKey).

Note that the tape directory on a tape that has been Appended only shows the files of the first COPY.

Using :FAST on a container with CopyWrite allows access to more types of directory, including PKZIP and TAR containers.

<File option>

— :File — <filename> —

The file option takes an optional <file title>

Ex:

```
FILES A/=:File FILES/A ON TEST
FILES (*):=File FILES/ALL ON TEST,FL,AUTORM
FILES =:File,Term
USAGE (*):=File USAGE/ALL
```

The FILE option allows output to be sent to a file of type JOB. It will shut off output to the terminal unless the <terminal option> is also specified. The name of the output file, if no <file title> is specified, is FLEX/JOBOUT.

If the <autorm option> is not set then the file given by <file title> must not be present, or FLEX will respond with "#Output file already exists".

<FIND option>

— :FIND — ^ — <find target> —

The :FIND option filters the files found depending on the presence of the target in the data content of the file.

If "^" is present, the files returned do not have the target, otherwise the ones with the target are returned.

Literal FINDs treat the file as a SECTORSTREAM so any data between the start and the end of the physical file is scanned, including gaps due to bad blocking, sequence and mark fields, and allocated space after the end of the file. Record boundaries are ignored.

If no literal specifier is given, the Find becomes a Whole Word search and also will stop the search at the logical end of file. Further, if the file is not badly blocked and its FILEKIND has a defined sequence number field, only the text field will be searched and the edges of the text field will be allowed as word terminators.

:FIND may not be used in conjunction with :LIN or :FAST. Any attempt to do so will result in the error:

When a FIND uses FLAT, the UNSORTED option is set, to avoid sorting large disk file headers. The results of the FIND can be sorted using the SORT modifier or REPORT SORT.

See also:

<find target>

<FLat option>

Used in conjunction with the FILES command to enable FLEX to search for the files in a fast linear search through the SYSTEMDIRECTORY. . The search can be limited by use of a selection expression (using the SELECT command). The main advantage of using this option is that a search through the entire family directory can be done approximately two hundred to one thousand times faster than ordinary GETSTATUS directory calls. Flat can be followed by an optional integer to restrict the search to files which have at least one row on a specific member or a multi pack family. The files are not found in sorted order, so in the FILES command (not in the USAGE command) it is necessary to sort them. The <UNSORT option> can be used to avoid this extra overhead. FLAT may also be used to searching old: SYSTEMDIRECTORIES; see the 'MAINT ALT FLAT' syntax of the MAINT command.

See also:

[<linear option>](#) [<fast option>](#).

Linear Flat and Fast are mutually exclusive. The most recently set will reset the others

<INTER option>

When FLEX Inquiry is run with a non-blank parameter, it will terminate after doing the command unless the <inter option> has been added at the end of the "U" statement.

For example,

```
U FLEX FILES XYZ:INTER
```

would show the list of files in the directory XYZ, and then wait for the next command. This is especially useful in a COMS environment, as FLEX Inquiry can be run with the parameter ":INTER".

<LEVEL option>

Differs slightly from the other options in that the keyword 'LEVEL' is never used. This is a name for the ability to specify that FILE displays only show the file names to a certain level or depth. The LEVEL option is specified by using the numbers

1-14 as <colon options>. For example, FILES :2 would show only the first two levels of filename after the usercode. It has a different meaning in the USAGE command. Specifying a Level N with USAGE causes a preliminary FILES:N, then a USAGE is done on each directory found. See the USAGE command for examples.

<LINear option>

Used in conjunction with the FILES command to enable FLEX to search for the files in a fast linear search through the SYSTEM/CATALOG. The search can be limited by use of a selection expression (using the SELECT command). The main advantage of using this option is that a search through the entire catalog can be done approximately one thousand times faster than ordinary GETSTATUS directory calls. For efficiency's sake, the files are not returned in sorted order.

This option may also be used to searching old: SYSTEM/CATALOGs; see the 'MAINT ALT CATALOG' syntax of the MAINT command.

See also:

[<flat option>](#) [<fast option>](#)

<MEMSZ option>

Normally, MZIP/GZIP look at the system to determine how much memory to use. MemSz is essentially a debug option to override the memory size used. It is normally (Default), but can be set to (Big), (Med), and (Small) corresponding to 16MW, 1 MW, and 64kW respectively. Add at least 25% if ZipMore in set. MemSz is not kept between sessions.

See also:

[<ZipMore option>](#)

<MONitor option>

Causes FLEX Inquiry to periodically display a message indicating how far the search through the directory for files has progressed. This is especially useful to see the status of the search when a <selection expression> is being used.

<NOMSGs option>

Messages from the WFL jobs from FLEX will NOT be displayed at the terminal. This option may be used with the WFL and LOAD commands. The setting of the CANDE option MSG (set by SO MSG) may affect the display of these messages.

<NR option>

The NR or NONRES option is used to restrict the search to files that have backups on tape or disk, but which have no entries resident on disk. This option is not meaningful for non-cataloging systems. A valid synonym for NR is NONres.

See also:

[<resident option>](#) [<REO option>](#).

<Print option>

The print option is used to send output from the program to the line printer. Unless the TERMinal option is set, setting the <print option> will suppress the output at the terminal. For example, 'FILES XXX :CAT,PRINT' will send a listing of the files in directory XXX, together with the catalog information, to the line printer. To receive the printed output, you must quit the program. The output from FLEX inquiry will go to the DESTNAME currently in force for the CANDE session of the user, or to the default DESTNAME of the user if the USER command has been used to change the <usercode> of the FLEX session. Note that the [<file option>](#) may be used to send output to a JOBSYMBOL file.

<REdirect option>

— :REdirect — <filename> _____ |

:REDIRECT is similar to :Byte except that the file goes directly to a PC.

Normally CREDENTIALS have to be set up before this can function.

Ex.

```
FILES =:FL,Redirect:MYSERVER/PRIMARY/METALOGIC/"FILES.TXT"
```

Or

```
FILES=:FL,Redirect:"10.0.0.4"/PRIMARY/METALOGIC/"FILES.TXT"
```

The usercode and password used to access a server is controlled by the MAKECREDENTIALS Utility. It must be run from the usercode to be used to run Flex.

The MAKECREDENTIALS utility provides a method of encrypting credentials, eliminating the need to specify explicit credentials as part of the IOHSTRING attribute, and creates a credentials file for each system for which the user has remote access. The MAKECREDENTIALS utility can also be used to create a default credentials file.

MAKECREDENTIALS is a simple command mode utility that expects the following login credentials for a network host:

```
<host>  
<username>  
<password>  
<user domain> (optional)
```

These four pieces of information correspond to the IOHSTRING keywords as follows:


```
SERVER = <host>
CREDENTIALS = <username>/ <password>
USERDOMAIN = <user domain>
```

Examples

The following example shows how the MAKECREDENTIALS utility is run from the CANDE terminal emulator for a host of MYSERVER (10.0.0.10), a username of OPS, and a password of frog:

```
U *SYSTEM/NXSERVICES/MAKECREDENTIALS MYSERVER OPS"frog"
```

or if the IP address is to be used:

```
U *SYSTEM/NXSERVICES/MAKECREDENTIALS 10.0.0.10 OPS "frog"
```

Note that if the password on the PC is not all uppercase then the password must be quoted.

Using a prefix of *UNC/ on an output filename has the same effect as Redirect.

So

```
FILES =:FL,Redirect:MYSERVER/PRIMARY/METALOGIC/"FILES.TXT"
```

is equivalent to

```
FILES =:FL,FILE *UNC/MYSERVER/PRIMARY/METALOGIC/"FILES.TXT"
```

If any of the directories specified in the file name do not exist, Flex will attempt to create them. The ability to create them will depend on the privileges of the user code specified in the Credentials file.

A word of warning. A credentials file created under one usercode will not work if copied to another usercode.

The AutoRM option may be used with the Files,Byte,Append and Redirect options. Setting AutoRM allow a file specified in a Files option to be overwritten, whereas normal operation will return an error if the specified file already exists.

Ex.

```
FILES=:FL,Redirect:"10.0.0.4"/PRIMARY/METALOGIC/"FILES.TXT",AutoRM
or
FILES=:FL,AutoRM,File FLEX/OUTPUT
```

<REO option>

The <REO option> is used to restrict a file search to those file entries, which are resident on disk. (REO stands for Resident Entry Only). This option is not meaningful unless the system is cataloging. The output produced when using this option is a subset of that produced when using the <resident option>.

See also:

[<NR option>](#) [<resident option>](#)

<Resident option>

The RESIDENT option is used to restrict the search of files to those which have a

resident entry. Valid synonyms are R and RES. For non-cataloging systems this option is not meaningful.

See also:

[**<NR option>**](#)

[**<SAve option>**](#)

Saves the current values of many of the options and <colon options> in a file called FLEXOPTIONSFORME. This option is only meaningful when used with the OP command, e.g. "OP AUDIT,BRIEF,-USDATES :SAVE" .

[**<SETCode option>**](#)

Produces a listing of the GSMACHINE code generated when using the RULE compiler for SELECT, REPORT, or <RULEs ENTER>. This option is for METALOGIC internal use and is not of interest to most users.

[**<SETList option>**](#)

Prints a listing of the text of the parameter passed to the SELECT, REPORT, or RULEs ENTER commands, plus a listing of compilation errors if there are any. Regardless of the setting of the <setlist option> or the <setcode option>, if FLEX inquiry is being run interactively, the source and any errors will be shown at the terminal.

[**<SYNtax option>**](#)

The syntax of the current command will be checked, but it will not actually be executed. This command is useful for learning how to use the file maintenance commands, and with the WFL command.

The WFL job created when using the LOAD command will NOT be started, and will be saved in a disk file, under the directory "RELOAD".

[**<TERMinal option>**](#)

The TERMINAL option is used to allow output to the terminal when the <print option> and/or <file option> are also being used. To have an effect, the <term option> must appear in the options list after <print option> and/or <file option>. This option is assumed by default.

[**<TRuncate option>**](#)

Truncates the output from the LIST command to a width of 80 characters. It is set by default.

[**<TWobackups option>**](#)

The RELEASE command will require that a file has two valid backups before it will

allow allow that file to be RELEASE'd. If not set, only one backup will be required. With the BACKUP command it causes a duplicate backup to be made.

<UNSEquence option>

The output from the LIST and WRITE commands will NOT contain sequence numbers.

<UNSort option>

The output from a FILE:FLAT will not be returned in normal order, but in the order the file headers appear in the SYSTEMDIRECTORY. Since no sort is involved, results will appear slightly quicker.

<USdates option>

All dates shown by FLEX inquiry will be in MM/DD/YY format, otherwise the European convention of DD/MM/YY will be used.

<WAIT option>

If the <wait option> is OFF, then FLEX will not wait for your input after each page of output. If the <wait option> is ON, then FLEX will wait for your input after transmitting each page. If the value of the wait option is set to a number (e.g. OP WAIT 3), then FLEX will wait for input for that number of seconds, before showing the next page. This option has no effect on printed output.

<Xlate option>

Used in conjunction with the LIST or WRITE commands to show non-graphic characters in hexadecimal. The output produced contains two lines for each record in the file. The top line is all of the printable EBCDIC characters and the first hexadecimal character of unprintable EBCDIC characters, the second line is blank if printable and the second hexadecimal character of unprintable EBCDIC characters. For example, ESC "J" XYZ ESC "K" NUL would appear as

```
2JXYZ2K0
7      7 0
```

since the ESC character is 48"27" in hexadecimal, and NUL is 48"00". Note that the LIST command converts non-graphic characters to the DEL character if the <xlate option> is not set.

<ZIPMORE option>

ZipMore or ZM controls how much effort goes into the compression in MZip and GZip. If it is OFF, the compression is done with as little CPU time as possible. If ON, about 50% more CPU is used to improve the compression, typically ending with 10% smaller compressed file size. ZM ON files normally UNZIP more quickly than ZM OFF ones.

ZM can also take an integer value, though this is not saved between sessions. The

only current assigned meaning is for ZM 2 in MZip. This value will often add a further 50% CPU time over ZM 1, and drop the final compressed file size a further 7%.

All these % figures are data dependent and each file will behave somewhat differently. However, a file with ZM 2 compression will normally need about double the time to UNZIP, vs ZM 1.

See also:

[<memsz option>](#)

The ACTION option

On the following pages, further discussions of the <action option> and the <linear option> are given. Through use of the <action option>, a user may easily step through his files, and specify an action to be performed on each file. When used, this option presents a screen that displays several filenames, some pertinent attributes, and fields beside each file where an action may be entered. At present, the following actions are allowed:

BACKup, RELease, REMove, DESTROY, List, LFILES, LOAD, Quit, Help and Refresh

When you have entered all of the actions for the current page, press XMIT. If you do not wish to take any action on the present page of files, press SPCFY, and you will proceed to the next page. The syntax for the <action option> is the same as for the 'FILES' command, with the addition of the <action colon option>. For example, if a user wanted to examine his resident files under the directory 'XXX/=', he would enter 'FILES XXX/= :R,ACT'. The files presented may then be 'acted' upon.

After all of the files have been examined, or the user has entered 'QUIT' as a command, the program will enter the authorisation phase. If a DESTROY, RELEASE or REMOVE has been requested for a file, it will appear now on a second screen, called the Authorisation screen, (13 files at a time on a 24 line screen) and will ask for an 'OK' to actually do the REMOVE, RELEASE or DESTROY.

| Enter Help and XMIT for instructions | | | | | |
|--------------------------------------|---------------------------|--------------------|----------------|-------|-----------------|
| Action | Title & filekind | Segments | Altered | Bkups | Accessed |
| ▶ | ◀ BOB/DO ON DEV | | | | |
| | SEQ & DIRECTORY | (14) | 12/05/08 | 2 | 29/01/09 |
| ▶ | ◀ BOB/DO/TEST ON DEV | | | | |
| | SEQ | (14) | 13/11/08 | 2 | 13/11/08 |
| ▶ | ◀ BOB/DO/INDENT ON DEV | | | | |
| | JOB | (14) | 20/11/08 | 2 | 20/11/08 |
| ▶ | ◀ BOB/DO/EMTPPARAM ON DEV | | | | |
| | JOB | (14) | 11/12/08 | 2 | 11/12/08 |
| ▶REM | ◀ BOB/XX ON DEV | | | | |
| | JOB | (15) | 05/10/04 | 2 | 17/07/08 |
| ▶REM | ◀ BOB/CNS ON DEV | | | | |
| | SEQ | (45) | 24/01/08 | 2 | 17/07/08 public |
| ▶ | ◀ BOB/FTP/CONVERT ON DEV | | | | |
| | ALGOL | (14) | 21/10/99 | 2 | 09/11/04 |
| ▶ | ◀ BOB/JOB ON DEV | | | | |
| | JOB & DIRECTORY | (14) | 12/11/08 | 2 | 12/11/08 |
| ▶ | ◀ BOB/JOB/BAD ON DEV | | | | |
| | JOB | (14) | 03/09/08 | 2 | 03/09/08 |
| ▶ | ◀ BOB/JOB/MULTI ON DEV | | | | |
| | JOB | (14) | 20/11/08 | 2 | 20/11/08 |
| Familyname : DEV | | | Usercode : BOB | | |
| 6339 | Utility File Handler | Version 53.530.284 | 15:30:52 | | |

```

      Authorisation Screen

You are about to REMOVE the following files. To verify this :
a) Enter Y !xmit!   to REMOVE all files on this page
b) Enter N !xmit!   to skip this page of files and go to the next page
▶N c) Enter H !xmit!   to get help before proceeding
d) Enter Q !xmit!   to ignore this page of files and go to command mode
e) Enter R !xmit!   to refresh the current page
OR, to omit a file, move the cursor to the <filename> and press SPCFY

(BOB)BOB/XX ON DEV
(BOB)BOB/CNS ON DEV

      Familyname : DEV                      Usercode : BOB
  
```

Please note that the settings of other <colon options> may affect the processing of the selected files (e.g. the SYNTAX option).

By entering a letter in the forms box, you give FLEX inquiry confirmation that the files shown should or should not be processed. To omit a file from consideration, move the cursor down to the <filename>, and press SPCFY. The file will be deleted from the list, and the menu will be presented again. Entering 'N' causes FLEX inquiry to skip all of the files on this page, and continue with the next page to be authorised.

The <syntax option> may be set by entering 'S' in the forms box. If 'Q' is entered, none of the files will be processed, and FLEX inquiry will return to FLEX mode. Entering 'R' or CTRL 00 will refresh the current screen.

If 'Y' is entered, the indicated action will be performed on the files listed on this page of the authorisation screen, e.g. the files will be Destroyed, Removed, or Released.

The action can be pre filled by using any of the following commands with the :Action:

BACKUP, LFILES, LOAD, PURGE, RELEASE, REMOVE, DESTROY, MZIP, GZIP.

Ex: Backup to MyTape bob/=:act

| Action | Title & filekind | Segments | Altered | Bkups | Accessed |
|------------------|-----------------------------------|----------|----------------|-------|-----------------|
| ▶BACKUP ◀ | BOB/DO ON DEV SEQ & DIRECTORY | (14) | 12/05/08 | 2 | 29/01/09 |
| ▶BACKUP ◀ | BOB/DO/TEST ON DEV SEQ | (14) | 13/11/08 | 2 | 13/11/08 |
| ▶BACKUP ◀ | BOB/DO/INDENT ON DEV JOB | (14) | 20/11/08 | 2 | 20/11/08 |
| ▶BACKUP ◀ | BOB/DO/EMPTPARAM ON DEV JOB | (14) | 11/12/08 | 2 | 11/12/08 |
| ▶BACKUP ◀ | BOB/XX ON DEV JOB | (15) | 05/10/04 | 2 | 06/02/09 |
| ▶BACKUP ◀ | BOB/CNS ON DEV SEQ | (45) | 24/01/08 | 2 | 17/07/08 public |
| ▶BACKUP ◀ | BOB/FTP/CONVERT ON DEV ALGOL | (14) | 21/10/99 | 2 | 09/11/04 |
| ▶BACKUP ◀ | BOB/JOB ON DEV JOB & DIRECTORY | (14) | 12/11/08 | 2 | 12/11/08 |
| ▶BACKUP ◀ | BOB/JOB/BAD ON DEV JOB | (14) | 03/09/08 | 2 | 03/09/08 |
| ▶BACKUP ◀ | BOB/JOB/MULTI ON DEV JOB | (14) | 20/11/08 | 2 | 20/11/08 |
| Familyname : DEV | | | Usercode : BOB | | |

The word BACKUP could then be deleted for any file which is not wanted.

The FLAT and LINEAR options

The <flat option> and the <linear option> are used in conjunction with the FILES command to enable FLEX to search for files using a fast linear search through the SYSTEMDIRECTORY (or the SYSTEM/CATALOG for the <linear option>). The search can be limited by use of a <selection expression> (using the [SELECT](#) command).

Using the <linear option>, a search through the entire catalog (that is, all of the resident and non-resident files on **all** of the packs) can be done approximately one thousand times faster than a search done using GETSTATUS directory calls. However, the <linear option> is only available on Cataloging systems. The <flat option> searches through all of the resident files on a single pack. Note that the files are not returned in sorted order with the <linear option>.

These options may also be used in conjunction with the 'MAINT ALT' syntax of the MAINT command to search old SYSTEM/CATALOGs or SYSTEMDIRECTORYs. See the 'MAINT ALT' syntax of the [MAINT](#) command for details. The <linear option> with MAINT ALT CATALOG can be used even on a non-cataloging system, which is useful for catalog maintenance when the catalog must be temporarily off line.

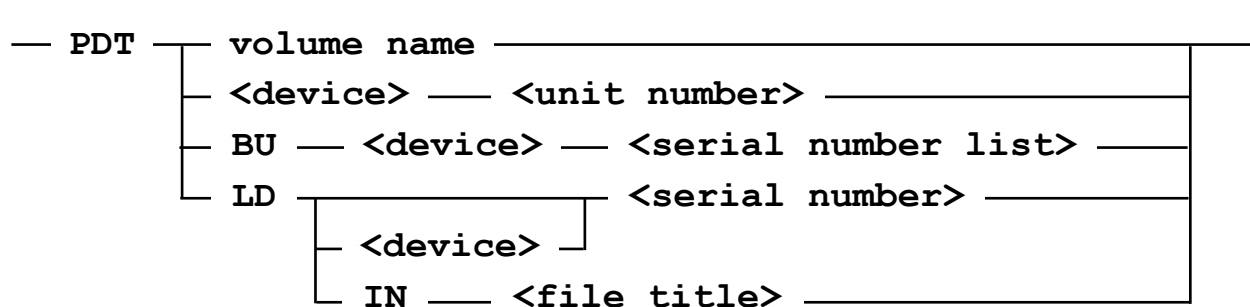
Only attributes contained in the SYSTEM/CATALOG can be used in the <selection expression> or REPORT, HEAD, or FOOT specified for use with the <linear option>. These include:

| TITLE | RESIDENTENTRY | CATALOGTIMESTAMP |
|------------------|--------------------|------------------|
| BACKUPSERIAL (1) | BACKUPVOLUMEID (1) | GENERATION |
| BACKUPSERIAL (2) | BACKUPVOLUMEID (2) | FILEKIND |

When using the <flat option>, only attributes of resident files can be used in the <selection expression> or REPORT. Not all of the attributes of a resident file can be returned; an example is SECURITYGUARD. For descriptions of these attributes, return to FLEX mode and enter a variant of 'HELP ATTS'..

The <flat option> can be particularly useful when used with the FILES, SELECT, and REPORT commands to provide an ultra-fast replacement for SYSTEM/FILECOPY. See [the Filecopy section](#).

PDT command



<device> ::= MT | CD | PK

The PDT command shows a tape directory on the LIBRARY/MAINTENANCE tape assumed to be on the tape unit specified by the parameter. A synonym is 'TAPEDIR'. Only a privileged user may use the 'MT <unit number>' variant.

Note that the output can be sent to the printer with the <print option>, or to a file with the <file option>.

The PDT BU variant is used to show the files that have certain volumes as backup references. For each volume in the <serial number list>, any file that has it as a backup volume will be listed. Note that files on family tape are spurious entries, and can be deleted using the TAPEDELETE STD report. This command is only valid on cataloging systems

The LD form is used to access the info in a LIBMAINTDIR file. If the 'IN' form is used, the full title of the LIBMAINTDIR must be given.

Note that 'F <pattern> in <file title>' is another way of looking at the contents of a LIBMAINTDIR.

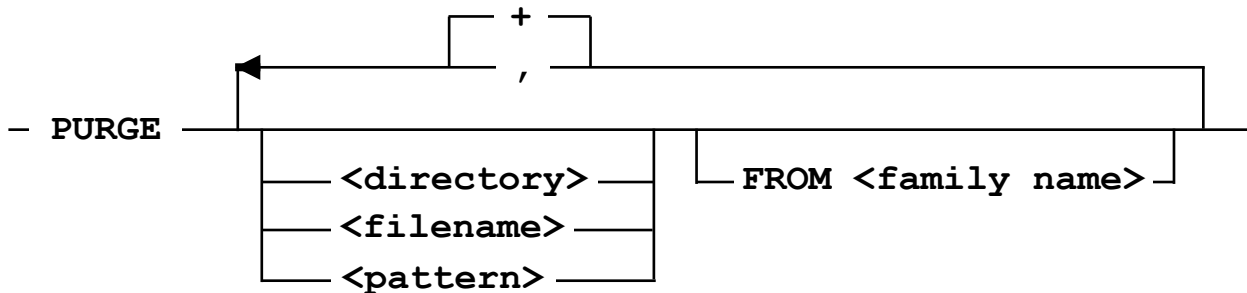
When <serial number> is used, FLEX tries to find the name using the Volume Library, Volume Directory, or TRIM, one or more of which must be present.

If <device> is omitted, FLEX assumes first MT, and failing that PK/CD.

Example:

```
PDT MYTAPE
PDT CD 34
PDT BU MT C00023 C00038
PDT LD C00023
PDT LD IN (PROD)LIBMAINTDIR/DBBKUP/20040312/C00023 ON DLPK
```

PURGE command



The PURGE command is used to eliminate all backup references for a file or list of files.

The specified files will be offered one at a time for Purging, and if an 'OK' response is given, the indicated file will be Purged. Other valid responses are QUIT, to stop the search, or any other non-blank input to skip the current file. PURGE is also a valid action in the action screen.

This action is like that of a WFL CATALOG PURGE on a file). If no 'FROM <familyname>' part is specified, the current primary family will be assumed.

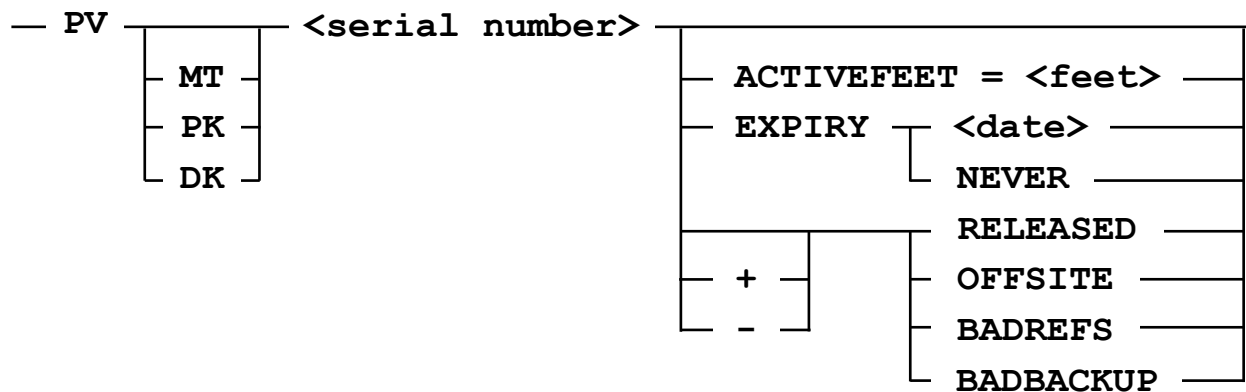
If the <CANDE option> is set, all of the above actions will also be applied to the corresponding OBJECT file, i.e. if the file XYZ is released, then OBJECT/XYZ will also be released if the <CANDE option> is set.

Since Purging the catalog information does NOT have any effect on the physical volume where the file may be backed-up, recovery is possible in certain cases where an accidental Purge has occurred; namely, if the file had a backup, and an old copy of the SYSTEM/CATALOG is available, the site manager could use the <linear option> of the FILES command to search for the SERIALNO of the backup volume. This process is fairly annoying, however, so it is best to be sure before you Purge a file.

See also:

| | | | |
|---|---------------------------------------|--------------------------------------|-----------------------------------|
| <twobackups option> | <action option> | <cande option> | REMOVE |
| <syntax option> | <linear option> | <filename> | <directory> |
| quotes | <usercode> | DESTROY | |

PV command (Cataloging Only)



The PV command prints for the requested volume and volume family requested. This command is valid only on cataloging systems. <serial number> may be from one to six alphanumeric characters. See the LISTVL command to print information about all of the volumes in the volume library at once.

CD Roms, CD Images and Disk Farms are entered in the volume library as kind PK. The response to a PV command will use the following device mnemonics if it is not a real PACK.:

CD = CD ROM
CI = CD Image
DF = Disk Farm

- PV shows the volume library and FLEX statistics file information. It also allows FLEX statistics file entry to be updated.
- <serial number> may be from one to six alphanumeric characters.
- If no device kind is specified, MT is assumed.

The optional syntax after <serialno> is only for privileged users, and allows them to modify the information in the statistics file.

- ACTIVEFEET indicates the length of the tape in feet which is used by files still in the catalog. This can be used to change the priority of the tape in a MERGE. <feet> is a positive integer.
- EXPIRY changes the expiry date, although this does not take effect unless the volume is not a LIBRARY/MAINTENANCE tape.

<date> is in the format DD/MM/YY, or MM/DD/YY depending on the USDATES option.

NEVER is assumed if the date is more than 999 days after the creation date. If EXPIRY is not explicitly set, it defaults to SAVEFACTOR days after the creation date. By default, FLEX does not take any action if a tape expires.

The EXPIRY field in METATAPELIB is updated if TAPECONTROL is active.

RELEASED, OFFSITE, BADREFS, and BADBACKUP are statistics file flags. If the

+/- is omitted, + is assumed.

- RELEASED is set automatically by MERGETAPE when a FLEX tape has no more backup references. An INSTALL parameter can also make MERGETAPE release KIND=TAPE volumes.
- BADREFS is set if FLEX sees a file, which is backed on the tape, is not actually present on it.
- BADBACKUP is set if a tape is not in the Volume Library or is marked DESTROYED.
- OFFSITE is a user changeable flag, which is normally employed to denote a tape which is not conveniently available to the system.

If TAPECONTROL is active, setting and resetting OFFSITE sets the tape LOCATION to OFFSITE and SITE respectively.

```
p>v mt HP3001 badrefs
-----Volume Library entry for (MT) [HP3001]-----
Serialno HP3001 , # 1 of 1 , TAPE
Family name       : VMSA06160AF
Family created on  : 09/06/06
Family expiration date: 09/07/06
Family creation site : 1
Volume Information after last MERGETAPE run
Creation date      : 09/06/06
Expiry date        : 09/07/06
Backup References   : 10 (None Resident) Occupy about 5.0 MB <<< Bad reference
s noted
```

```
p>v pk pc0015
-----Volume Library entry for (CD) [PC0015]-----
Serialno PC0015 , # 1 of 1 , CD-ROM , base
Family name       : PC0015F2008366A
Family created on  : 31/12/08
Family creation site : 1
Volume Information after last MERGETAPE run
Creation date      : 31/12/08
Expiry date        : 30/01/09
Backup References   : 437 (362 Resident) Occupy about 180 MB
#
```

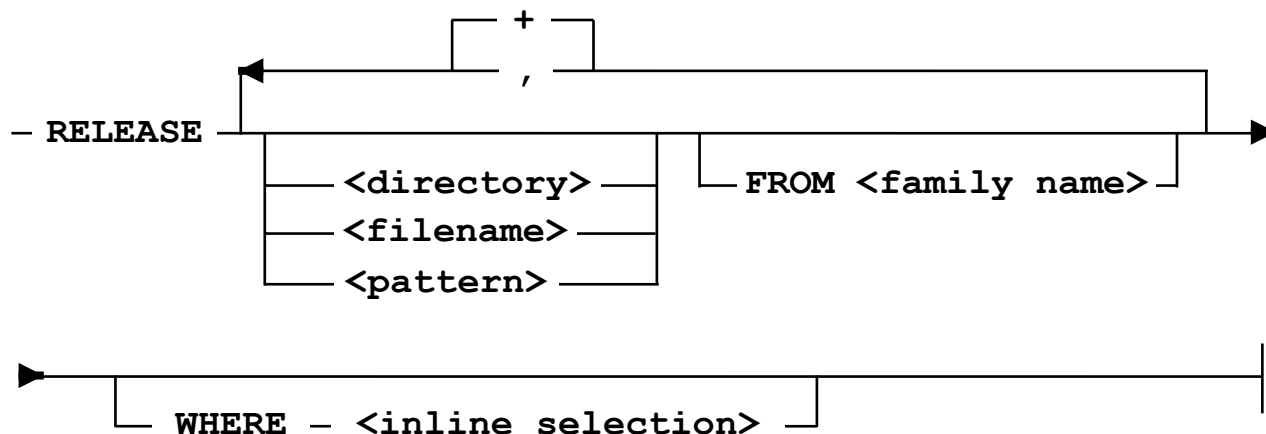
Quit command

— QUIT —

The QUIT command is used to exit the program. 'END' is a valid synonym. There are no side effects to quitting the program.

If running on a TD830-compatible terminal, to exit from HELP mode and return to FLEX mode, move the cursor inside the box in the upper left hand corner and press SPCFY. You will then be able to give regular commands to the FLEX inquiry.

RELease command



The RELEASE command is used to safely remove a file or list of files from disk, without altering any backup information. It differs from the 'REMOVE' command in that the file will only be removed if a suitable number of backups exist. The default is one backup; the <twobackups option> may be used to ensure that each file/generation has two valid backups before removing it.

The specified files will be offered one at a time for a REMOVE, and if an 'OK' response is given, the indicated file will be removed. Other valid responses are QUIT, to stop the search, or any other non blank input to skip the current file.

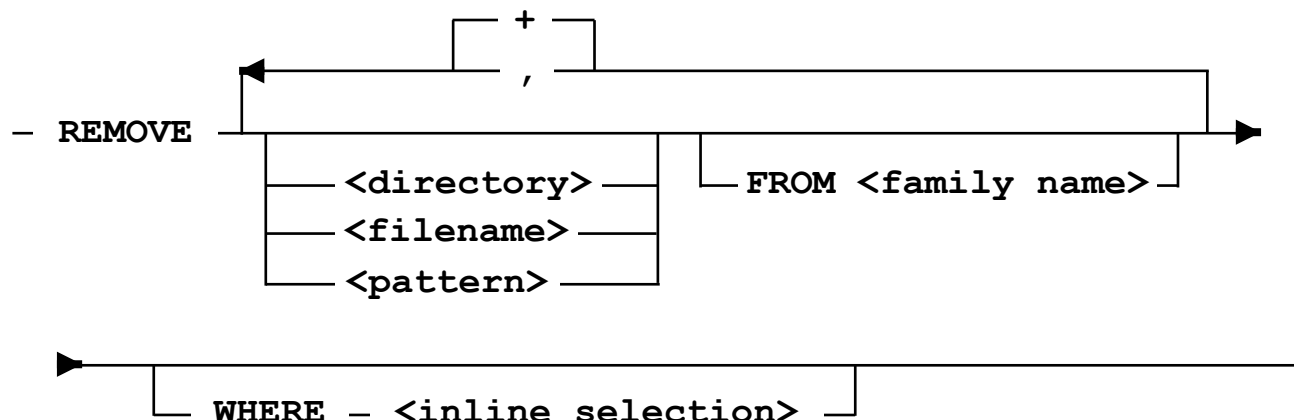
If the <CANDE option> is set, all of the above actions will also be applied to the corresponding OBJECT file, i.e. if the file XYZ is released, then OBJECT/XYZ will also be released if the <CANDE option> is set.

The WHERE clause can be used to invoke an 'inline selection'. The text following WHERE (<inline selection>) should be valid SELECT code. A WHERE clause will be used in preference to an active SELECT. The where clause can be used after a directory list and before any colon options.

See also:

[DESTROY](#) [<action option>](#) [<CANDE option>](#) [REMOVE](#)
[<twobackups option>](#) [<syntax option>](#) [<filename>](#) [<directory>](#)
[quotes](#) [<usercode>](#)

REMove command



The REMOVE command is used to remove a file or list of files from disk, without altering any backup information. The specified files will be offered one at a time for the REMOVE, and if an 'OK' response is given, the indicated file will be removed. Other valid responses are QUIT, to stop the search, or any other non-blank input to skip the current file.

If the <CANDE option> is set, all of the above actions will also be applied to the corresponding OBJECT file, i.e. if the file XYZ is Removed, then OBJECT/XYZ will also be Removed if the <CANDE option> is set.

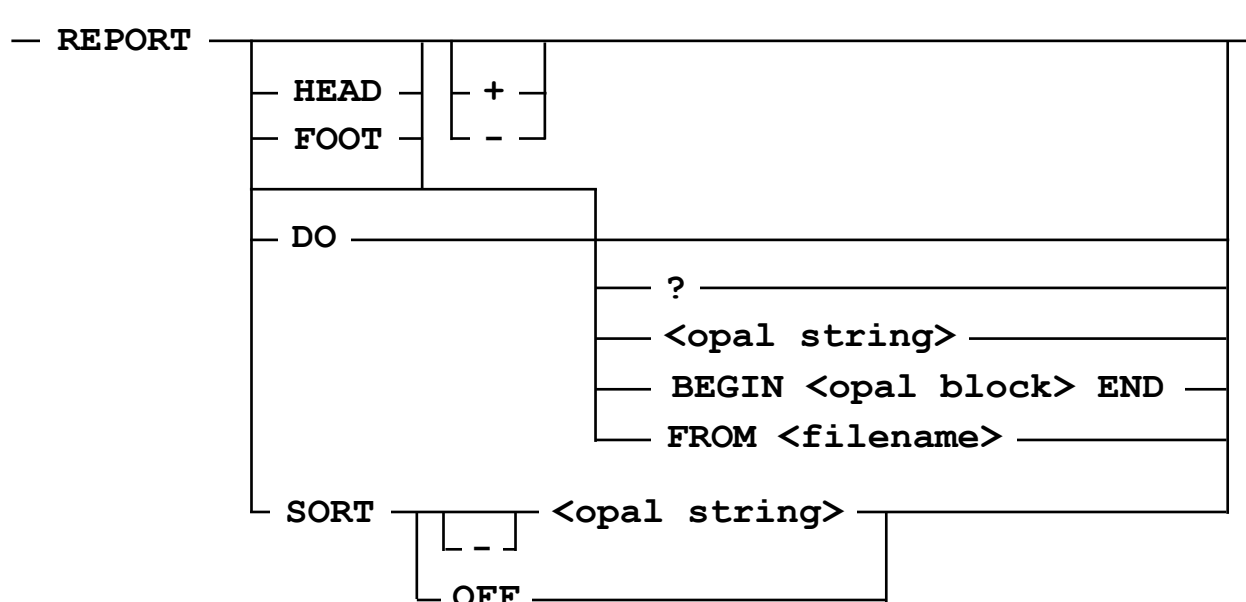
Since the REMOVE command does not consider the backup status of a file, this command is not as safe as the RELEASE command if you wish to retain the file, i.e. want to be able to retrieve it from backups. On the other hand, to completely get rid of a file, the DESTROY command should be used instead of the REMOVE command, as it will PURGE the backup information, which frees up tape space.

The WHERE clause can be used to invoke an 'inline selection'. The text following WHERE (<inline selection>) should be valid SELECT code. A WHERE clause will be used in preference to an active SELECT. The where clause can be used after a directory list and before any colon options.

See also:

| | | | |
|----------------------------------|-----------------------------------|---------------------------------------|--------------------------------------|
| RELEASE | DESTROY | <syntax option> | <CANDE option> |
| <filename> | <directory> | quotes | <usercode> |

REPort command



The **REPORT** commands allow the user to build customised reports using expressions of file attributes and special functions for use with the file search commands in **FLEX**.

The **REPORT HEAD** command is used to specify the text to be used as a heading for a report. Defining a **REPORT head** produces no output at the time, but save the code to be executed on the first file found in a **FILES** command. If no files are found then the **REPORT HEAD** code is not executed.

The **REPORT** command is used to specify the text to be used for each file found. This will replace the default text shown for each file returned by a **FILES** command.

REPORT FOOT is used to specify the text to be used as a footing for each directory scanned. The Report Foot code is executed after every directory scanned even if no files are found. The **LASTDIR** attribute returns true if the last directory in the list is being processed. So a Rep Foot used to create a **WFL** job could include code like:

```
Begin
...
...
If LastDir Then
    Show("END JOB");
End
```

Example DO file using LastDir

```
SELECT DAYS (TODAY,ACCESSDAY) > 365
REPORT HEAD
Begin
    #Cnt:=#Segs:=0;
    Show("BEGIN JOB WFL/FLEX/REMOVE;" ,/,
        "REMOVE ");
End
REPORT
Begin
IF #Cnt.Sum(1) = 1 THEN
    Show(Filename)
Else
    Show(", ",FILENAME) ;
    #Segs:=(#Segs+Segments)
End
REPORT FOOT
Begin
If LastDir Then
    If #Cnt=0 Then
        Show("BEGIN JOB WFL/FLEX/NOREMOVE;" ,/,
            'DISPLAY "No files selected to remove";',/,
            "END JOB")
    Else
        Show(';DISPLAY "',#Cnt," files ",#Segs,' segments
removed;"',/,
            "END JOB")
End
FILES TIDY/=,OBJECT/TIDY/= :FILE WFL/FLEX/REMOVE
WFL STARTJOB WFL/FLEX/REMOVE
```

If FROM <file name> is specified, the source code for the report will be read from the specified file. The file should be a JOBSYMBOL file that contains the text of the <report>, and may be formatted in any way desired. Any text appearing to the right of a % sign is treated as a comment and is not processed by the compiler.

If no text appears after the REPORT command, and there is a CANDE workfile, FLEX will try to compile the <report> from the workfile. This can be useful when debugging a complicated <report>.

The code for the above report variants can be temporarily disabled using the - syntax.

```
rep head -
#The current REPORT HEAD will NOT be used on subsequent requests
```

This does not discard the code and it can be re-enabled using the + syntax.

```
rep head +
#The current REPORT HEAD will be used on subsequent requests
```

'REPORT -' will temporarily disable use of the current <report>, <report head>, and

<report foot>. 'REPORT +' re-activates the current <report>. The <report head> and <report foot> must be reactivated separately using 'REPORT HEAD +' and 'REPORT FOOT +', respectively.

REPORT SORT

The results from a directory search may be sorted using an OPAL report. This can be specified in-line using a SORT modifier or can be specified like a header using the variant command, REPORT SORT.

The string segments in the report are truncated to 128 chars if no length is explicitly given. The total length is truncated to 512 chars. The report is executed once per eligible file and the result is used as a string key. That key determines the order the files are returned. E.g.

```
Files =CAT= WHERE SEGMENTS > 99 SORT - SEGMENTS:FL  
or
```

```
REP SORT SORT - SEGMENTS:FL
```

followed by

```
Files =CAT= WHERE SEGMENTS > 99
```

The syntax of the SORT clause is:

```
_____ SORT [ _ ] _____ <opal string> _____ |
```

where - denotes sorting down. To turn off a REP SORT, use REP SORT OFF.

Note that if there is a WHERE clause in a command, it must precede the SORT clause.

REPORT DO

This is a special case. The code is compiled and executed immediately. This is expected to be used in DO files to set up variable to be used in subsequent Reports or Selects

Ex.

```
Report Do  
Begin  
  If HostName="TEST" The  
    #SegLimit:=10000  
  Else  
    #SegLimit:=100000;  
End
```

A Report DO can also be used to interrogate permanent variables:

```
REP DO $TEST.Perm  
or
```

```
REP DO
```

```
Begin
```

```
    Show("Perm Report",/,  
        "Test=" $Test.Perm,/,  
        "Live=", $Live.perm) ;
```

```
End
```

There may be a <footing> specified if the STD command has been used to get a standard report. The <footing> is ignored when using the FORGET command and the FILES:ACT variant of the FILES command.

To see the current source for any of the Report variants use the ? syntax.

Ex: Rep ? could return:

```
REPORT  
Segments 8,,Filename  
#Current REPORT is enabled
```

There may be a <heading> specified if the STD command has been used to get a standard report. The <heading> is ignored when using the FORGET command and the FILES:ACT variant of the FILES command.

Reports are defined using the Opal language. A full description of the Opal language is available in the Opal Programming Reference Manual. A brief description can be found in the [Opal Section](#) of this document.

<opal string>

A simple report can consist of an <opal string> , similar to a string expression in WFL, but with significant extensions for ease of use.

Ex.

```
REPORT FILENAMEID,/,  
"    Age ",Days(Today,Creationday) 6,, "Days Size ",  
Segments*180/1000 5.1,, "kB"
```

In an Opal string the ,(comma) may be used to concatenate strings. The / character forces a new line in the output.

A numeric constant after an expression defines the space used to display that expression so

```
Days(Today,Creationday) 6
```

will show the age of a file in days in a column 6 characters wide.

```
Segments*180/1000 5.1
```

will display the size of the file in kilo bytes as a decimal value with 1 place after the point and 5 before.

FileNameID, Today, Creationday and Segments are attribute which return values for the current file.

HELP ATT can be used to check what is returned by an attribute.

Ex.

```
help att filenameid
FILENAMEID Returns STRING in the form of a File title
Semantics : FILENAMEID returns the TITLE of the current file
            without any ON FAMILY or preceding (USER) or *.

```

```
help att creationday
CREATIONDAY Returns INTEGER as Julian Day
CREATEDAY is a synonym for this Attribute
UNDEFINED for Non-Resident files,
and its value is for the Resident entry only.
Semantics : CREATEDAY returns the value of the DATE file
            attribute as YYYYDD. Use CREATIONTSDAY for YYYYDDD.

```

Files TEST/= with the above report active gives something like:

```
TEST
  Age    1465 Days Size    184.3 kB
TEST/1
  Age     190 Days Size     2.5 kB
TEST/2
  Age    2288 Days Size     2.5 kB
TEST/3
  Age    2288 Days Size    47.9 kB
TEST/4
  Age    2212 Days Size     2.5 kB
TEST/D0
  Age    2607 Days Size     2.5 kB
TEST/D0/1
  Age    2607 Days Size     2.5 kB
TEST/D0/PARAMLEN
  Age    1539 Days Size     2.5 kB
TEST/GS
  Age    2967 Days Size     2.5 kB
TEST/GS/SUB5
  Age    2961 Days Size     2.5 kB
TEST/GS/6SUB13
  Age    2962 Days Size     2.5 kB
TEST/PP

```

<opal block>

For more complex reports, generally used when reporting to an output file rather than the screen, an <opal block> may be used.

The use of blocks is controlled by the key word BEGIN and is permitted for normal REPORTs, HEAD and FOOT.

The definition of an <OPAL block> is identical to that of ODTSEQUENCE in SUPERVISOR. The usual OPAL block constructs such as BEGIN, END, IF, THEN, WHILE, DO, and CASE are all available.

Statements in an <OPAL block> must be terminated by a semicolon (;).

New OPAL statements, some familiar to SUPERVISOR users, are permitted from a REPORT BEGIN block; these include the following:

| | |
|---------|-------------------|
| DISPLAY | (<opal string>) |
| ODT | |
| SHOW | |
| WFL | |
| ABORT | |
| EXIT | |
| QUIT | |

The ODT, DISPLAY and SHOW statements are identical to their SUPERVISOR equivalents. SHOW displays string contents back to the originating station or to an output file if requested; the DISPLAY command forces a display message of the string to be generated. ODT passes the string contents to the MCP using the DCKEYIN function; it is only available to Privileged or SecAdmin usercodes.

The WFL statement allows WFL-only commands to be locally processed (i.e using the current session attributes such as USERCODE and FAMILY) and, when associated with a REPORT, will act on each file returned by a SELECT. Because of this, the WFL statement should be used with care.

The ABORT command is similar to the SUPERVISOR version in that the current script execution is aborted. If the ABORT is executed inside a DO file, then no further activities will be executed. The text held in <OPAL string will be displayed as the scripts aborts:

```
#29066 FLEXLIB:ABORT:too many files
#DO interrupted at line 00000620 of DO file ...
```

The EXIT statement allows the current REPORT code to be exited early, passing control to the next file; this is identical to the EXIT statement in SUPERVISOR ODTSEQUENCES.

The QUIT statement is different in that the current scripted function, such as a FILES command, is gracefully terminated but later activities in the script will be processed normally.

PRINT has not been implemented as printing behaviour is controlled with the usual FLEX options and command modifiers. If OP+PRINT is set or the :PR modifier is used, all SHOW statements will be printed.

Some examples: the simple report 'REP TITLE' is identical to:

```
REP BEGIN
  SHOW (TITLE) ;
END;
```

To REMOVE files as they are reported using the WFL function:

```
SELECT DAYS (TODAY,CREATEDAY) Gtr 7
REPORT BEGIN
```

```
WFL ("REMOVE ",FILENAME) ;           % acts on current file
END ;
```

Variables are fully supported in REPORT BEGIN blocks; assignments are permitted as statements as in SUPERVISOR and string or real values stored by assignments are not passed back to FLEX for display.

To email each selected file as an attachment from a FLEX DO:

```
REPORT BEGIN
  $TEXT:="^ATTACH "&TITLE;
  If #RES:=MAIL("To:support@metalog.com;Subj:Req File",
    $TEXT) Lss 0 And #RES Neq -999 Then
    SHOW("Mail failure: ",#RES);
END
```

Practical Usage

Using reports and <selection expressions> makes the FLEX inquiry an extremely flexible tool for file maintenance. Any operation that involves a class of files can probably be done easily and quickly using FLEX reports. Some examples are: generating a job for mass compilations, finding and re-blocking files, generation of COPY or REMOVE jobs using selection expressions to replace (and surpass) SYSTEM/FILECOPY, and custom reports.

For example, the job header could be specified using the REPORT HEAD variant, the layout of the file list given by the REPORT command, and the end-of-job information would be specified using the REPORT FOOT variant (see example). It should be noted that many of the standard reports used in the FLEX inquiry have been made in this way, e.g. FILES, FILES:ATT, FILES:CAT, FILES:ACT, LFILES, and FORGET. The files used are included with the software for your reference.

The <report> is applied when using the following commands: FILES and LFILES.

A simple example of a FLEX session that would produce and start a job to copy all of the files in the *SYMBOL directory except "dangerous" ones to a tape is given here: These commands would be input to FLEX one at a time, or through a FLEX "DO" file. See the STD command for a quick way to access some standard reports and DO files.

```
SELECT NOT (FILEKIND=NEWPSYMBOL OR "MAKEUSER" ISIN TITLE)
REPORT HEAD "BEGIN JOB SYMBOLCOPY; USER=X/Y; QUEUE=23; COPY & COMPARE",
"DUMMYFILE"
REPORT FOOT "FROM SYSTEM(PACK) TO SYMBOLFILES(TAPEPE);END JOB."
REPORT "      ",TITLE
FILES *SYMBOL :FILE JOB/SYMBOLCOPY
WFL STARTJOB JOB/SYMBOLCOPY
```

Complex examples

Examples of two HTML reports are provided here. The generated reports can be

viewed using any web browser. Each report is generated by a Flex do file. A controlling WFL to run both reports is provided.

Each report creates an HTML file and emails it to a nominated address.

There is a zipped Container at [HTML.CON.zip](#)

(Download, UNZIP, Copy to the MCP system and UNWRAP *= OUTOF HTML.CON)

The container contains

```
*EXAMPLES/FLEX/DO/HTML/CODEFILES
A DO file to report on code files older than a specified compiler
version.
*EXAMPLES/FLEX/DO/HTML/FAMDETAILS
A general file reporting DO file
*EXAMPLES/FLEX/JOB/HTMLREPORT an example job to run either report above
and email the resulting file.
```

The first three lines of the job consist of two CONSTANT declarations:

```
CONSTANT BIG=50000,OLD=365;
CONSTANT EMAIL="BOB.1.mail",
              DIR="HTML"
```

These should be changed to suit your needs.

BIG is used to specify the size in sectors of the threshold for 'Big Files'

OLD is the age in days of 'Old Files'

Both of these are used for the FAMDETAILS report.

To state the obvious EMAIL is the email address where the reports should be sent.

DIR specifies the directory where the report files should be created.

In Opal the expression UPPER(TRIM(\$PARAM)) will up-case and trim leading and trailing blanks from a passed parameter. WFL is not quite so powerful so Subroutine TRIMUPCASE performs the same task but with much more code.

The Job takes a string parameter which can be

```
"FAM <family name>"
```

to generate the Family Details Report which lists all Old, Big and sensitive files.

or

```
CODEFILES <level>
```

to produce a Code file report.

Ex.

```
ST *EXAMPLES/FLEX/JOB/HTMLREPORT ("FAM DEV")
```

An example of a report created by can be found at [FAM_DEV.HTML](#)

Or

```
ST *EXAMPLES/FLEX/JOB/HTMLREPORT("CODEFILES 50")
```

Reports all code files on all families with code version < 50

An example of the report created by this is at [CODEFILES50.HTML](#)

The CODEFILE report scans all online families. To restrict it to scan particular families change line 291000 of *EXAMPLES/FLEX/DO/HTML/CODEFILES

```
Files (*)= on =:FL,Byte
```

Ex. for DEV DISK AND PACK change the line to

```
Files (*)= ON DEV, (*)= ON DISK, (*)= ON PACK:FL,Byte
```

The Opal code used for these reports is about as complex as it gets. Please don't be disheartened if you don't understand it all. We are always happy to provide examples to customers to match their requirements. Send any requests to [Support](#).

See also

[Files command](#) [<file option>](#) [<byte option>](#) [<redirect option>](#)
[Opal](#)

RULEs command (Full Flex Only)

This command is described in the Full Flex reference manual

SAMe command

— SAME —————|

The SAME command is used in conjunction with the FILES command to allow the re-display of a page of files. This command may be entered when waiting at a page break during a FILES command. The SAME command may be used in conjunction with the FILES> syntax.

This command is equivalent to both NEXT+0 and NEXT-0 .

See also:

<FILES command> [<NEXT command>](#)

SAVE command

— SAVE —————|
 └ AS ─┘

The SAVE command is used to save the current Selection expression and the current Reports (including head and foot). A Flex DO file is created with the title

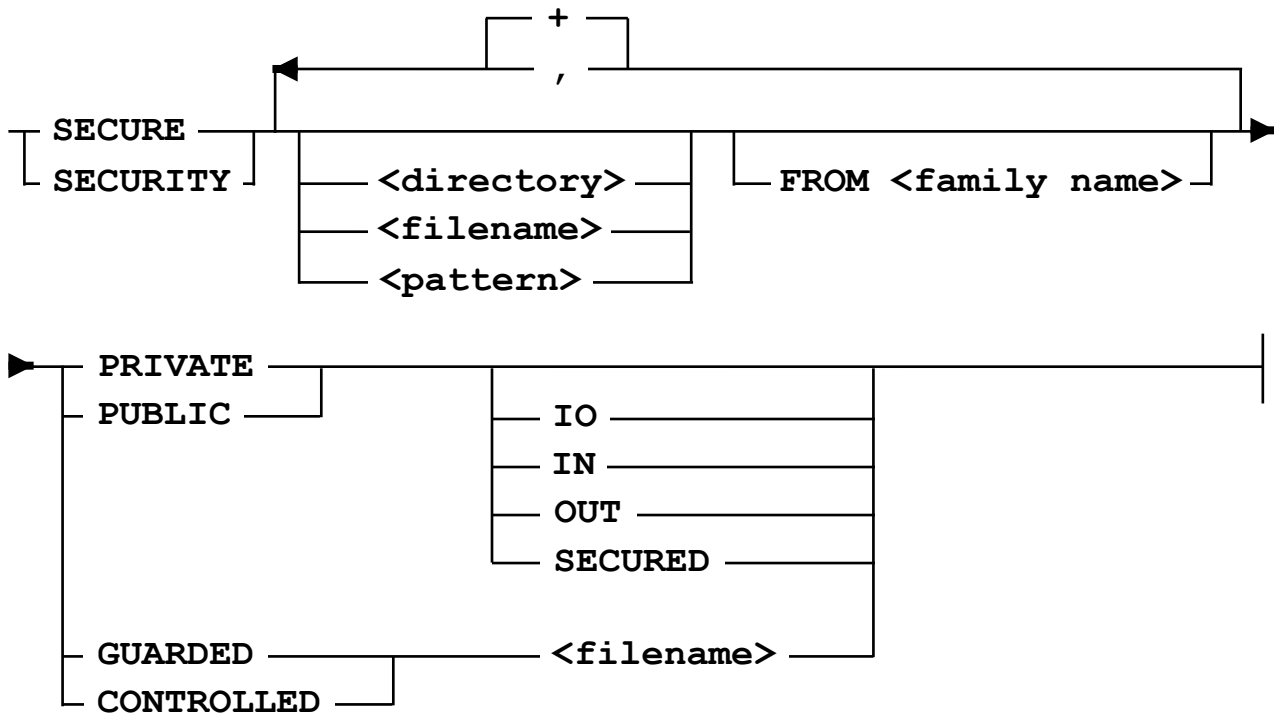
specified. A DO of this file would reinstate the Selection and Reports to their state when the SAVE was entered, this state also includes whether they are active or not. (SEL- etc.).

Note that the AS is optional; if no filename follows a SAVE AS command then the file title AS will be assumed.

See also

[DO command](#)

SECure command



The SECURE command is used to change the security of a directory or a single file. It will NOT change the security of a file that already has the correct security attributes; in particular, it will not change the security (and therefore the time-stamp) unless the security actually needs to be changed. Note that a selection expression may be specified to further limit the files to be considered.

Only a privileged user or a user with the Granulated privilege CHANGESEC may change the security for a <directory>, or the security of another user's files.

See also:

[<filename>](#) [<directory>](#) [quotes](#) [<usercode>](#)

SElect command

| | |
|------------|-----------------------|
| — SELECT — | |
| | — + — |
| | — - — |
| | — ? — |
| | <opal string> _____ |
| | FROM <filename> _____ |

The SELECT command is used to establish conditions that a file or file entry must meet to be selected when using a directory search command in FLEX.

There are two ways to specify a <selection expression>: either from a file or directly as a part of the SELECT command. The file should be a JOBSYMBOL file that contains the text of the <selection expression>, and may be formatted in any way desired. Any text appearing to the right of a % sign is treated as a comment and is not processed by the compiler.

If no text appears after the SELECT command, and there is a CANDE workfile, FLEX will try to compile a <selection expression> from the workfile. This can be useful when debugging a complicated <selection expression>.

In both cases, if the <selection expression> compiles OK, it will be stored as the current selection expression. This selection expression will be applied in each of the subsequent file related commands of the FLEX inquiry.

'SELECT -' will temporarily disable use of the current <selection expression>, without losing the expression. 'SELECT +' re-activates the current <selection expression>. The current <selection expression> can be interrogated by 'SELECT ?'.

Using <selection expressions> can greatly reduce the amount of output for a given request, making visual examination easier. Note that the amount of time required for a file search is only slightly reduced by using a <selection expression>; FLEX must still examine each file or entry and determine if it fits the condition.

Selects are defined using the Opal language. A full description of the Opal language is available in the Opal Programming Reference Manual. A brief description can be found in the [Opal Section](#) of this document.

Example

```
SELECT FILEKIND=JOBSYMBOL
SELECT DAYS (ACCESSDAY, TODAY) > 365
SELECT SEGMENTS > 500000
SELECT PPED
SELECT CODEFILE AND CODEVERSION < 49 %MCP 4.9std
```

The <selection expression> will be applied when using the following commands:

BACKUP, [FILES](#), [LFILES](#), [LOAD](#)

SPLit command

— SPLIT —

The SPLIT command causes FLEX's current LINE file (used for report printing) to be closed and released to the Print System as a print request.

STart command

— START —

 <file name>

 ON <family name>

 (<parameter list>)

This command is analogous to the CANDE START command, and treats the <filename> as a WFL job, which it then does a STARTJOB against.

However, when it is run interactively with a file with parameters, and no parameters are provided, FLEX will prompt the user for the value of the parameters.

STD command

— STD —

 <identifier>

The STD command allows quick access to standard reports. By searching the usercode and system directories for the files in the directory METALOGIC/ FLEX/ DEFAULTREPORTS/ <name>/= , it can present a menu of the available reports or DO files.

One possible output from the STD command might be:

```
The following default reports are available:
1. *DU                2. *ARC                3. *HELP
4. *CLEAR             5. *CRUNCH             6. *COMPGEN
7. *PACKDIR           8. *CLEANSTD           9. *CODEATTS
10. *BACKUPPRINT      11. *COSTJUSTIFY
Enter a number from 1 to 11 or 0 to exit
An asterisk (*) indicates that the report is a system file
```

Several of these warrant special mention. The PACKDIR report is designed to be used with the <print option>, and produces an output similar to the FILEDATA PACKDIR report. The FLEX report is more compact, and can be shown for a limited

The CODEATTS report shows a screen similar to the LFILES screen of FLEX, but where the attributes are particularly applicable to codefiles. To use this, enter:

then

```

Title      : (BOB)OBJECT/AC/DB/INQ
-----
General attributes      Codefile info attributes      Boolean attributes
-----
Filekind :          DMALGOLCODE      Executions:          0      CPed          : FALSE
Segments :             18      Priority :          0      PPed          : FALSE
Codeversion:          53.170      Parameters:          0      PPedtransprnt: FALSE
Cycle/version:        001:0000      Restarts :          0      RPed          : FALSE
Created 13/03/08 @ 14:55:55      Nonexecutable:        FALSE
Accessd 13/03/08 @ 14:55:56      Systemfile:          FALSE      Unsafe          : FALSE
Altered 13/03/08 @ 14:55:55      Language :          DCALGOL      Suppressed      : FALSE
Changed 08/12/08 @ 10:30:54      Sharing :          LIBDONTCARE      Inuse           : FALSE

Security Information      Capable?      Limits      Stack size info
-----
Security type :          PRIVATE      DMS:   Y      MaxIO:          0      D1stack:          10
Security use :             IO      IPC:   N      MaxPROC:          0      D1WSavg:          0
Protection :             SAVE      SORT:  N      WSlimit:          0      D2stack:          1536
Guardfileset :          FALSE      LIBR:  N      WSavg :          0
Security admin :          FALSE
Secadmin transp:          FALSE

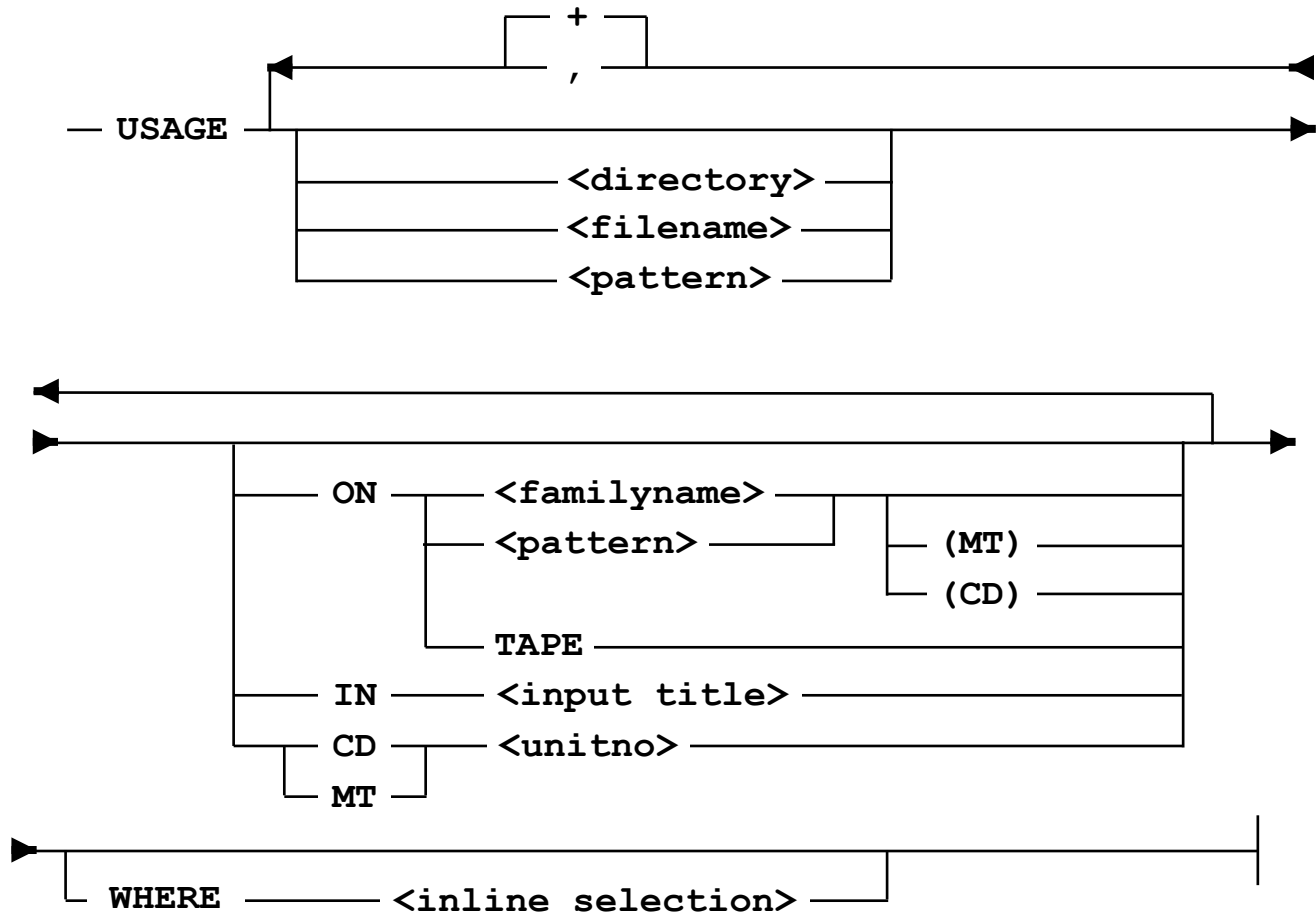
6795      Push SPCFY for more. XMT to break or NEXT +/-4      09:31:12

```

The user can create standard reports under their usercode in the METALOGIC/FLEX/ DEFAULTREPORTS directory, and these will be found automatically. A privileged user (e.g. the site manager) can place standard reports under the *METALOGIC/FLEX/DEFAULTREPORTS directory, and these reports will be available to all (depending on the security of the report files). FLEX will interpret these files based on the suffix appended to the title. As an example, suppose we had an action called "FLIST" that we wished to do frequently. Further, assume that this is complicated enough that it requires a REPORT, a REPORT HEAD, and a DO file to accomplish. Then the files should be named:

Each of these files should be in the appropriate form to be read from a file. For example, in the REPORT case, only the OPAL text for the report should be present (don't put "REPORT" as the start of the file). This is the format required for the "REPORT FROM <file>" syntax.

USAGE command



The **USAGE** command produces a report showing number and sizes of files in the given **<directory>**. In addition, if the **<brief option>** is not used, exception conditions such as the number of files without backup, and a breakdown by **FILEKIND** will also be shown. This command is useful to provide a quick snapshot of file and disk usage.

The full report is very similar to the report produced for each user by **FAMILYMANAGER**. Three colon options are particularly important with **USAGE**. The **<FLAT option>** and the **<LEVEL option>** and the **<brief option>**.

The **FLAT** option allows for a very fast response for large numbers of files. (N.B. By default, **FLAT** requires a privileged usercode or the Granulated privilege **GSDIRECTORY**.)

The **LEVEL** option allows for generating directories automatically. The effect of **':N'** is to do first a **FILE:N**, then apply the **USAGE** command to each directory found. Note that the other options only take effect after the **LEVEL** option, so using **FLAT** with **LEVEL** will lead to as many runs through the directory as there are directories returned by the preliminary **FILES:N** stage. **USAGE:N** can be very powerful when combined with a **SELECT** condition.

The **BRIEF** option gives a one line report for each directory. This is the report normally used when the **LEVEL** option is also used.

The full Usage report shows the title of the largest file and gives totals of space wastage. The 3 categories of wastage are:

- "Mis-aligned Blocks" - caused by blocksize not being a multiple of 180 bytes.

- "End Block" - caused when the last block is not full.
- "Uncrunched" - the space released if all were crunched.

Examples:

USAGE report for your own usercode
USAGE *BD/= ON BKUP:BRIEF one line report on BD directory
SEL CODEFILE AND PDPPED(TITLE)
USAGE (=)=:1,FLAT,BRIEF report for all PPED code files under usercodes by usercode.

```
Usage (=)= where codefile and pped:fl,br,1
Usage Report for (=)=
-----
Directory                Files      MegaBytes      Largest
-----
(BOB)=                    2         0.05         0.03
(IPP)= - - - - -          8         6.61         3.24
(471)=                    1         0.29         0.29
(SHAFT)=- - - - -        1         0.53         0.53
-----
TOTALS                    12         7.48         3.24
#
```

2604 Metalogic Flex Version 58.580.014 12:42:02

See also:

[<BRIEF option>](#) [<FILE option>](#) [<FLAT option>](#) [FILES command](#)
[<LEVEL option>](#) [<PRINT option>](#) [STD command](#) [<filename>](#)
[<directory>](#) [quotes](#) [<usercode>](#)

USER command

```
[ USER _____ <usercode> _____ ]
[ HELLO ] [ = ] [ / <password> ]
```

The USER command allows the FLEX user to change usercodes for the current FLEX session. If a valid usercode/password is given, FLEX will use the new usercode as the default for all file searches, options, and in deciding privileges. If a <password> is required for the given <usercode>, it must be supplied. If an incorrect <usercode>/<password> is given, the program will be DS'ed. An <accesscode> may be supplied after the usercode.

Examples:

```
USER KHAN/GHENGHIS  
USER RIPPER/JOHN ; ACCESS = JACK/KNIFE
```

See also :

[<ACCESS command>](#)

WFL command

– WFL _____ <valid WFL statement> _____|

The WFL command is used to send WFL statements to the Work Flow Language compiler. The parameter is passed directly to the WFL compiler, without any syntax checking. The <nomsg option> may be used to prevent messages from the job appearing at the terminal, the <syntax option> may be used to check the validity of the parameter, without actually executing the job.

See the Work Flow Management User's Guide for more information.

Example:

```
WFL START MY/JOB  
WFL COPY&REMOVE TEST/FILE TO ARCHIVE (PACK)
```

See also:

<nomsg option> _____ [<syntax option>](#)

WHAT command

– WHAT _____|

The WHAT command identifies the program and version, and tells you the name of your workfile, if any. It also provides information about the current environment where the program is running. It is useful when running with the <audit option> on or when sending results to the printer as a way of showing the conditions of the run.

Example:

```
what
#Utility File Handler Version 53.530.28 8:36 AM Friday, February 6,2009
# Using FlexLib:53.530.16 Opal:53.530.48 GSTables:53.530.37 Magus:53.530.17
# Enter 'HELP' for instructions
# No worksource
# Usercode: BOB LSN(262)
# Destname: SITE Hostname: DELL MCP (cataloging)
# FAMILY DISK = DEV OTHERWISE DISK
#
6774 Utility File Handler Version 53.530.28 08:36:43
```

Write command

— WRITE —
 |
 | **<file name>** |
 |
 | ON **<family name>** |

Like LIST:PRINT

DESTROY command

DESTROY
ZAP —
 |
 | + |
 | , |
 |
 | **<directory>** |
 | **<filename>** |
 | **<pattern>** |
 | FROM **<family name>** |

The DESTROY command is used to eliminate a file or list of files PERMANENTLY from the system, by Removing the file from disk AND Catalog Purging the backup information from the catalog.

The specified files will be offered one at a time for Destroying, and if an 'OK' response is given, the indicated file will be Destroyed. Other valid responses are QUIT, to stop the search, or any other non-blank input to skip the current file. DESTROY is also a valid action in the action screen.

To delete the backup information but leave the file resident, use the PURGE command.

If no 'FROM <familyname>' part is specified, the current primary family will be assumed.

If the <CANDE option> is set, all of the above actions will also be applied to the corresponding OBJECT file, i.e. if the file XYZ is released, then OBJECT/XYZ will also be released if the <CANDE option> is set.

The DESTROY command has quite a drastic effect on a file, in that it makes the file difficult or impossible to retrieve. Note that since Purging the catalog information does **NOT** have any effect on the physical volume where the file may be backed-up, recovery is possible in certain cases where an accidental DESTROY has occurred; namely, if the file had a backup, and an old copy of the SYSTEM/CATALOG is available, the site manager could use the <linear option> of the FILES command to search for the SERIALNO of the backup volume. This process is fairly annoying, however, so it is best to be sure before you DESTROY a file.

See also:

| | | | |
|---------------------------------------|---------------------------------------|--------------------------------------|-----------------------------------|
| <twobackups option> | <action option> | <cande option> | REMOVE |
| <syntax option> | <linear option> | <filename> | <directory> |
| quotes | <usercode> | PURGE | |

Extra features for cataloging users

This chapter is to describe the many advantages of our product FLEX Inquiry to customers using Unisys' Large Systems cataloging. METALOGIC has been involved for over twenty years with Unisys' cataloging. We have put more new users on cataloging in recent years than any other organisation, including Unisys. We believe strongly in Unisys' cataloging - so much so that one of our major products, FLEX, uses cataloging as a base. Our full FLEX package is the most powerful disk and tape management available on **any** system - by a long margin. A non-trivial part of this is due to the real-time capabilities and MCP integration that cataloging provides. One of the major components of the full FLEX package is the user interface utility, FLEX Inquiry. This is available as a separately priced product, and, by itself, provides many features that are of special interest to cataloging sites. We feel that the particularly strong advantages it provides for cataloging users should be highlighted here.

LOAD

FLEX Inquiry can automatically generate jobs to reload non-resident files whose backups have been made using COPY & BACKUP (using the full FLEX package or FILECOPY). All that is needed is the filename, a list of filenames, a directory, or any combination of these. A generation may be specified. FLEX Inquiry will determine the minimum number of tapes needed to satisfy the request, and generate appropriate LIBRARY/MAINTENANCE jobs. No further user intervention is required, and the operator only needs to put up the tape or tapes with the indicated serial numbers.

FILES :LINEAR

The current or and old SYSTEM/CATALOG can be searched for any file or files that satisfy a selection expression. This search spans all of the packs in a single call, and is approximately **1300 times** faster than equivalent GETSTATUS calls. For example, a directory search that ordinarily would take one hour and twenty minutes can be done in 11 seconds. This search can filter out files based on a complex boolean expression of any file attribute(s) that can be found in the catalog block.

"UN-PURGE" or oops! function

Because FLEX Inquiry can use old catalogs for its searches, the ultimate "oops !" function can be provided by making periodic copies of the CATALOG to another pack. If a file is erroneously Catalog Purged and Removed from disk, the old catalog can be searched for the file and the serial numbers of the tape(s) that it is backed-up on. This can be a lifesaver.

PDT BU MT <serialno list>

This command prints out a list of all the files with backup references to any of the volumes with serial numbers in the given <serialno list>. This list can be generated at the speed of the FILES:LINEAR command.

DELETE

If you have ever had trouble remembering the syntax of the CATALOG DELETE statement, all you need is a Unisys' TD830-compatible terminal. FLEX Inquiry uses the special features of these terminals to present a screen with each of the generations of a file and the corresponding catalog information. To delete a generation, simply move to the box which shows the generation you wish to delete, and press SPCFY. FLEX Inquiry programmatically Catalog Deletes the generation.

RELEASE & DESTROY

The RELEASE command is a "safe" remove, in that it will only remove a file if it has an appropriate number of backups. The DESTROY command is a combination of REMOVE and CATALOG PURGE, to allow permanent cleanup of files. Both of these commands are available from the action screen - the menu of files for Unisys' TD830-compatible terminals, obtained through FILES :ACT in FLEX Inquiry.

LISTVL

This facility allows practical use of the Catalog and Volume Library as a simple tape library system. The first list shows all of the volumes in the Volume Library, sorted by serial number, together with all of the information obtainable by PV requests on a single line. In addition, the number of files for which each volume is in use as backup is given. The second report shows them sorted by name. A list of scratch entries is given. Volumes that are in use as backup, but not in the volume library, are listed. A list of expired volumes that are not in use as backup is given, and a page of statistics on the catalog is given. In addition, short reports may be generated that show only expired volumes, or only tapes that are used as backup for a small (specifiable) number of files. All of these reports can be run against an old CATALOG.

MAINT & Diagnostic functions

Several diagnostic tools are provided for examining the catalog. All of these functions can be run against an old CATALOG. The most useful function in this group allows the volumes in the volume library from another machine to be merged into the current live CATALOG. Even more useful is the ability to merge families from one machine to another, or to consolidate families on a single machine **without losing the backup information !**

MAINT CHECK VL may be run to identify & report on volume library blocks that are

illegal, etc.

An Overview of the OPAL Language

This section is to give you a feel for the OPAL language. It is not a comprehensive treatment; you must refer to the OPAL Language Manual for full technical details. In fact, because of OPAL's similarity to WFL or ODT statements, this section may be all that you need to study to write working OPAL programs.

OPAL is a language that is based largely on WFL. It is designed to allow for easy specification and solution of operational problems — in fact the name OPAL stands for OPerations Algorithmic Language. OPAL was created because operational problems always seem to have exceptions, or need to be changed at a rate that would swamp most DP departments. In addition, the information available from the system often needed a systems programmer to retrieve the data; such programming skill is expensive.

OPAL addresses these problems by making system information available as keywords or attributes in the language. These can be put together in strings, boolean, or arithmetic expressions to state and solve operational problems. In FLEX Inquiry, there are two general types of problems that are addressed by OPAL: selecting a group of files, and reporting in a flexible way.

Selecting is done by specifying a boolean expression via the SELECT command. A `<boolean expression>` is just a statement that is either true or false. In FLEX, each file is tested against the boolean expression, and if the expression is true, the file will be returned to the user — either to the screen, a file, or the printer.

The best way to get a feel for selection expressions is to see some examples. These are actual OPAL expressions, and can be passed directly to the SELECT command:

```
SEGMENTS > 5000 AND ROWSINUSE GEQ 10
MAXRECSIZE = 14 AND NOT INUSE OR FILEKIND=ALGOLSYMBOL
DAYS(TODAY,USEDAY) < 2 AND SEGMENTS < 50000
"RESUME" ISIN TITLE AND FILEKIND = {TEXTDATA,DATA,SEQDATA}
```

Each of these examples would select a subset of all of the files in a given directory. Note that in addition to attributes, there are OPAL functions to allow some manipulation of the attributes, e.g. "DAYS" and "ISIN". The string and date handling functions are very full-featured.

The reporting part of this is very similar; strings are built up, as in WFL, from the language components and various string operators. Some examples of short reports are:

```
TITLE
", " , TITLE
TITLE&" : "&FILEKIND&" ("&SEGMENTS&") "
```

A very rich set of examples can be found by listing the files in the directory *METALOGIC/FLEX/DEFAULTREPORTS, which is supplied on the FLEX release

tape. These show both reports and selection expressions. It should be pointed out that most of the output from the FLEX Inquiry is generated by soft reports written in OPAL (these are some of the examples). The remainder of this section will be a quick reference guide to some of the more common functions in OPAL. For a listing of all of the available attributes enter "HELP ATTS" in FLEX. HELP ATTS will also accept wild cards.

SUMMARY of some OPAL Language components

- The usual relational operators: <, >, GEQ, LEQ, =, IS, ISNT
- string operators EQL, NEQ, ISIN, HDIS, TLIS and INCL similar to those in ERGO
- lists, i.e. MAXRECSIZE = {14,52,420} means that the maxrecsize is one of 14,52 or 420
- boolean operators: NOT, AND, OR, EQV, IMP, XOR
- USERDATA(<locator id>), where <locator id> is any node in the USERDATAFILE, returns the appropriate attribute [privileged only]
- CODEFILE, SYMBOLFILE, and DATAFILE — shorthands for filekind classes
- CASE expressions, e.g. CASE EXTMODE OF (3,3,5,9,10)
- MIN, MAX, ABS, NABS, +, -, /, *, DIV, MOD arithmetic functions
- LENGTH, STRING, HEX, OCTAL, TAKE, DROP HEAD, TAIL, REPEAT string functions
- FILEIDS(<title>) returns the number of identifiers or levels in the title
- FILEID(<title>,n) returns the nth file identifier in the title. If n=0, the usercode is returned
- MEMBER(<target>,<expression list>, <optional ELSE>) returns the case number where <target> occurs in <expression list>
- SYSTEM(x), where x is SERIALNUMBER, TYPE, or MCPLEVEL, as in WFL
- DAYS(<julian date 1>, <julian date 2>) gives the number of days between the two dates, respecting leap years, etc.
- TIMEDATE(x), where x is one of: HHMMSS, YYYYMMDDHHMMSS, DISPLAY, MONTH, DAY, DAYNUMBER, YYDDD, YYMMDD, MMDDYY, DDMMYY, YYYYDDD, YYYYMMDD, MMDDYYYY, DDMMYYYY
- DAYINWEEK, DAYINMONTH, WEEKINMONTH, WEEKINYEAR, YEAR, DAYINYEAR, MONTH, TODAY
- DAYNAME(<integer>) and MONTHNAME(<integer>)

- TIME(x) converts "clicks" to a display string
- special character lookup, e.g. #[ESC], #[DC1], #[SUB], etc.
- DECAT (derived from PL/I) to pull apart strings
- NUMBER(<integer>) returns a text representation of the number
- TDPAGE and TDADDRESS for special TD830 handling

Replacing FILECOPY and Snap Backups

This section is for site managers only. It describes a method for using FLEX Inquiry to speed up the process of making safety backups instead of SYSTEM/FILECOPY.

At many sites, the standard way to make backups of key files is to run SYSTEM/FILECOPY each night during the week to find all of the files that have been updated since yesterday, then copy them to a LIBRARY/MAINTENANCE tape. On the weekend, a larger run of FILECOPY would back up every file on the pack. In some cases, some simple exclusion is added.

All of this is well and good, except for the time that FILECOPY takes to find the files that satisfy the "updated yesterday" criterion. On a pack with 10000 files or so, this can take over an hour, before a job goes in to start copying the files. The equivalent run using FLEX Inquiry takes less than 15 seconds !

In addition to the speed improvement, the selection capabilities of FLEX Inquiry are much more versatile than those of FILECOPY. The job created by FLEX is an ordinary WFL job, and it does an ordinary copy to a LIBRARY/MAINTENANCE tape. The speed improvement is entirely in the scanning for the appropriate files.

Simulation of FILECOPY

FLEX can take its input from a file, as can FILECOPY. Two example files that simulate a FILECOPY updated run is on the release tape, and are called METALOGIC/FLEX/DEFAULTREPORTS/FILECOPY/=. A more common selection expression might be `SELECT DAYS (TODAY,USEDAY) < 2` ; this would find all files that have been updated yesterday. FILECOPY can also select files that have been created or updated recently; however, it can only do one type of request at a time. Similar selection expressions in FLEX might be:

```
SELECT DAYS (TODAY,ALTERDAY) < 2
SELECT DAYS (TODAY,CREATEDAY) < 2
```

but it can also do both:

```
SELECT DAYS (TODAY,USEDAY) < 2 OR DAYS (TODAY,ALTERDAY) < 2
```

or some combination of these:

```
SELECT DAYS (TODAY,USEDAY) < 2 AND NOT DAYS (TODAY,ALTERDAY) < 2
```

The BETWEEN syntax is just an "AND" in FLEX:

```
SELECT DAYS (TODAY,USEDAY) > 2 AND DAYS (TODAY,USEDAY) < 10
```

The directory to be searched is restricted through the "FILES" command used. For example, after applying the selection expression and report, entering

```
BACKUP (SMITH)=
```

would restrict the search to files under the SMITH usercode. The files would be

copied to a tape named FLEXBACKUP<yddd>. Of course, to get the speed advantage over FILECOPY, we need to use the <FLAT option> as follows:

```
BACKUP TO SMITH090209 (SMITH)= :FLAT
```

This option requires that FLEX is run from a privileged usercode or a usercode with the Granulated privilege GSDIRECTORY, if the directory being scanned is not yours. To include or exclude files, add a term to the selection expression. For example, to include all files with "AUDIT" in the title, and exclude all files of filekind MCPCODEFILE or that are over 10000 segments in size, we could use:

```
SELECT DAYS (TODAY,USEDAY) < 2 AND "AUDIT" ISIN TITLE  
AND NOT (FILEKIND = MCPCODEFILE OR SEGMENTS > 10000)
```

An example FILECOPY replacement scheme is provided:

EXAMPLES/FLEX/FILECOPY/DO is a parameterised do file. The do file contains a selection expression which excludes some file kinds:

```
Not Filekind={BACKUPPRINTER,DBDATA,CATALOG,DBRESTARTSET,FTAUDIT,FTPDATA,  
KEYSFILE,SYSTEMDIRFILE,XDISKFILE,PROMBURNERDATA}
```

This list should be edited to suit your needs.

EXAMPLES/FLEX/FILECOPY/JOB is a WFL which can be used to Run Flex with the FILECOPY DO file.

The Job takes a string parameter which can be a command followed by a backup ID. The command can be FULL, DIFF or INC. Indicating a Full backup, a Differential backup or an Incremental backup. In the default case the ID will be a familyname and all files on that family will be backup up to a tape named <familyname><yymmdd><type letter>. <type letter> is F for Full, I for incremental and D for differential.

Code at the end of the job can be changed to select different directories for an ID.

```
IF ID ="USERDEV" Then  
DIR:=" (=) = ON DEV"  
ELSE  
DIR:=" (*) = ON " &ID;
```

In the sample code we specify that an ID or USERDEV will back up all usercoded files on family DEV.

Examples of Starts of the job:

```
ST EXAMPLES/FLEX/FILECOPY/JOB("FULL DEV")  
would backup all non excluded files on DEV to Tape DEV090209F  
ST EXAMPLES/FLEX/FILECOPY/JOB("FULL USERDEV")  
would backup all non exclude usercoded files on DEV to Tape  
USERDEV090209F
```

The date and time of the most recent backup of each type for each ID is held in permanent variables. This allows an incremental backup to select all files modified since the most recent incremental or a differential to select all files altered since the

last full backup. The Do file EXAMPLES/FLEX/FILECOPY/DO/CHECK reports the date and time of the most recent backup of each type for each ID.

Ex. from Flex DO EXAMPLES/FILES/FILECOPY/DO/CHECK:

```
Filecopy backup report
Id      Full Backup      Differential      Incremental
DEV      06/02/09 12:28:42.719 06/02/09 15:23:53.328 06/02/09 12:21:18.297
DEVBOB   Never                Never            06/02/09 15:31:49.531
DEV_A    06/02/09 15:46:23.156 Never            Never
PRIVATE  09/02/09 10:56:07.141 09/02/09 11:27:37.328 09/02/09 11:27:54.484
#
```

One part of FILECOPY that cannot be simulated in FLEX Inquiry is the creation of partitioned tapes. If you frequently make multi-reel backups, we suggest that you consider the full FLEX package. The FAMILYMANAGER program therein actually makes single reel tapes, can take multiple selection expressions as part of an artificial intelligence expert knowledge system, and can effectively partition the pack by usercode.

Copyright

Copyright © 1979-2020 Metalogic S.à R.L., Luxembourg.

All rights reserved.

Last revision: Apr 2020

METALOGIC believes that the information described in this manual is accurate and reliable, and much care has been taken in its preparation. However, no responsibility, financial or otherwise, is accepted for any consequences arising out of the use of this material.

The information contained herein is subject to change. Revisions may be used to advise of such changes and/or additions.

ACKNOWLEDGEMENTS

To John Hurley, for conceiving FLEX Inquiry and backing the idea with 3 years of hard work.

To the other implementers: Bob, Brian, David, Don, and Jeff.

As always, some of the best ideas did not come from the implementers. To all of you who contributed, many thanks.

M E T A L O G I C S.à R.L.

PO Box 11

L-7508 Lorentzweiler

Luxembourg

<http://www.metalogic.eu.com>