

Flex Site Management

Reference Manual

Relative to versions

Flex **590.01**

FamilyManager **590.01**

Mergetape **590.01**

Recent Changes6

Mar 09 Flex:530.36 Better RULES MANAGER info	6
Jan 09 FMgr:530.05 Remove Summarise and Analyse	6
Jan 2009 FMgr:530.04 De-imp Discard	6
Dec 2008 FMgr:530.03 Inherit JOBSUMMARYTITLE.....	6
Jul 2007 FMgr:520.11 New parameter:Prefix	6
Jul 2007 MTpe 520.03 New Prefix parameter	7
Apr 2007 FMgr:520.07 New Option:HIDEDONTBACKUP	7
Mar 2007:Flex 520.03 Report Rules Opal version	7
Dec 2006 FMgr:510.11 More consistent naming for Jobs	8
Aug 2006 MTpe:510.16 CATALOG PURGE of alt CD image.....	9
Jun 2006 MTpe:510.14 Automatic LOCK of FLEX volumes.....	9
Dec 2005 FMgr:510.06 Append continuation tapes	10
Nov 2005 FMgr 510.05 Prevent syntax errors	10
Sep 2005 MTpe:510.10 New IGNORE option	10
Mar 2005 MTpe:510.06 Restrict COPYREPORT option.....	11
Mar 2005 MTpe:510.04 New MB LIMIT ©REPORT.....	11
Mar 2005 FMgr:510.01 New parameter:SERIALNO OUT	11
Mar 2005 MTpe 510.03 Replace feet by megabytes	11
Mar 2005 MTpe:510.02 Boolean params & Density	12
Feb 2005 MTpe:510.01 MERGE =	12
Dec 2004 MTpe:500.20 REPORT =.....	12
Dec 2004 MTpe:500.19 New 'SERIALNO OUT'	12
Nov 2004 Flex:500.08 OPTION FOR LIBMAINTDIR	13
Nov 2004 FMgr:500.10 Check CopyWrite Library Status	13
Nov 2004 MTpe:500.17 COPYWRITE CHECK	13
Oct 2004 MTpe:500.16 LIBMAINTDIR FOR TAPE LOADS.....	14
Sep 2004 MTpe:500.15 FARMS v SCRATCH POOL	14
Sep 2004 MTpe:500.14 ALTERNATE IMAGE SPEC	14
Jun 2004 MTpe:500.09 Don't re-release Disk Farm files.....	14
Jun 2004 MTpe:500.07 No Compare for CopyWrite.....	15
May 2004 MTpe:500.03 Require Rules Manager user code	15

May 2004 FMgr:500.07 Alternate Family for CDIMAGE	15
MAY 2004 MTpe:500.02 CDIMAGE Merging	16
Nov 2003 FMgr:490.04 Better validation on Initialisation.....	19
Introduction.....	20
Overview	21
FLEX rules	22
About this Manual	22
Recent enhancements	22
Cataloging Overview	24
Volume Library	24
Getting Started.....	27
Utilities overview	27
METALOGIC/FLEX/LIBRARY	27
OBJECT/FLEX	27
Example scripts and WFL jobs.....	27
METALOGIC/FLEX/FAMILYMANAGER	27
METALOGIC/FLEX/MERGETAPE	28
METALOGIC/COPYWRITE.....	28
Installation.....	28
Setting up FLEX and CATALOGING	30
Setting up Cataloging.....	30
Setting the Cataloging option.....	30
Volume adding Families.....	32
Create a Rules File.....	33
Backup Strategy	36
Choosing a backup strategy	36
Multi or single family pack dumps	36
Backup Media	36
Tapes	36
Disk farms	36
CD.....	36
CD Images	36
FLEX Rules	38
RULES Base.....	39
RULES Manager.....	39
Syntax of Rules.....	39
<Rule Action>:.....	39
<Rule Message>:	41
<Rule Expression>.....	42
Rule Evaluation	42
Rules Examples	43
Global Rules.....	44
Family Rules	44
User Rules	44
RULES command	44
Global rules	45
Summary of RULES Commands	46
RULES - User variants.....	46
RULES CONFIRM	47
RULES ENTER.....	47
RULES LIST	48
RULES OP	49
RULES SAVE	51
RULES TEST	51

RULES - Group Leader variants	52
Group RULES CONFIRM	52
Group RULES ENABLE	53
RULES GROUP	53
Group RULES SHARE	54
RULES - Manager variants	54
RULES FORGET	56
RULES MANAGER	56
RULES THRESHOLD	56
FAMILYMANAGER	58
Commands	59
<family list>	59
<media>	60
<options>	60
<boolean setting>	61
BACKUP Command	61
CLEANUP Command	61
DUMP Command	61
TEST Command	61
Media types	62
CD	62
CD Images	63
RULES	65
Running FAMILYMANAGER	67
Tape Backups	68
Bigfiles and History	70
FAMILYMANAGER commands	71
SELECT modifier	72
Operational Considerations	74
CHECK files	75
How Rules are Evaluated	76
Segment Limit Processing	78
The RULES THRESHOLD	78
Basic Shares	79
Group Definition	79
Advantages of SHARES	80
How Limits are applied	81
Assignment Of Serial Numbers	82
Rules Debugging Techniques	82
COPYWRITE and FLEX integration	83
MERGETAPE	88
Running MERGETAPE	89
Examples	89
Syntax	90
<merge options>	90
<select specs>	90
<except list>	91
<serialno options>	91
<true/false>	91
<release specs>	91
Tape Merges	91
<merge specs>	93
CD	93
CDIMAGE	93
DETAIL	93
DRIVES	93
LOADS	93
FEET/MB	93
SHORTCOPY	93

WORKPACK	93
AUTO	94
AUTO Defaults	94
DRIVES	94
NOSHORTCOPY	94
DETAIL	94
LOADS	94
LIMIT	94
CDImage Example	94
Syntax of the string parameter	94
< specs >	95
<selection/release>	95
<serial list>	95
<selection>	95
<merge specs>	95
CD	95
CDIMAGE	96
DETAIL	96
DRIVES	96
LOADS	96
FEET/MB	96
SHORTCOPY	96
WORKPACK	96
AUTO	96
RELEASE Command	97
<filter>	97
< specs >	97
NONE	97
WASBACKUP	97
ZERO	97
VIA	97
STATS	97
TAPELIB	97
Default	98
USERLIB	98
<selection/release>	98
RELEASE	98
GEN	98
EXCEPT	99
<serial list>	99
SELECT	99
<selection>	100
UNSAFE	100
OFFLINE	100
AGE	100
VIA USERLIB	100
MERGE 'Command'	100
<merge specs>	101
OFFSITE	101
TIMEOUTS and Interactive Merge	101
MERGETAPE reports	102
MERGE dialog report	102
Volumes released by MERGE command	103
Summary Report	103
Merging Process	103
Scanning phase	104
Merge phase	106
Loading Files (Tape Merge)	106
Loading Files (CDIMAGE Merge)	107
Recovery when Loading Files	108
Creating merged tapes	108
Release phase	109
Error Messages from MERGETAPE	110
Merging Tapes	110
Merging from Disk farms	111
Merging CDs	113

Merging CD Images	114
Configuring FLEX	115
FLEX DEFAULT command.....	115
Install utility CONFIG menu	115
Migration.....	116
Flex Recovery Features	117
Case 1. You have lost one member of a multi-pack Family:.....	117
Case 2. You have lost a whole disk family:	117
Case 3. You have lost the Halt-Load Family:	118
Case 4. You have lost the DL CATALOG Family:	118
Case 5. You have lost the entire system	118
Case 6. You have lost the entire system and tape library.....	119
Copyright	120

Recent Changes

Mar 09 Flex:530.36 Better RULES MANAGER info

The RULES MANAGER command has been improved to allow better control of the setting of primary and alternate Rules Manager usercodes and accesscodes. Changing or setting a Rules Manager requires the usercode and accesscode to be valid in the USERDATAFILE and passwords must be supplied if applicable. All Rules Manager usercodes must be marked as privileged (PU).

The command 'RULES MANAGER ?' will show current Rules Manager details if run by a FLEX usercode that is PU or SECADMIN.

A typical response might be:

```
RULES MANAGER ?
-- Rules Management details --
Master Usercode:      FLEX
Master Accesscode:    RULES
Alternate Usercode:    Not Set
Alternate Accesscode: Not Set
#
```

Further details and syntax can now be seen using the HELP RULES MANAGER command in FLEX Inquiry.

Jan 09 FMgr:530.05 Remove Summarise and Analyse

The Flex Usage command has long been a better way to do this. With the recent addition of wastage info, there is no good reason to use Analyse and Summarise in Familymanager, so they have been deleted.

Jan 2009 FMgr:530.04 De-imp Discard

Load:Flat <familyindex> in Flex Inquiry is a better way to do the DISCARD function, so it has been de-implemented in FamilyManager, saving about 5% in the source and object file sizes.

Dec 2008 FMgr:530.03 Inherit JOBSUMMARYTITLE

Setting the JOBSUMMARYTITLE attribute in a job which rune Familymanager would cause the copy job generated by Familymanager to fail syntax.

The Jobsummarytitle used by the generated jobs will now be the JobSummaryTitle of FamilyManager's parent with the job name appended.

Jul 2007 FMgr:520.11 New parameter:Prefix

Tapes created by Familymanager currently have prefixes of FLEXCOPY or the name of the family being dumped. A new parameter (PREFIX) is now allowed which

will override the default. The specified prefix must be 10 or fewer characters and be valid for tape names. The Prefix parameter is also valid in Mergetape.

Ex

```
("DUMP DEV,PRODUCTION PREFIX DEVPROD")  
or  
("DUMP = PREFIX=HOSTA_ALL")
```

Jul 2007 MTpe 520.03 New Prefix parameter

Tapes created by Mergetape currently have prefixes of FLEXCOPY or the name of the family being dumped. A new parameter (PREFIX) is now allowed which will override the default. The specified prefix must be 10 or fewer characters and be valid for tape names. The Prefix parameter is also valid in Familymanager.

Ex.

```
("MERGE DEV PREFIX REDDEV")
```

Apr 2007 FMgr:520.07 New Option:HIDEDONTBACKUP

A new option HIDEDONTBACKUP can be added to the parameter list passed to FamilManager. If this option is present then files not backed up do to a DONTBACKUP rule will not be listed in user reports.

Typically the same files are reported as 'not backed up' in every FamilyManager run and can make it difficult to recognise more significant actions in the reports.

Summaries will still report the number of files 'not backed up'.

Mar 2007:Flex 520.03 Report Rules Opal version

Report the current Opal version and The Opal version used to compile the Rule in a RULE OP command.

FamilyManager reports can now be directed to email. For sites without Supervisor,the Metalogic Mail Library can be configured via Marc if the Mail directive is set up.

The Marc command DIRECTIVE MAIL = *METALOGIC/MAILLIB/DIRECTIVE sets up the Directive.

Once set up MAIL becomes a valid Marc command.

```
Ex. MAIL STATUS  
MAIL HELP
```

Two new Rules Options have been added:

```
MANAGEREMAIL - Sends User reports in the form of HTML email
```

messages to the Rules Manager. If the Rules Manager usercode has an email entry in the UserData file then that address is used otherwise the Usercode of the rules manager is used as the address. If set as a family option then the FamilManager summary report is sent as email instead of being printed.

USEREMAIL - Sends User reports as an HTML email to the User. If the user has an email entry in the Userdata file then that address will be used otherwise the Usercode is used as the address.

A major revision has been made to the RULES TEST command.

The RULES TEST command allows the user to simulate the effect that FLEX FAMILYMANAGER will have when the user's rules are applied to his files. A report is sent to the terminal describing where the results of the test can be found. Valid Options: RULES TEST RULES TEST USER <usercode> RULE TEST FAM <familyname> Use of the USER construct Or FAM construct is restricted to the <Rules Manager>.

The ordinary user needs only enter RULES TEST to run a simulation for his files.

After a RULES TEST command, the user must use the <RULES CONFIRM> command to actually activate any User specified rules.

RULES TEST FAM <familyname> reports in different ways depending on the reporting option set for the Family. See RULE OP ? FAM <familyname> If MANAGEREMAIL or USEREMAIL is set then all user reports and the summary report is emailed to the Rules Manager. Otherwies all user reports are written to disk under the rules manager usercode on family being tested. The summary is printed.

Dec 2006 FMgr:510.11 More consistent naming for Jobs

If running with the DEBUG option, appending to tapes or after detection of WFL syntax errors, FAMILYMANAGER will lock the generated WFL COPY jobs on the dumped family. The names of these files were inconsistent and did not reflect the type of backup performed nor the medium used. Also, where multiple FAMILYMANAGER jobs were running in series, it was possible for CDIMAGE and CD dumps to occasionally fail because of job file overwrite problems.

The COPY jobs now have the following naming convention using backup command type, the output media, the name of the disk family being handled and creation date:

```
JOB/<backup type>_<output medium>/<family>/<date>/<reel>
```

Where a multi-family backup is being run, <Family> will be replaced by the text "FLEXCOPY" and FAMILYMANAGER's mix number.

For example:

```
JOB/BACKUP_FARM/DEV/2006336/0  
JOB/DUMP_CDIMAGE/WORK/2006335/2  
JOB/BACKUP_TAPE/FLEXCOPY12345/2006330/1
```

Aug 2006 MTpe:510.16 CATALOG PURGE of alt CD image

MERGETAPE will now automatically perform a CATALOG PURGE for CD image files held on the alternate CDIMAGE family that have been released during a MERGE FARM run. Previously, only the file's catalog entries on the primary CDIMAGE family were purged.

Further, the locking of FLEX volumes by the automatic RELEASE NONE run after FAMILYMANAGER completes will now not be logged in MERGETAPE's dialog print file.

Jun 2006 MTpe:510.14 Automatic LOCK of FLEX volumes

In FLEX cataloging environments, MERGETAPE will now unconditionally assign all volumes created by MERGETAPE and FAMILYMANAGER as LOCKED in the TRIM Tape Library system, if this is available. A tape that is LOCKED may only be released by MERGETAPE on the SAME HOST that originally created the tape. This can be seen by viewing the HOSTNAME field in a TP <serial> response. If TRIM is not installed, the above does not apply.

When a tape is LOCKED in a FLEX cataloging environment, it cannot be released by MERGETAPE unless all the following are TRUE (note that the local backup ref count must already be zero for MERGETAPE selection):

- MERGETAPE is running on the host that originally created the tape.
- If MERGETAPE is running from a TRIM slave and that tape volume exists in the Master's Volume Library, the backup references count is checked on the Master and must be ZERO.

Additional error messages may be generated by MERGETAPE subject to the conditions above:

```
Vol:123456: Tape has non-zero MASTER backup refs  
Vol:123456: Volume is LOCKED and not local host
```

Dec 2005 FMgr:510.06 Append continuation tapes

The restriction in WFL on the total length of a Copy statement can force Familymanager to split a dump into multiple tapes even though the physical capacity of the tape has not been reached. If TapeLength is set to 0 (Infinite) and the specified density is capable of appending then Familymanager will make the subsequent copies append to the original tape. This new option is MCP dependent. It is not available on MCPs < 4.8 and on 4.8 it must be 48.nnn.175 or later. On MCP 4.9 it must be 49.nnn.49 or later. MCPs 5.0 and above are fine.

A new option APPEND has been added. The APPEND option is only valid for Tape destinations. If set then Familymanager will search for the most appropriate volumes to be used for this. If the duplicated option is set then the most recent backup with the same volume prefix will be used. If duplicated is not set then the second most recent matching volume will be used.

```
Ex. RUN *METALOGIC/FLEX/FAMILYMANAGER("DUMP DEV APPEND")
```

When using Append either by specification, or if automatic due to a job split, then LIBMAINTDIR files become very important for recovery. Although it is possible to recreate LIBMAINTDIR files from a tape, it can take a very long time. Libmaintdir files should therefore be backed up, as should a copy of the current catalog.

Nov 2005 FMgr 510.05 Prevent syntax errors

There is an MCP imposed limit on the size of a WFL Job. For Dump or Backup runs where a large number (> 200000) of files are to be copied it was possible to exceed this limit. If a duplicated copy was requested, with the singledrive option set then two copies were done in the one job which reduced the limit to nearer 100000 files.

Familymanager now monitors files to be copied against the Job limit and will split the generated job if the limit is about to be exceeded.

If a generated job fails with syntax errors a waiting entry is now created. The name of the waiting entry is METALOGIC/FAMILYMANAGER/FAILURE with an RSVP indicating the name of the Job which failed. Once the failure has been noted an AX can be entered of the waiting entry.

Sep 2005 MTpe:510.10 New IGNORE option

A new option, IGNORE, allows a Disk Farm merge to remove merged or zero-ref disk farms from the Volume Library and TRIM Tape Library even if the FARM removal has previously failed, usually because the farm has been manually deleted. Using IGNORE will allow such volumes to be discarded and the volume serial numbers re-used.

```
RUN *METALOGIC/FLEX/MERGETAPE("DEV FARM(META1) AUTO IGNORE")
```

This change also ensures that a MERGETAPE run only releases eligible entries from the appropriate volume subset (i.e. CD, CDIMAGE, TAPE or FARM) dictated by

the MERGETAPE parameter. Previously, expired, non-released volumes from other subsets would have been released unconditionally causing these volumes to remain indefinitely in the Volume Library and ignored by future MERGETAPE runs.

Mar 2005 MTpe:510.06 Restrict COPYREPORT option

Setting the COPYREPORT option in a Disk Farm Or CDImage merge would cause the output Job to fail with syntax errors.

The COPYREPORT option is now ignored for Disk Farm or CDImage merges.

Mar 2005 MTpe:510.04 New MB LIMIT ©REPORT

The MB option now accepts values > 255.

A new boolean option COPYREPORT has been added. This option causes all copies to include the &REPORT option which creates a printed report of Copies, instead of logging them.

If the output job has to be split due to MCP limits, the continuation Job(s) will append to the existing tapes if possible.

Mar 2005 FMgr:510.01 New parameter:SERIALNO OUT

A new parameter allows FamilyManager to append to existing tapes using the Library/Maintenance option "LIBMTAPPEND=TOEND".

The syntax is:

```
--- SERIALNO --- OUT --- <serialno> ----->
                                   |
                                   +--- , --- <serialno> ----+
```

The second serial number is needed if appending to 2 tapes on a DUPLICATED run. FamilyManager will flag an error if it cannot find the LIBMAINTDIR for a specified serial number.

Examples:

```
SERIALNO OUT 33
SERIALNO OUT A00044 , BKUPMT
```

Using SERIALNO OUT is incompatible with RETRIES options other than NONE. Setting RETRIES to another value will flag an error.

In line with MergeTape, FamilyManager now uses only MegaBytes for tape lengths.

Mar 2005 MTpe 510.03 Replace feet by megabytes

Mergetape now keeps all size info in MegaBytes (MB) instead of feet. The first time

an older Statistics file is updated, the average feet will be converted to average MB.

MERGE = will now keep updated average MB values for the tapes it creates or modifies.

Mar 2005 MTpe:510.02 Boolean params & Density

Boolean parameters are now handled more consistently. A boolean parameter appearing in its own is considered to be set to true. All boolean parameters can be followed by =TRUE or =FALSE.

A new parameter of DENSITY has been added. This is used to specify the density mnemonic of the output tapes to be used, overriding the default values.

Ex.

```
DENSITY=FMTDDS2 .
```

Feb 2005 MTpe:510.01 MERGE =

Using '=' instead of the family name in a MERGE request, enables a new form of merging. Unlike previous MERGEs, this form merges all the files from all families on the input tapes, including files with conflicting names or generations. This also means that the tapes mounted will nearly always be released in the same run. This form of MERGE no longer asks for or needs a WORKPACK, copying directly from the input to the output tapes. However, as the software exists at the moment, LIBRARY/MAINTENANCE will ask for each input tape to be mounted twice, once briefly to get the directory, and once to copy the files off. This extra mount should be painless for sites with tape robots. By doing the copy from tape to tape, this MERGE is practical only for installations with at least 3 tape drives. A number of 2 drives is the lowest accepted, but should be used as a last resort because each input tape must then be mounted 4 times.

MERGE = is valid with all forms of MERGE involving tapes or CDs, but not CDIMAGE or DISKFARM media. It is also disallowed for the OFFLINE option as the OFFLINE list is still family dependent.

Dec 2004 MTpe:500.20 REPORT =

REPORT = reports on the status of all the families which have backed up files.

Dec 2004 MTpe:500.19 New 'SERIALNO OUT'

OUT is a new keyword after the SERIALNO clause which can be used in a merge to specify the serial numbers of the 2 output tapes. If the tapes are good backup tapes in the Volume Library, the copy will have a LIBMTAPPEND=TOEND clause.

The new syntax is:

```
SERIALNO OUT <serial> , <serial>
```

Nov 2004 Flex:500.08 OPTION FOR LIBMAINTDIR

The DEFAULT command screens now allow for LIBMAINTDIR files to be made when MergeTape or FamilyManager make tapes. There are 3 settings:

- 0 (Auto) - Make LIBMAINTDIR files if the density can have a drive which is LOCATECAPABLE. This is the default.
- 1 - Always make a LIBMAINTDIR.
- 2 - Never make a LIBMAINTDIR.

If a tape has a LIBMAINTDIR, and is mounted on a LOCATECAPABLE drive, COPY can use direct positioning to speed up file access. A LIBMAINTDIR file is also needed to enable a subsequent LIBMAINTAPPEND=TOEND copy.

Nov 2004 FMgr:500.10 Check CopyWrite Library Status

FAMILYMANAGER will now verify that the COPYWRITE library is available and active, before commencing all backup operations to CD, CD image or Disk Farms.

The following messages may be seen:

```
'Missing GETCOPYWRITESTATUS entrypoint in MAGUS'  
'COPYWRITE library is inactive or not installed'  
'COPYWRITE DSS functionality is DISABLED'
```

The first message indicates an unlikely incompatibility problem with an older version of MAGUS. If COPYWRITE is already installed, then it may be necessary to re-install using the COPYWRITE utility.

e.g. from CANDE:

```
U *METALOGIC/COPYWRITE REINSTALL
```

Nov 2004 MTpe:500.17 COPYWRITE CHECK

MERGETAPE will now verify that the COPYWRITE library is available and active, before commencing all merge operations to CD, CD image or Disk Farms. The following messages may be seen:

```
'Missing GETCOPYWRITESTATUS entrypoint in MAGUS'  
'COPYWRITE library is inactive or not installed'  
'COPYWRITE DSS functionality is DISABLED'
```

The first message indicates an unlikely incompatibility problem with an older version of MAGUS. If COPYWRITE is already installed, then it may be necessary to re-install using the COPYWRITE utility e.g. from CANDE:

```
U *METALOGIC/COPYWRITE REINSTALL
```

Also, MERGETAPE will now verify and, if necessary, update the creation date of CDIMAGE or FARM volumes created during the merging process. Previously, the creation dates of such volumes were zero.

Oct 2004 MTpe:500.16 LIBMAINTDIR FOR TAPE LOADS

If 2 versions of a file were on a tape that MERGETAPE used for loading it, the wrong version could have been loaded. This could only happen on a non-FLEX tape. This problem has now been avoided where the tape has a LIBMAINTDIR. This change also allows LIBMAINTAPPEND=TOEND tapes to be used, although MERGETAPE will not as yet create them.

Sep 2004 MTpe:500.15 FARMS v SCRATCH POOL

MERGETAPE will now ignore any SCRATCHPOOL assignment when creating disk farm volumes. Normally, COPYWRITE checks for specific SCRATCHPOOL values to provide specialised actions so MERGETAPE now disables this by default.

Sep 2004 MTpe:500.14 ALTERNATE IMAGE SPEC

Similar to the implementation described in FAMILYMANAGER DNote 500.07, it is now possible to create CD images on a family other than that assigned to the FLEX_CWIMAGE configuration variable.

As with FAMILYMANAGER, this is controlled using the following syntax change to the CDIMAGE modifier:

```
----- CDIMAGE -----+-----+-----|
                        |               |
                        +-- = <Alt.family> --+
```

For example:

```
RUN *METALOGIC/FLEX/MERGETAPE ("DEV CDIMAGE=ALTIMAGE AUTO")
```

Jun 2004 MTpe:500.09 Don't re-release Disk Farm files

This change supports that discussed in COPYWRITE DNote 500.07 which protects the RELEASE FARM of files from a Disk Farm that have already been released by an earlier run of MERGETAPE.

The RELEASE command has been extended to permit the keys words CD, CDIMAGE, FARM or TAPE. This permits MERGETAPE to release volumes of the

appropriate type that have zero references and are marked as RELEASED or WASBACKUP. The RELEASE FARM variant that was previously used to housekeep Disk Farm files will now only operate if a threshold value is given.

```
----- RELEASE -----+--- CD -----+-----+-----+-----+-----+-----+
                        +--- CDIMAGE ---+                               |
                        +--- TAPE -----+                               |
                        +--- FARM -----+-----+-----+-----+-----+
                                   +--- > <integer>-- +
```

Jun 2004 MTpe:500.07 No Compare for CopyWrite

The COMPARE option will now be ignored with any Disk Farm merging operation as it is not supported by COPYWRITE.

May 2004 MTpe:500.03 Require Rules Manager user code

All MERGETAPE runs must now be run under the Rules Manager usercode; previously, MERGETAPE could be run under any PU usercode but this changes aligns MERGETAPE to use the same security restrictions as the FAMILYMANAGER product. It is not necessary to run MERGETAPE with the Rules ACCESSCODE

May 2004 FMgr:500.07 Alternate Family for CDIMAGE

For DUMP and BACKUP activities to CD disk images (using the CDIMAGE modifier, FAMILYMANAGER will use the family specified by the Magus config variable FLEX_CWIMAGE as image file destination.

It is now possible to override this setting using the following modified syntax:

```
--- DUMP --- <Family> --- CDIMAGE ---+-----+-----+-----+-----+-----+
                                         |                               |
                                         +--- = <Alt.Family> ---+
```

When an alternate CD image is specified, the CD images are generated as normal on the default image family and then automatically COPY&BACKUP by CopyWrite to the alternate family. Copywrite removes the image from the original family but leaving a non-resident file entry pointing to the alternate family. This is referred to in Metalogic software as a 'HARD LINK'. You may specify as many as alternate families as required (as long as the family is volumed).

In this example, the default image family is DISK and an alternate image family is called ALTIMAGE. After creation of the image, a PD of the image on each family appears as:

```
PD *DEV/CI0017F2004142A

FILE *DEV/CI0017F2004142A ON DISK (PROMBURNERDATA)
NOT RESIDENT
FAMILY SERIAL NUMBER: 000505
ENTRY 1 (PROMBURNERDATA):
CYCLE: 1 VERSION: 0
SECURITY: PRIVATE
TIMESTAMP: Friday, May 21, 2004 (2004142) AT 11:52:15
LASTACCESS: Friday, May 21, 2004 (2004142) AT 11:52:02
BACKUP MEDIA IS: PK
SERIAL: 000600
BACKUP MEDIA IS: PK
SERIAL: 000600
```

On ALTIMAGE (pack serial number 600):

```
PD *DEV/CI0017F2004142A ON ALTIMAGE

FILE *DEV/CI0017F2004142A ON ALTIMAGE (PROMBURNERDATA)
CREATION DATE= Friday, May 21, 2004 AT 11:52:01 BST
LASTACCESS DATE= Friday, May 21, 2004 AT 11:52:01 BST
ALTER DATE= Friday, May 21, 2004 AT 11:52:01 BST
ATTMODIFY DATE= Friday, May 21, 2004 AT 11:52:16 BST
BACKUP DATE= Friday, May 21, 2004 AT 11:52:01 BST
COPYDEST DATE= Friday, May 21, 2004 AT 11:52:01 BST
COPYSOURCE DATE= Friday, May 21, 2004 AT 11:52:01 BST
EXECUTE DATE= Friday, May 21, 2004 AT 11:52:01 BST
READ DATE= Friday, May 21, 2004 AT 11:52:01 BST
TOTAL SECTORS: 88,896 (5,556 PER AREA)
SECURITY = OWNER *:RWX, GROUP <none>:NO, OTHER:NO
(PRIVATE - USAGE: READ/WRITE) LOCKEDFILE
FAMILY SERIAL NUMBER: 000600
ENTRY 1 IS RESIDENT, NOT CATALOGED (PROMBURNERDATA):
CYCLE: 1 VERSION: 0
SECURITY: PRIVATE
TIMESTAMP: Friday, May 21, 2004 (2004142) AT 11:52:15
NO ARCHIVE BACKUP RECORD FOR FILE
```

May 2004 MTpe:500.02 CDIMAGE Merging

MERGETAPE now supports the merging of CopyWrite Disk Farms. These changes have been released in conjunction with enhancements to the CopyWrite package (please see COPYWRITE Software Changes [500.02](#) and [500.03](#)).

The MERGE command has been enhanced to enable Disk Farm merging:

```
-----+-----+--- <Family> ----- FARM ----- ( <Farm Host> ) -----  
      |           |  
      +-- MERGE  --+
```

The use of <Farm Host> is mandatory since, at this time, it is not possible to merge farms which may be on multiple PC hosts.

MERGETAPE will now automatically correct average MB values for both CD image and Disk Farms if they are found to be incorrect. MERGETAPE requires these values to determine the weighting of eligible volumes to be used in the merging process. Previously, Disk Farms values would always be 0 or 1 which affected volume selection. For Disk Farms, MERGETAPE will automatically search for the volume's LIBMAINTDIR file and calculate the average disk space by extracting the size of each file. If the Disk farm LIBMAINTDIR file is missing, MERGETAPE will determine the hostname of the Disk farm from the TRIM Tape Library subsystem and attempt to re-create the LIBMAINTDIR using a CopyWrite COPY.

MERGETAPE expects to find LIBMAINTDIR files under the RULES Manager usercode on the DL LIBMAINTDIR family.

During the merging process, if MERGETAPE determines that a Disk Farm is eligible for deletion (i.e. the Disk Farm volume has 0 references), an ADD *= FROM NULL request is initiated requesting CopyWrite to purge the farm at the specified host. When the delete is successfully completed, MERGETAPE will automatically delete any associated Tape Library entry, remove the LIBMAINTDIR and delete the Volume Library entry.

All the above activities are logged in MERGETAPE's dialog print file. This is an example of Disk Farm volume DF0020 being deleted from PC host META1:

```
12:26 ..Volume Library and TRIM entries will be removed  
12:26 ..Volume DEV_A [DF0020] will be released  
12:26 ..Disk Farm DF0020 will be deleted from META1  
12:26 ....Removing old LIBMAINTDIR for DF0020  
12:26 ....(FLEX)LIBMAINTDIR/DEV_A/20040512/DF0020 ON DISK  
        successfully removed  
12:26 ....Deleting DF0020 from Volume Library
```

The WFL deletion of an extinct Disk Farm would typically look like:

```
#BOT 14081 (IPP)WFLCODE
#BOT 14082 (IPP)FILE/TRANSFER
#BOT 14083 (IPP)FILE/TRANSFER/SERVICES
#14083 CopyWrite:Metalogic CopyWrite Version 50.500.03 was compiled at
    16:41:8 on the 18th May 12004 (MCP 49.189.8737)
#13802\14085 BOT (IPP)SESSION/"Mix# 14083"/"10.0.0.2"/
    META1/IPP/COPYWRITE
#14083 CopyWrite:Release ALL 4 FILES in SPARE_A
#14083 CopyWrite:File Release Completed. Volume Release Check Pending
#14085 CopyWrite:[META1] Disk Farm C:\ASeriesBackup\DELL8500MCP\
    SPARE_A[DF0017] Removed
#14085 CopyWrite:[META1] Deleting 4 Files in C:\ASeriesWaste\
    SPARE_A[DF0017]_21_May_2004_155812
#14085 CopyWrite:[META1] Deleted C:\ASeriesWaste\
    SPARE_A[DF0017]_21_May_2004_155812\FILE000.dir
#14085 CopyWrite:[META1] C:\ASeriesWaste\
    SPARE_A[DF0017]_21_May_2004_155812 Deleted
#13802\14085 EOT (IPP) (IPP)SESSION/Mix# 14083/10.0.0.2/META1/
    IPP/COPYWRITE
#EOT 14083 (IPP) (IPP)FILE/TRANSFER/SERVICES
#EOT 14082 (IPP) (IPP)FILE/TRANSFER
#EOT 14081 (IPP) (IPP)WFLCODE
```

If Disk Farms volumes hold very large files which drop out of the catalog on a regular basis, it is possible that disk space shortages may become an issue on PC hosts. This can occur because such files will never be released until the volume is merged or drops to zero reference by natural means.

To help minimize this problem, a new modifier to the RELEASE command is now available:

```
--- RELEASE ----- FARMS -----+-----+-----+
                                   |           |
                                   +-- <KB Threshold> --+
```

The RELEASE FARM command will analyse all existing Disk Farms in the Volume Library searching for large files (or older generations of those files) that have dropped out of the catalog. By default, MERGETAPE will use a threshold file size of 5 Megabytes (5000 KB) to select files but this may be raised or lowered by using <KB Threshold> to specify a lower limit in KiloBytes.

For example:

```
RELEASE FARM 8000
```

The above command would look for expired files that have now dropped out of the catalog and are larger than 8 Megabytes. MERGETAPE uses each Disk Farm's LIBMAINTDIR file to select eligible files so this must file be resident or the volume will be excluded. Further, the family associated with each eligible file must be online and available or the release will be skipped.

When CopyWrite receives a request to release a file from a Disk Farm, the file is not

removed but replaced by a 'RIP' capsule which is essentially a null file, occupying minimal disk space. This means that a reference to the file remains in the Disk Farm but the disk space has been returned.

At this time, neither MERGETAPE nor CopyWrite 'know' that a released file has been processed before so multiple RELEASE FARM runs may re-release the same file. This problem will be fixed in a later version of Copywrite by marking the file as released in the associated LIBMAINTDIR. A RELEASE FARM command is much faster than a Disk Farm merge and can easily return significant disk space.

In practice, a combination of merging and RELEASE is envisaged as the best way to housekeep Disk Farms. As with CD images, MERGETAPE can be run silently, without operator intervention, to merge Disk Farms by PC host, at scheduled times.

For example, to merge family DEV with farms on PC host META1:

```
RUN *METALOGIC/FLEX/MERGETAPE ("DEV FARM (META1) AUTO")
```

This run uses automatic settings that control Farm selection, by default, using less than 25 MBytes to a maximum of 10 volumes. These values can be readily overridden by run-time modifiers.

Nov 2003 FMgr:490.04 Better validation on Initialisation

During FAMILYMANAGER's initialisation phase, the availability of the FLEX RULES file and the setup of the RULES manager usercode and accesscode will now be verified. Previously, FAMILYMANAGER would not give any warnings if these essentials features were not established.

If the RULES file has not been set up or is unavailable:

```
'RULES file is not valid or does not exist'
```

If neither the RULES Manager usercode nor its alternate usercode have been setup:

```
'RULES Manager USERCODE is NOT assigned'
```

If neither the RULES Manager accesscode nor its alternate accesscode have been setup:

```
'RULES Manager ACCESSCODE is NOT assigned'
```

In any of the above circumstances, FAMILYMANAGER will emit the error and then terminate gracefully.

Introduction

File Management is an expensive and time consuming business. The FLEX package is a tool designed to reduce that cost. File Management on Unisys MCP systems has been a concern of METALOGIC since 1976, when the first version of the FAMILYMANAGER program was created. Later, in 1978, we helped pioneer the concept of Virtual Disk by providing general utilities to assist its implementation.

In 1983, when METALOGIC set out to enhance its long-established Virtual Disk System, the following major design goals were identified to implement a complete backup strategy:

- Significantly increase the efficiency of pack usage at most installations.
- Allow improved balancing of pack channel capacity.
- Allow site management to share disk space between their users.
- Monitor files including detection of dangerous or anti-social file usage.
- Provide a powerful yet simple interface to the Unisys File Directory subsystem.
- Implement a safe and simple to use method to ensure recovery from lost files due to pack crashes, software errors, site catastrophes etc.
- Use standard Unisys software where possible and, in particular, never directly read user files. All tapes should be created and read by LIBRARY/ MAINTENANCE.
- No patches to Unisys System Software and should not require the purchase of any other packages such as DMSII.
- Allow for easy and automatic re-loading of non-resident files, making best use of the tape drives.
- Consume minimal resource, in processor time, memory space, and pack usage. These resources should be negligible during prime time.
- Allow easy management of a library of tapes.
- Provide meaningful but concise reports, both to site management, and to individual users, as well as relevant reports on inefficient or dangerous file usage.

It was quickly realised that to achieve these goals a new level of software sophistication would be needed. It was found in the recent Artificial Intelligence developments of Expert Knowledge Systems. FLEX (File Location EXpert) which is the product of many man-years of development, achieves or surpasses these original aims.

Overview

The Site Manager, and optionally the end user, runs a utility called FLEX Inquiry which handles all the conversational transactions. In the Site Management environment, the primary task of FLEX primary task is to allow file-selection rules to be defined, to report errors, run simulations on proposed new rules and control various run-time options.

FLEX Inquiry is also a major File Management tool in its own right. It provides greatly enhanced commands, which effectively replace the following system programs

SYSTEM/FILEDATA
SYSTEM/FILECOPY
SYSTEM/LISTVOLUMELIB

FLEX may be run either in batch mode or interactive mode on a variety of screens and terminal emulators. However, the sophisticated user has not been ignored, as many options and advanced features are also provided. A design objective of FLEX was to make all other generalised file management utilities obsolete.

A second program, FAMILYMANAGER, is responsible for applying the files selection criteria or 'Rules'. It also reports its actions to each user, if required, builds WFL jobs to perform the actions specified by the rules and applies segment limits if enabled, in short it manages the disk and disk pack families of the system.

FAMILYMANAGER can generate reports in various ways. A summary report is always produced for the Site Manager, it is intended to be an “at a glance” summary page for each user of the system. It will also report its actions individually to each user if required, to either a disk file a printer file or an emailed report.

The third program, MERGETAPE, manages the backup volumes created by FAMILYMANAGER. MERGETAPE is primarily a tool for the tape librarian. It is aware when backup volumes no longer have active files stored on them and instructs the tape librarian to scratch them. It will also consolidate backup volumes, by identifying the active files on each and merging them onto new volumes. This function is necessary to keep the number of backup volumes down to some site defined minimum working set.

Each of the programs relies heavily on two libraries in which are centralised all the utility code, Rule Definition and Interpretation functions, and the interfaces to the MCP. These two libraries are SL'ed with the function names FLEXLIB and MAGUS.

No patches to any Unisys software are required.

Any package to manage files needs some “database” in which the state of files is kept. FLEX is no exception to this rule. FLEX uses a database provided free to every Unisys A-Series customer by Unisys itself — the MCP Cataloging feature. Since Unisys supports the MCP, Metalogic is not required to, and the FLEX user benefits. The CATALOGING option may be unfamiliar to many Unisys sites as there was little available documentation for many years.

FLEX rules

Both FAMILYMANAGER and FLEX Inquiry can use 'rules' written in a Metalogic scripting language called OPAL. The OPAL language allows very powerful and concise filter or report scripts to be written controlling the selection of files, in the case of FAMILYMANAGER, for copying to backup media or housekeeping functions.

FLEX rules are discussed in more detail in [FLEX Rules](#); information on the OPAL language can be found in the **Metalogic OPAL Language Reference manual**.

About this Manual

This manual is addressed to site management. It is intended for reference, and should be read after the FLEX Inquiry manual. In particular, all information about installing FLEX is in the FLEX Inquiry manual. The OPAL Language Reference Manual should also be referred to as it relates closely to the section on RULES. The structure of this manual is as follows:

Section	Brief Description
Cataloging Overview	An introduction to MCP cataloging
Getting Started	Getting up and running with Flex
Backup Strategy	Plan how to backup your different volumes
Flex Rules	Define and enforce your file management rules
FamilyManager	Instructions on the FamilyManager program
MergeTape	Instructions on the MergeTape program
Configuring Flex	Fine tuning
Flex Recovery Features	Using Flex to recover lost files

Recent enhancements

Over the years, FLEX has evolved into a powerful tool for adopting a comprehensive backup strategy. In particular, FLEX can now use media such as CD or DVD, local disk storage (as CD images) or direct PC disk farms created over a TCPIP network. In particular, cheap, fast PC disk storage offers a much more viable storage medium because the hardware is inexpensive compared to disk or tape subsystems currently available on Clearpath MCP systems.

These new features have only been possible due to the implementation of the Metalogic COPYWRITE software, which offers considerable functionality beyond the FLEX environment. COPYWRITE intercepts LIBRARY/MAINTENANCE requests by detecting usage of certain attributes in the COPY statement allowing different kinds

of backup media to be easily created not just from FLEX but any MCP COPY statement.

The COPYWRITE software is not included in the basic FLEX Site Management package and a separate licence must be purchased from Metalogic. For more details see [CopyWrite](#)

Cataloging Overview

The MCP cataloging system implements automated methods to assist the location of backup copies of both disk files and tape files. Support for cataloging is enabled via system option 23 (CATALOGING), which if subsequently set, requires a system halt-load for activation.

On systems with CATALOGING reset, the MCP will search for individual file names in a system structure called the FAST (File Access Structure). The FAST behaves similar to an index sequential set, returning the physical address of the disk file header in the directory file *SYSTEMDIRECTORY/001 (also known hereon as the FLAT directory).

The FAST structure is held on the DL CATALOG family in a system file called:

***SYSTEM/ACCESS/001**

If the CATALOGING option is set and the system halt-loaded, some changes immediately occur. The FAST is now moved to the following file (known, more familiarly, as the CATALOG) :

***SYSTEM/CATALOG/001**

When a file is made permanent on a cataloging system, the MCP enters an entry for that file into the CATALOG file. By default, when CATALOGING is enabled, 3 generations of the same file are maintained though this is easily controlled with the SYSOPS CATALOGLEVELSET command allowing file retention from 1 to 6 generations.

SYSOPS CATALOGLEVELSET 4

When the FAST is now searched for a requested file, the MCP returns a pointer to an area within the CATALOG file called the CATALOG BLOCK. along with other information, the CATALOG BLOCK holds a pointer, for resident files, to the FLAT directory.

If a permanent file, that has been cataloged, is subsequently removed but no backup copies of the file exist, the catalog entry for the file is deleted. However, if the file has been backed up, the file is then marked in the catalog as NONRESIDENT.

Only one version of a non-resident file can be resident at any time . Cataloging systems can keep track of 2 backup copies of each generation of a disk file as determined by the file attributes CYCLE and VERSION.

Volume Library

The Volume Library is an internal table within the CATALOG file that maintains information about all volumed disks and tapes known on the system. Tapes and disks are considered 'volumed' once they have been added into the Volume Library by the WFL VOLUME command:


```
VOLUME ADD DEV (KIND=PACK, SERIALNO=501)
VOLUME ADD BACKUPTAPE (KIND=TAPE, SERIALNO="A00012")
```

When a disk volume has been added into the Volume Library, the files on that family be the subject of backup copy operations using the COPY&BACKUP Library/Maintenance command.

When a tape volume has been entered into the Volume Library, files from a volumed family can be copied to that tape using COPY&BACKUP, creating entries in the CATALOG file.

Once a file has been copied using COPY&BACKUP, the system PD command will report backup information.

Ex.

```
PD (BOB)TEST ON DEV
FILE (BOB)TEST ON DEV (DATA) & (DIRECTORY)
  CREATION DATE= Tuesday, February 1, 2005 AT 11:47:07 GMT
  LASTACCESS DATE= Tuesday, February 1, 2005 AT 11:47:25 GMT
  ALTER DATE= Tuesday, February 1, 2005 AT 11:47:07 GMT
  ATTMODIFY DATE= Tuesday, February 1, 2005 AT 11:47:07 GMT
  BACKUP DATE= Monday, November 26, 2007 AT 18:14:24 GMT
  COPYDEST DATE= Friday, December 2, 2005 AT 11:34:39 GMT
  COPYSOURCE DATE= Wednesday, June 4, 2008 AT 03:11:03 BST
  EXECUTE DATE= Tuesday, February 1, 2005 AT 11:47:07 GMT
  READ DATE= Tuesday, February 1, 2005 AT 11:47:25 GMT
TOTAL SECTORS: 1,024 (1,024 PER AREA)
SECURITY = OWNER BOB:RWX, GROUP <none>:NO, OTHER:NO
(PRIVATE - USAGE: READ/WRITE)
FAMILY SERIAL NUMBER: 000504
ENTRY 1 IS RESIDENT, CATALOGED (DATA):
  CYCLE: 1 VERSION: 0
  SECURITY: PRIVATE
  TIMESTAMP: Tuesday, February 1, 2005 (2005032) AT 11:47:14
  BACKUP MEDIA IS: MT
  SERIAL: HP0003
  BACKUP MEDIA IS: MT
  SERIAL: HP3004
```

Note the backup information at the end. This file is backed up on two tape volumes HP0003 and HP3004.

The PV command will show details of the Tape:

```
PV MT HP3004  
-----VOLUME LIBRARY ENTRY FOR (MT) [HP3004]-----  
SERIALNO HP3004 , #0001 , TAPE , ONSITE  
FAMILY NAME: FLEXCOPY05346AG  
FAMILY CREATION DATE: Monday, December 12, 2005  
FAMILY EXPIRATION DATE: Wednesday, January 11, 2006  
FAMILY CREATION SITE: 1  
VOLUME SEQUENCE NUMBER OF FIRST VOLUME 1, NUMBER OF VOLUMES 1
```

Getting Started

Utilities overview

The FLEX package is available in two flavours: FLEX I and FLEX III (or Full Flex). For a time, as a marketing venture, a third variant called FLEX II existed but this caused much confusion with customers and was eventually withdrawn.

The FLEX I package consists of the following files and utilities:

METALOGIC/FLEX/LIBRARY

This codefile is a SL-ed library (FLEXLIB) that has a core of entrypoints used by all other FLEX components. FLEXLIB procedures maintain the RULES file, execute OPAL rules and provide sophisticated CATALOG and FLAT handling capabilities. Neither FLEX package will function without FLEXLIB.

OBJECT/FLEX

Also known as FLEX Inquiry or FLEX Utility.

FLEX Inquiry provides an interactive, screen-based view of the disk subsystem allowing the user to search and inspect files in multiple locations and media. Using OPAL file selection and reporting criteria, the user can easily generate his or her own reports displaying selected information only on the files selected.

FLEX Inquiry also has specialist command such as PDT (directory analysis on media such as CD, wrapped containers and tape) and USAGE (allowing a variety of file usage summaries to be generated for a selected family). The utility is comprehensively covered in the **Metalogic FLEX INQUIRY Reference manual**.

Example scripts and WFL jobs

A variety of OPAL scripts and WFL jobs are available in the basic FLEX package providing useful techniques and methods for the use of FLEX Inquiry and writing OPAL scripts.

The FLEX III package includes the above from the FLEX I package plus the following utilities:

METALOGIC/FLEX/FAMILYMANAGER

Hereafter known as FAMILYMANAGER.

The FAMILYMANAGER utility is that component of FLEX which manages the files of all users at an installation, based on OPAL "rules" defined by the site management

and the users themselves. FLEX rules control whether files show be backed up, removed or deleted from the catalog.

Several hard-wired rules are pre-defined (e.g. every file must have 2 good backups except DMSII DBDATA files) but it is the responsibility of the site to set up and maintain its own rule database.

FAMILYMANAGER is easily run from WFL and can be configured, using simple command syntax, to generate incremental backups to tape, CD, CD image or Disk Farm using COPY&BACKUP. However, FAMILYMANAGER can only back up files to media other than tape, only if the COPYWRITE package has been purchased.

METALOGIC/FLEX/MERGETAPE

Hereafter known as MERGETAPE.

The MERGETAPE utility is responsible for the release of backup media that now hold no active files that have backup entries in the catalog pointing to that medium. This can occur by natural causes, over time but, normally, regular runs of the MERGETAPE utility are required to merge volumes with small numbers of cataloged, backed-up files to free up media. This is true of all FLEX media.

MERGETAPE will 'merge' all FLEX backup media, with command support for CD, CDImage and Disk Farms if the COPYWRITE package has been purchased.

METALOGIC/COPYWRITE

Hereafter known as COPYWRITE.

COPYWRITE is NOT part of the main FLEX III package and must be purchased separately. COPYWRITE provides significant extensions to the COPY command allowing the usage of CD, CD Image and Disk Farm as alternative media to tape.

COPYWRITE consists of a number of SL-ed libraries and applications to support network access to PC Disk Farms in particular. In the latter case, a PC package, called NTCOPYWRITE (with many useful PC utilities) , must be installed and correctly configured.

Discussions of the full capabilities of the COPYWRITE package is beyond the scope of this manual. More information concerning COPYWRITE can be obtained from the Metalogic web site:

<http://www.metalogic.eu.com/>

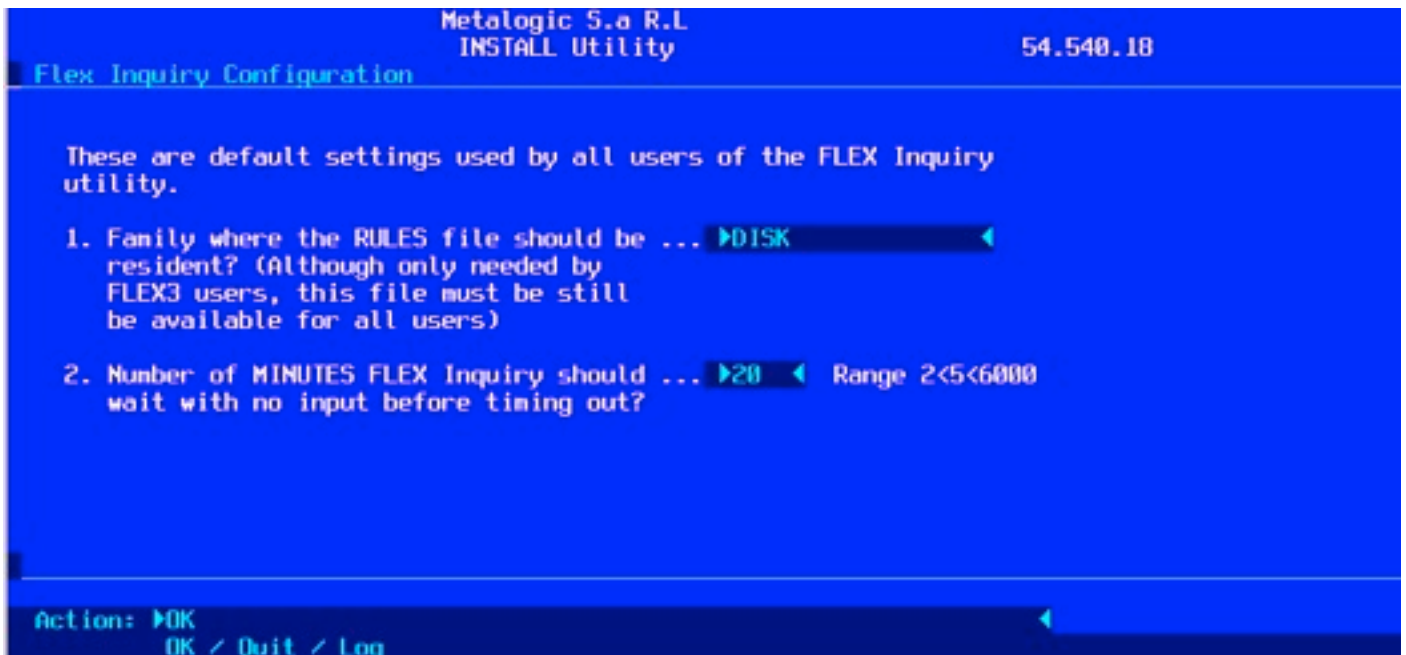
Installation

The **Metalogic Software Installation** reference manual includes the basic instructions for installing all Metalogic software and includes a short section on some of the questions that are asked by the INSTALL utility, specifically about the FLEX

operational environment.

All Metalogic software is installed using the INSTALL utility. The software is provided in Unisys wrapped container format and INSTALL provides a menu-driven interface for selective software installation.

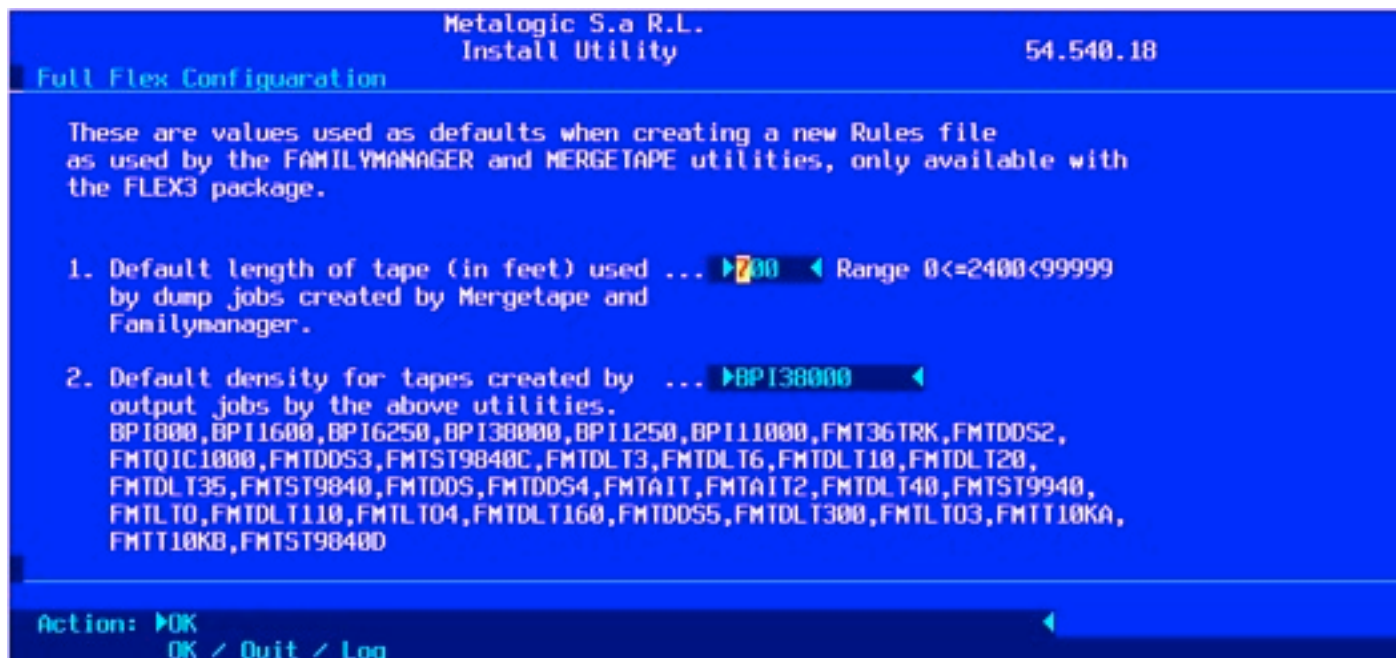
During the first installation of the software, the INSTALL utility will present the specific FLEX screens shown below:



The RULES file is an important part of the full FLEX package: it holds all the user-written rules controlling the housekeeping of files by usercode and family.

The FLEX package cannot run without this file and a family must be designated using the field displayed in 1. The RULES file is typically low-usage and will not cause any significant overhead if used on a critical family.

In question 2, a suitable value for the Flex Inquiry program inactivity timeout value (in minutes) should be given. The default is 5 minutes.



The second screen requires confirmation of two environment variables specific to the Full FLEX environment. For sites using tape media, question 1 requires an estimated 'feet of tape' representing the approximate length of the physical tape.

In earlier times, where tape densities were confined to smaller tape capacities such as BPI6250 and BPI38000, these values were important in the calculations performed to estimate the length of physical tape used by a FLEX COPY&BACKUP run. Now, tapes with DLT densities for use in tape robots, this value is meaningless since the tape capacity is enormous compared to the open-reel ancestors. So, for sites using high-capacity DLT tapes, this value should be set to 0.

FAMILYMANAGER copies to CD, CD Image and Disk Farms do not use this value as they have their own specific defaults.

Question 2 requires the default tape density to be used for all FLEX-generated COPY&BACKUP output media. This density will be inserted into the output job tape specification by FAMILYMANAGER and MERGETAPE. Although a default density must always be assigned, the value is NOT applicable for non-tape backup generation.

Both these settings can be overridden by FAMILYMANAGER run-time commands.

Setting up FLEX and CATALOGING

This should be done from a Cande window logged on as a privileged user.

Setting up Cataloging

Setting the Cataloging option.

If the Cataloging option is not already set on this system then this procedure will involve a longer than usual Halt/Load. Please ensure you do this at suitable 'quiet'

time.

```
Enter U FLEX
and then
INSTALL
```

This will give the screen

```
#Entering Installation Mode
#Installation Mode Functions available are :-
#1.  Initialise Cataloging on this machine.
#2.  Create an empty Rules base.
#3.  Initialise the Rules Base for FAMILYMANAGER.
▶ █ ◀
#Enter a number in the range 1 - 4 (QUIT to exit this level) or 'HELP'
```

Enter 1 and transmit to Install cataloging.

```
##### Warning #####
#There will be a CATALOG REBUILD after the next Halt Load
#This will take at least double the time of a FAMILY REBUILD
#You should then rerun FLEX Utility to continue installing Cataloging
#Most of this installation step requires exclusive use of the system
#This is a time consuming process!
#Be sure you have sufficient time available
#OK to set the system option CATALOGING?
▶ █ ◀
#Enter 'YES','NO' or 'QUIT'
```

Enter YES and transmit

```
#Please H/L now to create your Catalog.
#Rerun FLEX after the H/L to continue installation
Use the FLEX command 'INSTALL CATALOG' to continue
▶ █ ◀
#Enter anything to proceed
```

Enter any character at the home position and transmit.

Flex will quit. Halt/Load the system.

After the Halt/Load you will get a waiting entry Like:

```
W
-----Job--Task-Pri---Elapsed----- 1 WAITING ENTRY -----
      4/   4  85      11:00 Job STARTSYSTEM
      OK TO CREATE NEW CATALOG:PK501  DEV
```

This should be OKed

Volume adding Families

Log on to Cande as a privileged user.

Enter U FLEX
and then
INSTALL

This will give the screen

```
#Entering Installation Mode
#Installation Mode Functions available are :-
#1.  Initialise Cataloging on this machine.
#2.  Create an empty Rules base.
#3.  Initialise the Rules Base for FAMILYMANAGER.
> 1
#Enter a number in the range 1 - 4 (QUIT to exit this level) or 'HELP'
```

Enter 1

```
#OK to VOLUME all packs currently mounted?
> Yes
#Enter 'YES', 'NO' or 'QUIT'
```

Enter Yes and transmit.


```
#Note : There are two CANDE options which apply only on cataloging MCPs
#CATDEFAULT set, means all files created by CANDE will be cataloged
#CATALOGOK set, enables additional syntax in the MAKE and GET commands.
#You are advised to set CATDEFAULT only after all packs are volumed
> █ ◀
#Enter anything to proceed
```

The screen above gives some information about Cande Options related to cataloging. Entering any input takes you back to the main Install screen

Create a Rules File.

The Rules file is used to hold configuration information for the Flex package and details of all rules to be applied.

```
#Entering Installation Mode
#Installation Mode Functions available are :-
#1.  Initialise Cataloging on this machine.
#2.  Create an empty Rules base.
#3.  Initialise the Rules Base for FAMILYMANAGER.
> █ ◀
#Enter a number in the range 1 - 4 (QUIT to exit this level) or 'HELP'
```

Enter 2 and transmit

The 'Rules Manager' is a Specified user with a specified access code who has special privileges relating to Flex. Only the 'Rules Manager' can define System wide Family Management Rules or run the FamilyManager program used to apply these Rules

```
#Do you intend that usercode FLEX be the Rules Manager?
> █ ◀
#Enter 'YES','NO' or 'QUIT' or 'HELP'
```

Answering No gives

```
#Please run FLEX under the desired usercode, or use the USER command.
#Installation Mode Functions available are :-
#1.  Initialise Cataloging on this machine.
#2.  Create an empty Rules base.
#3.  Initialise the Rules Base for FAMILYMANAGER.
> <
#Enter a number in the range 1 - 4 (QUIT to exit this level) or 'HELP'
```

Answering Yes if you are not running with an access code gives

```
#Shall I define an accesscode in the userdatafile?
#It will be accesscode RULES, accesspassword MASTER.
#You may change the access password with CANDE (APASS)
> <
#Enter 'YES','NO' or 'QUIT' or 'HELP'
```

```
#If the USERDATA attribute ACCESSCODENEED is set,
#all jobs and sessions for FLEX must specify an accesscode.
#It is optional if ACCESSCODENEED is reset.
#However, FLEX requires both the usercode and the accesscode to perform Rule Manager functions.
#Should the accesscode be made mandatory?
> <
#Enter 'YES','NO' or 'QUIT'
```

Answering Yes if you are running with an access code gives

```
#Do you intend that usercode FLEX with accesscode RULES be the Rules Manager?
> <
#Enter 'YES','NO' or 'QUIT' or 'HELP'
```

Change Defaults

Action ▶ ☒ ◀
(Modify, Return)

```

US Dates          ▶N◀ Will US dates be used
Compare           ▶Y◀ Will Library Maintenance copy with &COMPARE
Verify           ▶N◀ Will Library Maintenance copy with &VERIFY
Crunch            ▶Y◀ Will FAMILYMANAGER crunch files by default
LibMaintDir       ▶A◀ Make tapes with LIBMAINTDIRs (Y,N,A[uto])
Load queue        ▶NONE◀ WFL Queue for LOAD jobs
  alternate ▶NONE◀ WFL Queue for large LOADs
Load Limit        ▶7  ◀ Limit on number of tapes for small LOADs
)

Tape Density ▶BPI800  ◀ Tape Length ▶0  ◀ In Feet
(BPI800,BPI556,BPI200,BPI1600,BPI6250,BPI38000,BPI1250,BPI11000,
 FMT36TRK,FMTDDS2,FMTQIC1000,FMTDDS3,FMT128TRK,FMTDLT3,FMTDLT6,
 FMTDLT10,FMTDLT20,FMTDLT35,FMTST9840,FMTDDS,FMTDDS4,FMTAIT,FMTAIT2,
 FMTDLT40,FMTST9940,FMTLTO,FMTDLT110,FMTDLT1,FMTDLT160
)

Savefactor        ▶  ◀ for output tapes
Scratchpool       ▶  ◀ for output tapes
Parallel Output   ▶Y◀ Will duplicated tapes be written to two drives
Loop Retries ▶NONE ◀ (None, Error, Full)

```

```

#Simulating command:RULES CREATE
#RULES File created
#FLEX run-time DEFAULTS have been successfully modified
#Installation Mode Functions available are :-
#1.  Initialise Cataloging on this machine.
#2.  Create an empty Rules base.
#3.  Initialise the Rules Base for FAMILYMANAGER.
▶ ☒ ◀
#Enter a number in the range 1 - 4 (QUIT to exit this level) or 'HELP'

```

```

#Do you intend to run FAMILYMANAGER on family DRPACK?
▶ ☒ ◀
#Enter 'YES','NO' or 'QUIT' or 'HELP'

```

Backup Strategy

Choosing a backup strategy

In practice, choosing a backup strategy depends on various criteria; this chapter deals with the various options in devising an appropriate strategy and choosing the media to use.

In FLEX environments without COPYWRITE, only tape is available as a backup medium. With COPYWRITE, all other options are available allowing a choice of backup media such as CD, local disk or remote PC disk. In summary, the addition of COPYWRITE in the FLEX environment offers considerable diversity in the media available for use as backup.

Multi or single family pack dumps

The FAMILYMANAGER utility can dump from a single family or multiple families in one run. To dump from multiple families the destination must be Tape. Since the main reason for dumping multiple families together is to reduce the fragmentation problem when using the current very high capacity tapes, this is not usually a significant restriction.

Backup Media

The FAMILYMANAGER utility can dump to various types of backup media.

Tapes

Tapes have been consistently the most frequently used backup medium for many years now. The dramatic increases in tape capacity with DLT tapes provide much greater

Disk farms

Are a collection of CDImages maintained on a nominated PC.

This requires Metalogic Copywrite and the third part product Copywrite NT

CD

Requires a CD/DVD Writer.

CD Images

Requires Metalogic Copywrite

Rules are created by the Rules Manager for all file entries on a family or for a particular usercode directory on a family. If permitted by the Rules Manager, a user may create rules for his own files. All rules are defined to the FLEX package with the help of the FLEX Inquiry. Once accepted, a new rule is stored together with all other rules in a file known as the Rules Base. This file is maintained by and accessible only through special interfaces in the FLEX Library. Creation and maintenance of the Rules Base is done with special commands in the FLEX Inquiry. However, only FAMILYMANAGER actually applies the rules.

All file entries of one usercode on a family are always processed together by FAMILYMANAGER. This means the smallest domain of application of a rule is file entries of one usercode directory on one family. It is important to understand that rules are applied to all file entries of a file, not just to the resident generation. When starting to process a usercode directory FAMILYMANAGER assembles all rules which apply to that usercode. It then in turn examines each file entry for the usercode. Every rule is applied to each file entry in turn. The order in which generations of a file are processed, or in which files within the usercode directory are processed is undefined.

There may be many rules which perform the same action on a file entry. For example, there may be a number of reasons to remove a file. In general it is better to keep a rule simple and make many rules, rather than to attempt to gather all the criteria to, for example, remove a file, into one complicated expression.

When a new rule is to be specified to the FLEX package, there are three pieces of information that must be supplied: the action to be taken, the circumstances upon which the action should be taken and a message which identifies the rule.

The reason for the message parts existence is as follows. Typically there are many rules and without some label it would be difficult to identify which rule was applied to a file entry. This labelling is the function of the rule message.

This section first describes the three components of a rule in some detail and then describes how different rules interact. A rule is a program written in the OPAL language and, as with any language, it is possible to create programmes that are invalid. The last section discusses how FAMILYMANAGER behaves if a rule program fails.

All other actions taken are determined by rules defined by the site management for the whole family, rules defined for one specific user by the site management or by a usercode nominated as the Group Leader by site management, and optionally, if permitted by site management, rules defined by the user for his own files.

FAMILYMANAGER operates correctly on cataloging systems with SYSOPS LONGFILENAMES set.

RULES Base

Rules are defined with the aid of the FLEX Inquiry and are stored in an encoded form in a file known as the 'RULES Base' or 'RULES File'. The internal structure of this file is not visible to users.

Modifications to the Rules Base are strictly protected. It is intended that overall control be vested in one person known as the RULES MANAGER.

RULES Manager

To ensure the RULES protection, generally only one usercode is permitted to define or change rules which affect other users. This usercode must be privileged and, as an extra level of security, must have an accesscode with its associated password. Both the usercode and the accesscode are known to the FLEX package. FAMILYMANAGER must run under this usercode and accesscode.

Syntax of Rules

```
- <rule action> - <rule message> -: <boolean expression>|
```

<Rule Action>:

	DONTBACKUP	
	DESTROY	
	REMOVE	
	RELEASE	
	FORGET	
	OFFLINE	
	LOAD	
	RIP	

A Rule action specifies what will be done to a file entry if a the rule is found to apply. All rules actions have their counterparts in commands of the FLEX Inquiry. Some Rule actions have no meaning for file entries which are not resident. Rules of this type are not applied by FAMILYMANAGER to non-resident file entries. Equally some Rule actions are not applicable to resident file entries and again, these types of rules are not applied to resident files.

File entries, which are resident on a disk or diskpack family, may be the subject of the following rule actions:

ACTION	DESCRIPTION
DESTROY	Remove from mass storage, and ensure no valid backup exists. In effect DESTROY removes all trace of a file. A user may not specify a DESTROY rule, only the Rules Manager may create rules of this type. The reason for this Rule type is to allow the Rule Manager to control dangerous files. ZAP is a non preferred synonym.
REMOVE	Acts without regard to the number of backups a file has. Often REMOVE rules are coupled with DONTBACKUP rules to allow temporary files to be resident for a limited time period only. If a file is due to be removed after the scan phase of Familymanager but is then subsequently modified, the Familymanager removal job will detect the change in the file timestamp and will display a warning message instead of removing the file.
RELEASE	Remove from mass storage, after ensuring that at least one backup exists.
FORGET	Deletes any backup references for this entry. It is equivalent to DONTBACKUP for resident entries with no backup references.
OFFLINE	Mark this entry to be copied to a new tape, then removed and forgotten from the catalog.
DONTBACKUP	Do not make backups of this file. It is a way of overriding the FAMILYMANAGER default of creating two backups of every file entry. DONTBACKUP is used where the cost of backing up a file is not warranted. A typical application is to define DONTBACKUP Rules for temporary work files.
RIP	Files marked RIP are released last if the segment limit mechanism of FAMILYMANAGER comes into play. RIP stands for Retain If Possible.

The only default actions for resident files are BACKUP and if necessary the file will be crunched. Neither of these actions can be specified as rule actions.

File entries which are non-resident on a disk or diskpack family, or file entries on the family TAPE may be the subject of the following rules types:

Action	Description
DESTROY	Ensure no valid backup exists. For non-resident files it is equivalent to FORGET. Only the Rule Manager may define DESTROY Rules. ZAP is a non preferred synonym.
FORGET	FORGET means CATALOG DELETE a file entry, thus forgetting backup references. FORGET rules act to make tape space available. If no FORGET rules are defined, the only way to limit the growth of the system catalog is by having users manually cleaning up their directories.
OFFLINE	means mark this entry to be loaded and copied to a new tape, then forgotten from the catalog.
LOAD	(Disk or Pack file entry only). LOAD means make resident by COPYING from a backup volume. Only the Rule Manager may define LOAD Rules. Typical applications of LOAD Rules include making files with only one backup resident so that the next FAMILYMANAGER run will create second backup, or pre-loading files needed for regular scheduled activities. For example, a site might run a payroll program once a month on a fixed day of the month.

There are no default actions for non-resident files. If no rule applies no action will be taken.

<Rule Message>:

————— <Opal string> ————— |

As FAMILYMANAGER processes the file entries of a usercode directory, the Rules will cause it to take decisions on the disposition of some of the files. These decisions, known as Rulings, are reported to the user, along with a list of the rules that were applied to his files. To enable a user to decide which rule caused a Ruling on one of his files the creator of a rule may cause FAMILYMANAGER to place a message against the ruling in the reports.

The message may be a simple name or it may include values of key attributes used by the rule.

A Rule message is in fact a simple OPAL program which produces a string. It has same syntax as the REPORT command of FLEX.

Each time a ‘/’ character is encountered, in an Opal string, the line is output, and a new line is started. When all lines have been constructed, the program is complete.

The comma may be used to concatenate expressions within an Opal string.

The appearance of an expression followed by an integer constant causes FLEX to adjust the length of the value of the expression to the value of the constant. For string expressions, the length is adjusted by truncating or concatenating the required number of spaces on the right. For real or integer expressions, the adjustment means:

```
STRING(<real expression>,<length>)  
or  
STRING(<integer expression>,<length>)
```

Should a line exceed the number of characters per record of the output medium (which might be a printer file, a disk file or a datacom terminal) the Rule message will be folded over several lines.

<Rule Expression>

A Rule expression is the part of a Rule which selects the file entries to which the Rule applies. It is a <boolean expression> in the OPAL language. The syntax of <Rule Expression> is exactly that of <selection condition> as used in the FLEX.

Examples

```
TRUEBACKUPS NEQ 2  
CODEFILE AND LASTACCESSDAY < 84001  
RESIDENTENTRY AND GENERATION NEQ 0
```

Rule Evaluation

FAMILYMANAGER first locates all rules which apply to the particular usercode on the specified family. The origin of the rules can be as follows.

The site manager, running under a privileged usercode and an accesscode known to FLEX as the Rules Manager, can set up rules for all users on a given family. These have first priority over all other sources of rules.

The Group Leader may also create rules for a specified user on any given family. These have second priority in evaluation, that is, they may not override the general rules for the family.

Finally if the Group Leader permits, the user may make rules to manage his own files. These rules are evaluated if no site specified rules apply. Thus the user cannot override the site manager, although he can negotiate with the site manager to modify the global rules on his behalf. Only the Rules Manager may define DESTROY or LOAD rules.

Having located the rules from all possible sources, FAMILYMANAGER proceeds to examine each file entry for each file under the usercode directory on the specified

family in turn. If FAMILYMANAGER is processing a DUMP command, both resident and non resident file entries are examined, otherwise only the resident files are examined.

For each such eligible file entry, FAMILYMANAGER first evaluates all DESTROY rules. If any apply the action taken will be DESTROY and all further tests are unnecessary. Note that the overall DESTROY rules for the family are evaluated first, then any DESTROY rules specified by the Rules Manager uniquely for the given usercode. If no DESTROY rules apply, then all REMOVE rules are evaluated. Should any REMOVE rule apply no further tests are performed. The effect of this ordering is that any DESTROY rule overrides any REMOVE rule and so on.

The complete priority ordering is:

Resident Entries:	Non-resident Entries:
DESTROY	DESTROY
REMOVE	FORGET
RELEASE	OFFLINE
FORGET (=DONTBACKUP)	LOAD
OFFLINE	
RIP	

For resident file entries, if no rule applies, the Catalog block of the file will be examined to ascertain if this particular generation has two good backup references.

Although there may be two backup references in the catalog block, this does not guarantee that they are both “good”. A backup volume may be scratched without changing the backup references of files pointing to the backup volume. A tape volume may be marked Destroyed in the Volume Library. The Volume identification might not conform to FAMILYMANAGER naming standards. Either of these reasons is sufficient for FAMILYMANAGER to deem a further backup necessary. If one of the backup references points to a “good” volume and the other does not, FAMILYMANAGER may exchange the backups so that the action of making a new backup will “push” the “bad” backup reference out of the catalog block.

The order in which rules are evaluated is designed to give priority to making disk space available. The second priority is to decrease the number of catalog blocks, thus freeing up backup volumes where possible.

Rules Examples

This section describes some near universal rules that, with minor modifications apply to most sites.

Global Rules

```
DESTROY "CHECK files":  
    TITLE INCL "FLEX/CHECKFILE" And DAYS(Today,CREATEDAY) Gtr 7;  
FORGET "DUMP disk file":  
    TITLE = "*DNDISK" OR FILEID(TITLE,1)="DP";  
FORGET "XREF files":  
    ("XREFFILES" ISIN TITLE AND NOT "MCP" ISIN TITLE) OR  
    FILEID(TITLE,1) EQL "XREF";
```

Family Rules

```
DONTBACKUP "UNISYS and CONTAINERDATA files":  
    USER = "ICXFER" Or FILEKIND=CONTAINERDATA Or  
    UPPER(FILENAMEID) TLIS ".CON"" OR  
    (FILESTRUCTURE=STREAM And MAXRECSIZE=1 And FILELENGTH Gtr 1000000);
```

User Rules

```
DESTROY "OLD TEST CODEFILES":  
    (FILEID(TITLE,1)="OBJECT" AND "TEST" IsIn Title AND  
     DAYS(TODAY,USEDAY)>30);
```

RULES command

The RULES command allows users and the Rules Manager to manipulate data in the Rules Base. The Rules Base is a file used by the FLEX package to store the rules entered by the Rules Manager and individual users. It is an ordinary data file, and its structure is invisible to the users of the FLEX package. The Rules Manager is the user of a designated privileged usercode running with the accesscode RULES. The actual usercode is normally specified during installation of FLEX.

There are three types of records in the Rules Base: USER;FAMILY and GLOBAL.

- Global Rule Blocks store rules which are applied to all families. Global rules may only be changed by the Rules Manager
- Family Rule Blocks store the rules applicable to a family. FAMILYMANAGER will refuse to run on a family for which there is not a corresponding Rules Block. Options which effect how FAMILYMANAGER will deal with a family are also stored in a Family Rules Block. Family Rule Blocks may be manipulated only by the Rules Manager.
- User Rule Blocks are where options and Rules applicable to one particular user on one family are stored by FLEX. Since files of one usercode may appear on many families, there can be many User Rule Blocks corresponding to one

usercode. Thus the rules and/or options applied by FAMILYMANAGER to one user's files may be quite different from one family to another.

The RULES command has many variants, most of which are unavailable to any user but the Rules Manager. The first section covers the variants, which are generally available; and is followed by a section on the privileged variants, which are permitted only to Group Leaders or the Rules Manager. The RULES OP command, which is used to set and reset options for both Family Rules Blocks and User Rule Blocks may be used by all users, although not all options may be manipulated by any user but the Rules Manager, is covered in its own section.

Increase the maximum size of the RULES file 10x. This requires a new RULES file version, since the avail markers were unable to mark that large an avail area. Please save your RULES file prior to first running this.

After initial hash table reading, scan the RULES file sequentially to validate individual RULES and available marker records. This greatly reduces the IO time needed to check out the RULES file (while increasing the CPU time needed).

FAMILYMANAGER will now treat files with LOCKEDFILE set as if they were RIP. If the file satisfies a DESTROY, REMOVE, or RELEASE rule it will be removed. SYSTEMFILE files are now included in the INUSE report.

Global rules

GLOBAL rules are a method of specifying rules which apply to all disk families. They are referenced using the same RULES commands, except that the keyword GLOBAL is used in place of the FAMILY <familyname> clause. The Rules Manager can set up a GLOBAL rules block, which will then act as family rules for disks. Note that GLOBAL rules do not apply to the TAPE family. He can also RULES ENABLE a user so that the user can have or specify global rules.

The user can have SITE or USER rules, but GROUP rules are not yet implemented in GLOBAL rules.

An exception occurs in RULES TEST, where both GLOBAL and a FAMILY must be specified. The FAMILY specifies which disk will be scanned, and after the scan the GLOBAL rules will be marked as TESTED. RULES OP can be specified for GLOBAL, but only takes effect for families where a user has no FAMILY RULES block.

GLOBAL rules for each rule type are executed after the FAMILY rules. Thus the sequence of rule execution is as follows:

DESTROY SITE FAMILY

DESTROY SITE USER FAMILY

DESTROY GROUP USER FAMILY

DESTROY USER FAMILY

DESTROY GLOBAL

DESTROY SITE USER GLOBAL

DESTROY USER GLOBAL

REMOVE SITE FAMILY ..

where DESTROY FAMILY means a DESTROY rule for the family, DESTROY SITE USER FAMILY is a DESTROY rule for a user specified by the Rules manager for a single family etc.

Summary of RULES Commands

Used by	Command	Brief description
U,G	CONFIRM	let FAMILYMANAGER use current rules for <usercode>
G,R	ENABLE	allow a user to enter their own rules
ALL	ENTER	enter rules into rules base from a file
G,R	FORGET	completely eliminate a user or family rules block
G,R	GROUP	allows groups of users to be defined
ALL	LIST	show contents of Rules base for user or family
ALL	OP	set defaults or options to be used by FAMILYMANAGER
ALL	SAVE	remove user or family rules and save in a file
G	SHARE	set the share number for a user in a group
ALL	TEST	simulate effect of FAMILYMANAGER with new rules
R	THRESHOLD	set the percentage of available space on a pack

Each of the above commands must be preceded by 'RULES', e.g. 'RULES CONFIRM'. The column marked "Used by" means that the indicated sub-command is for use only by the following types:

Abbreviation	Meaning
R	Rules Manager
G	Group Leader
U	Normal User
ALL	All the above user types

RULES - User variants

This section discusses the variants of the RULES command that are of prime

interest to the general user of FLEX. They deal with Rules defined by the user for his own files.

The commands allow the user to enter Rules into the Rules Base (RULES ENTER command); to test his rules, once entered, by simulating their effect (RULES TEST); and having performed a test, to “put them into production” (RULES CONFIRM); to see the current rules (RULES LIST); to check and modify certain options (RULES OP); and to delete rules from the Rules Base by copying them into a CANDE file (RULES SAVE).

All these commands can be modified by a FAMILY clause. In the Rules Base, the information is kept in blocks. Each user has a block for each family. If the FAMILY clause is omitted, the family assumed if the primary family of the current FLEX Inquiry session.

One important note — no user is permitted to use these commands until the Rules Manager has explicitly enabled it. This is done by means of the RULES ENABLE command.

RULES CONFIRM

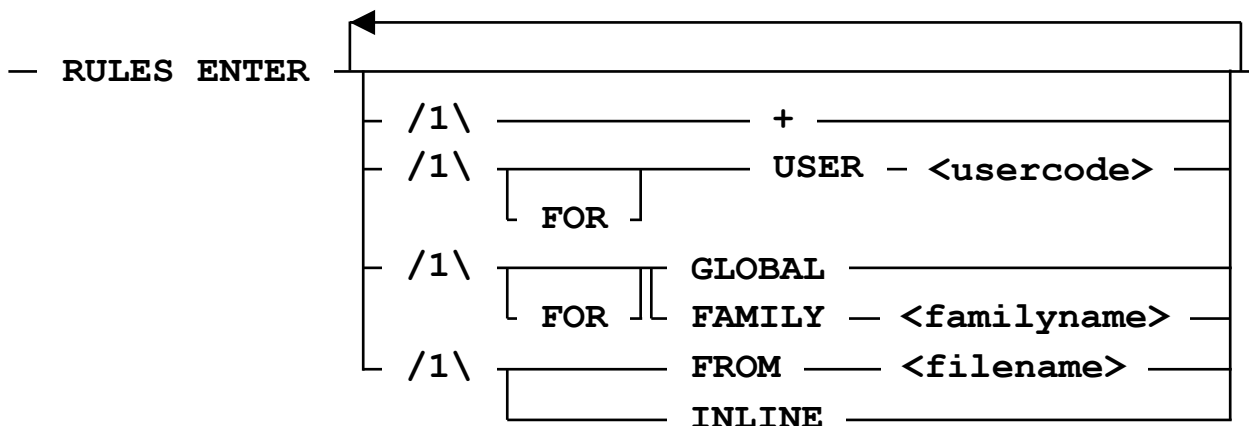


The RULES CONFIRM command is used to indicate to FLEX FAMILYMANAGER that the rules are to be used in subsequent live runs of FLEX FAMILYMANAGER.

The corresponding rules must be TESTed by using the RULES TEST command before this command will be allowed. The request will also be disallowed if the RULES OP NOREPORT has been used to turn off FAMILYMANAGER’s reporting to the user.

Note that if the FAMILY part is omitted, the current primary family will be used as default.

RULES ENTER



The RULES ENTER command allows the user to input rules into the RULES base. INLINE is used if this command is in a DO file. The records after the command will

be used for the input rules, until a record which starts with "%EOF%" in columns 1 to 5 is encountered.

GLOBAL indicates rules for all families.

The ENTER file may have any symbolic FILEKIND. If no 'FROM <filename>' or INLINE is specified, and there is a CANDE workfile, FLEX will use the workfile for the text of the rules.

Note that for the <Rules Manager>, if a FAMILY is specified without also specifying a USER, these rules will apply to the whole family, i.e. they become the new SITE rules for the given family.

If there are errors in the compilation of the rules, the rules will NOT be stored, and the errors will be listed in a file called ERRORS/RULES.

The new rules will be replace the already existing ones unless the '+' IS USED.

Beware leaving duplicate rules in that case. The result should be checked with a RULES LIST command.

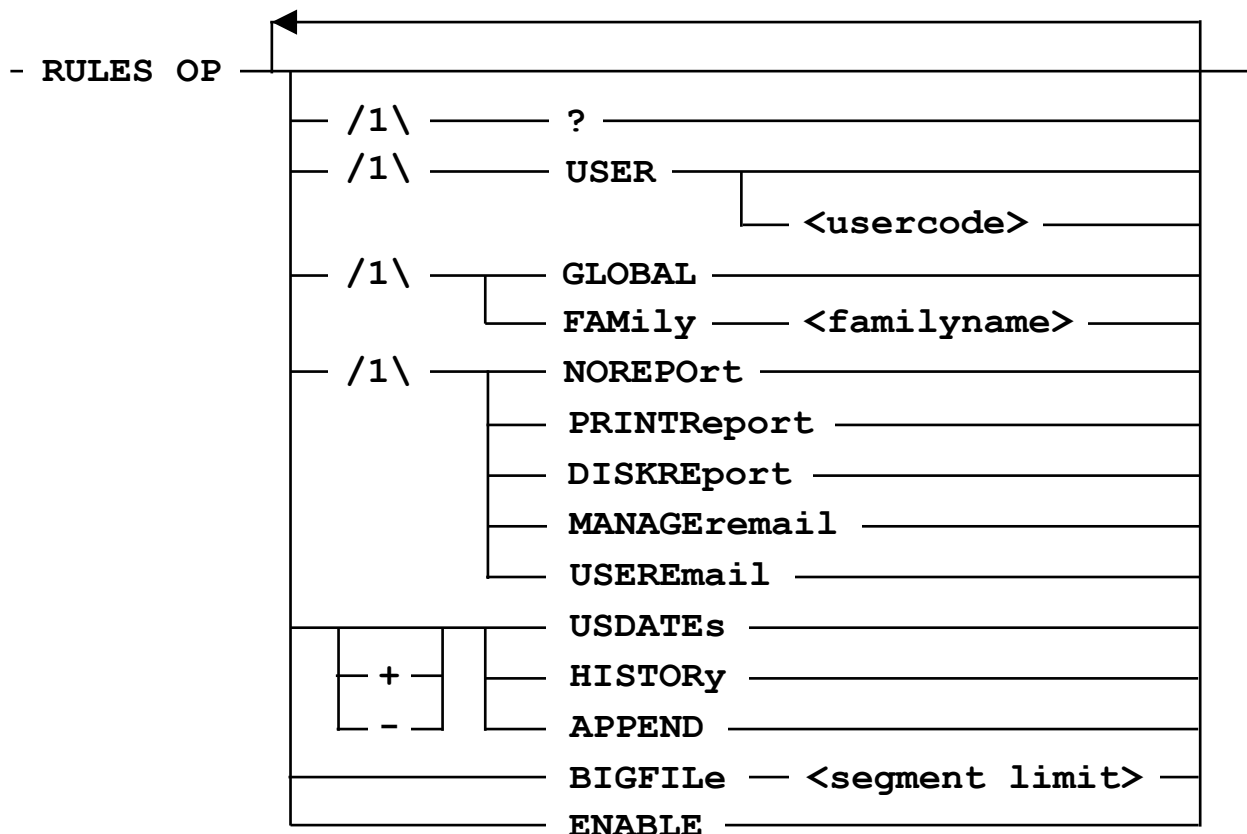
With the exception of rules entered by the rules manager, all extant rules are deactivated until a RULES TEST and a RULES CONFIRM are done.

RULES LIST

```
- RULES LIST [ USER <usercode> ] [ FAMILY <familyname> ]  
[ GLOBAL ]
```

The RULES LIST command allows the user or Rules Manager to see the rules that are currently in the rules base. Only the Rules Manager or a Group Manager may use the USER variant. Only the Rules Manager may use the Global variant.

RULES OP



The <RULES OP> command allows the user to set various options that affect the way FLEX will handle an individual user's rules.

The options allowed to a regular user all affect the reports produced by FAMILYMANAGER which show the effects it had on their files.

These options are as follows :

USDATEs	Print dates in US format, i.e. MM/DD/YY
HISTORY	FAMILYMANAGER will store summary information in a file titled:<Rules file title>/HISTORY/<familyname> E.g. Rules file : *METALOGIC/FLEX/RULES ON USER History file : *METALOGIC/FLEX/RULES/HISTORY/SRC ON USER
APPEND	FAMILYMANAGER will append data to the History file. If reset, the History file will be overwritten at each run.
NOREPORT	Suppress the reports produced by FAMILYMANAGER
PRINTREPORT	Sends reports to printer (this is the default)
DISKREPORT	Sends reports to a disk file under the appropriate usercode, called FLEXREPORTSFORME/<day of week>

MANAGEREMAIL	Sends User reports in the form of HTML email messages to the Rules Manager. If the Rules Manager usercode has an email entry in the UserData file then that address is used otherwise the Usercode of the rules manager is used as the address. If set as a family option then the FamilyManager summary report is sent as email instead of being printed.
USEREMAIL	Sends User reports as an HTML email to the User. If the user has an email entry in the UserData file then that address will be used otherwise the Usercode is used as the address.

For example, if a user wanted these reports for their files on a family called BATCH sent to a disk file, using European dates they would enter:

```
RULES OP FAMILY BATCH -USDATES,LOCASE,DISKREPORT
```

The <familyname> may be left out if you want FLEX to use your default <familyname>.

The user may not suppress reports with NOREPORT if he has user defined rules.

The remaining options are only for use by the <Rules Manager>.

For the <Rules Manager>, if a FAMILY is specified without also specifying a USER, these rules will apply to the whole family.

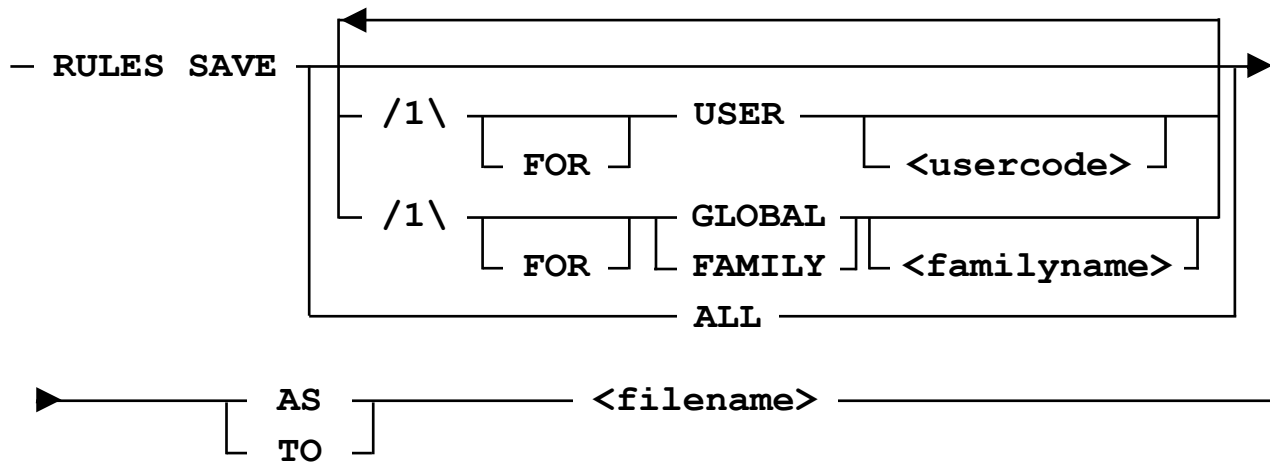
i.e. They become the new SITE options for the given family.

The ENABLE option is used by the Rules Manager to grant a user permission to create his own rules.

User created rules are applied after Site rules and cannot undo any Site rules.

If the '?' is specified a listing of the current options on the specified FAMILY is returned. Any other options (apart of course for FAMILY) specified at the same time will be parsed for syntax only.

RULES SAVE



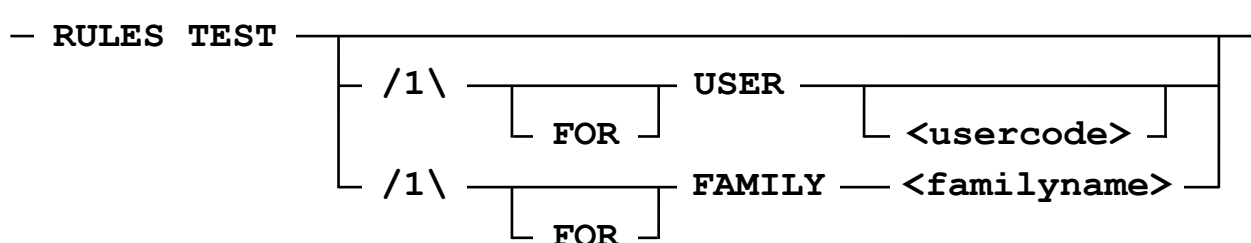
This command is used to find the rules for the specified user and/or family, and store it in a file for editing.

If the `ALL` option is not used then the current rules in the rules base will be discarded when a save is done successfully. The Rules Manager or a privileged Group Leader may include a `usercode` specification in the `<filename>`.

The `ALL` option in the `RULES SAVE` command creates a file in FLEX DO file format which if 'done' would restore all rules and options under the control of the current user.

A `RULES SAVE ALL` performed by the Rules Manager would create a file which could be used to completely recreate the rules file.

RULES TEST



The `RULES TEST` command allows the user to simulate the effect that FLEX FAMILYMANAGER will have when the user's rules are applied to his files. A report is sent to the terminal describing where the results of the test can be found. Valid Options:

- `RULES TEST`
- `RULES TEST USER <usercode>`
- `RULE TEST FAM <familyname>`
- Use of the `USER` construct Or `FAM` construct is restricted to the Rules Manager or a Group Manager.

The ordinary user needs only enter RULES TEST to run a simulation for his files.

After a RULES TEST command, the user must use the <RULES CONFIRM> command to actually activate any User specified rules.

RULES TEST FAM <familyname> reports in different ways depending on the reporting option set for the Family. See RULE OP ? FAM <familyname> If MANAGEREMAIL or USEREMAIL is set then all user reports and the summary report is emailed to the Rules Manager. Otherwies all user reports are written to disk under the rules manager usercode on family being tested. The summary is printed.

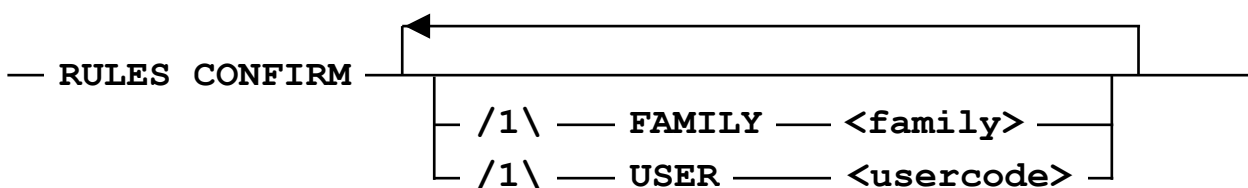
RULES - Group Leader variants

A Group Leader, or the Rules Manager, are users of FLEX and so may use the User variants above to manage their own usercode. However, they are given special privileges by certain reserved variants of the RULES commands.

These variants of the RULES command are only available to the Group Leaders, or the Rules Manager in his position as the highest level Group Leader. These commands allow Group Leaders: to permit a user, *which is directly owned*, to enter rules for himself (RULES ENABLE); to enter rules for a directly owned user (RULES ENTER); to delete such a user's Rules Block from the Rules Base (RULES FORGET); to set or modify the shares of a user on a family (RULES SHARE); to determine the space ownership hierarchy of the users which he owns (RULES GROUP).

No Group Leader, including the Rules Manager, can use these commands on a user which he does not directly own. For a example if Group Leader AA owns Group Leader BB, and Group Leader BB owns Group Leader CC, AA cannot modify CC

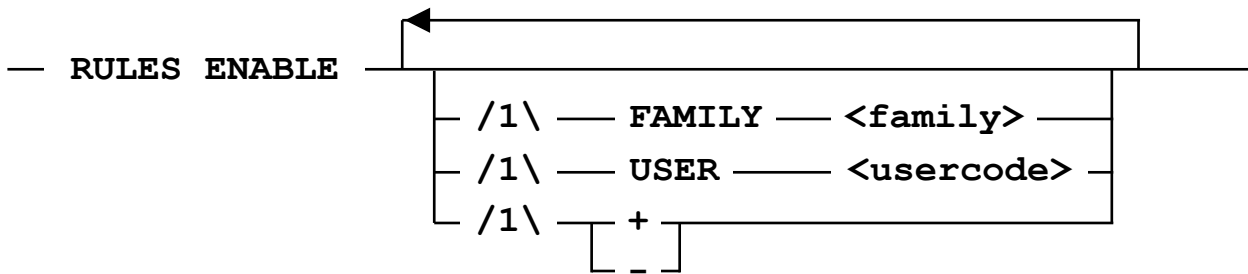
Group RULES CONFIRM



The RULES CONFIRM command for Group Leaders allows them to activate the rules of their owned users. As in the normal user variant, the rules must be tested before they can be confirmed. The request will also be disallowed if the RULES OP NOREPORT has been used to turn off FAMILYMANAGER's reporting to the user.

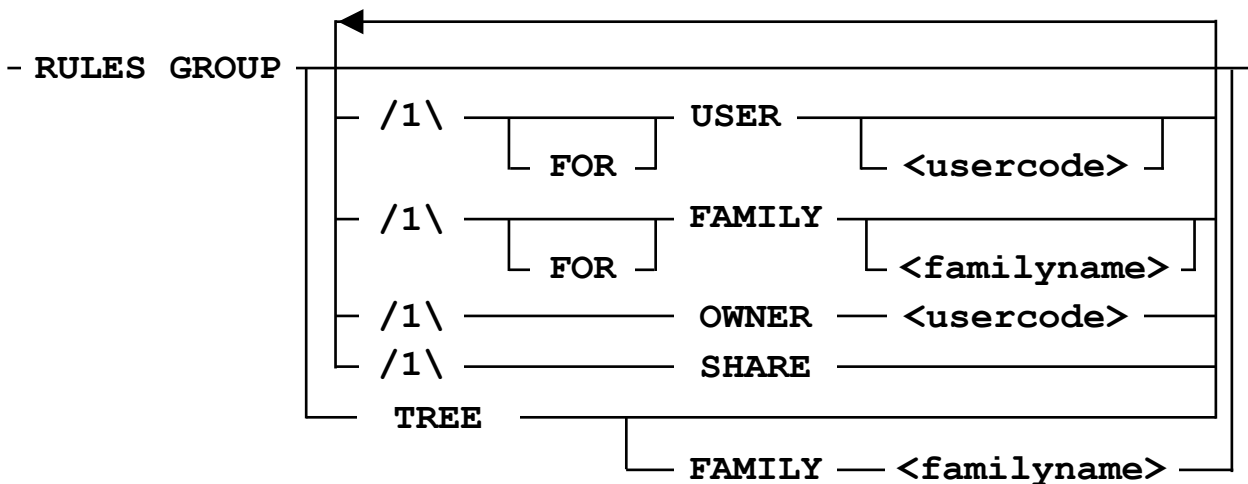
Note that if either the <usercode> part or <familyname> part is omitted, the current <usercode> running FLEX, and the corresponding <familyname> will be used as default.

Group RULES ENABLE



The **RULES ENABLE** command is used by the Group Leader to turn on (or off with the **'-'** variant) a user's ability to enter Rules for his own usercode. If no **'+'** or **'-'** appears, **'+'** is assumed.

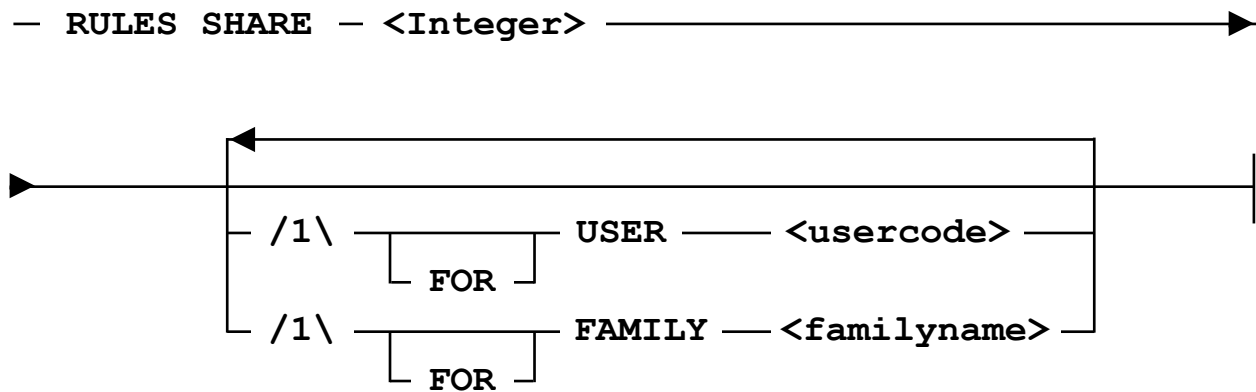
RULES GROUP



The **RULES GROUP** command allows a Group Leader to define a group of users. The owner of a group has the power to delegate ownership of subgroups to other members of his group. Only the immediate Group Leader for a user may allocate **SHARE** numbers, enter rule definitions with the **RULE ENTER** command and change options with the **RULES OP** command.

The **TREE** variant of this command prints a summary report from the History file. (see **RULES OP HISTORY**). The report is organised by group, and only descendants of the Group Leader are shown. Thus, if this command is entered by the Rules Manager, all groups and users will be shown.

Group RULES SHARE



The `RULES SHARE` command allows a Group Leader to set the `<share number>` for a user. Users in a group share the space allocated to their Group in proportion to their `<share numbers>`s. Only the most immediate owner can allocate a share to a user. Shares can be set by this command for users of an existing group by the Group Leader. To set the shares of a group, or when initialising shares when defining a group, the shares are allocated by the `SHARE` variant of the `RULES GROUP` command.

The Rules Manager allocates share numbers to users who are not owned by any other user. Users which are not in any hierarchy and have not an assigned share, are considered to be owned by the Rules manager with a single share (1). System files are considered to belong to the un-owned usercode "SYSTEM-NODE". It is usually initialised in the FLEX install process, which determines an appropriate share allocation.

Share numbers are used by `FAMILYMANAGER` to compute what files it should release from families where the percentage of available space is less than the `THRESHOLD` value.

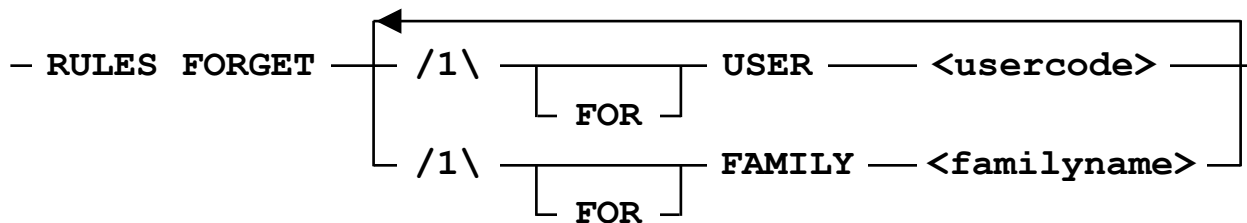
RULES - Manager variants

These variants of the `RULES` command are only available to the Rules Manager. They allow him to deal with the family as a whole. He may: enter rules for a family (`RULES ENTER`); delete a User Rules Block or Family Rules Block from the Rules Base (`RULE FORGET`); determine the point at which files will be released to make more space (`RULES THRESHOLD`); and set options valid for the whole family (`RULES OP`). Another privilege of the Rules Manager, is ownership of all otherwise unowned users, so he has access to all the Group Leader and User variants above as well. In some case there is an ambiguity in meaning between a User and a Rules Manager variant. If the Rules Manager wishes to access the User variant, he should temporarily change or remove his access code.

There is a possible side effect to the Rules Manager using FLEX Inquiry. When `FAMILYMANAGER` commences executing a `DUMP`, `BACKUP`, or `CLEANUP`

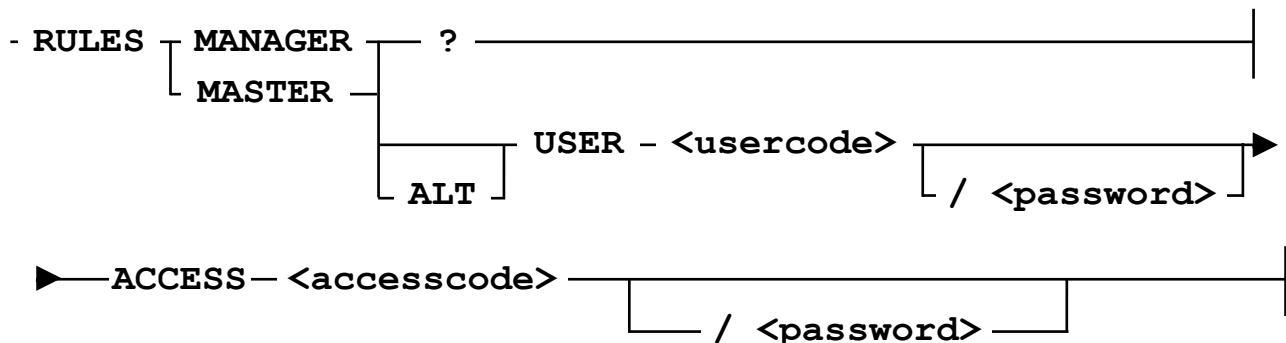
command, it “locks” the Rule Base to any changes so that there can be no ambiguity about which Rules are current. If FAMILYMANAGER terminates abnormally, the lock remains in force until FLEX Inquiry is run once from a privileged usercode. If the stack that had the lock is no longer in the mix, this run will unlock it. This side effect will thus always occur if the Rules manager usercode runs FLEX Inquiry. The lock also prevents two FAMILYMANAGERs running at the same time.

RULES FORGET



The `RULES FORGET` command allows the Rules Manager to discard a Rules Block (either a User Rules Block or Family Rules Block). This is the only way to remove a block that has been inadvertently been entered (for example, a spelling error).

RULES MANAGER



The `RULES MANAGER` command allows the Rules Manager to change either the primary or alternate Rules Manager usercode and accesscode. Both usercode and accesscode are validated in the Userdatafile so passwords must be provided if applicable. A Rules Manager usercode must have PU capability.

Any attempts to change or set an invalid Rule Manager will cause FLEX to generate a `SECURITY VIOLATION` message and the process suspended.

The command '`RULES MANAGER ?`' will show current Rules Manager details if run by a FLEX usercode that is PU or SECADMIN.

A typical response might be:

```
RULES MANAGER ?
-- Rules Management details --
Master Usercode:      FLEX
Master Accesscode:    RULES
Alternate Usercode:    Not Set
Alternate Accesscode: Not Set
#
```

Further details and syntax can now be seen using the `HELP RULES MANAGER` command in FLEX Inquiry.

RULES THRESHOLD



The `RULES THRESHOLD` command allows the Rules Manager to specify the percentage of available space which FAMILYMANAGER should maintain on a

family. If there is insufficient available space FAMILYMANAGER will release files of users who exceed their share in proportion to their share numbers.

FAMILYMANAGER

The FAMILYMANAGER utility is that component of FLEX which manages the files of all users at an installation, based on "rules" defined by the site management and the users themselves. FLEX rules control whether files show be backed up, removed or deleted from the catalog. Several hard-wired rules are pre-defined (e.g. every file must have 2 good backups except DMSII DBDATA files) but it is the responsibility of the site to set up and maintain its own rule database.

FAMILYMANAGER may also be run to produce file usage reports for perusal by both site management and individual users, to assist with the site rule definition process.

Pack family recovery is an important component of FLEX and, should a member of a multiple pack family become unavailable, it is the role of FAMILYMANAGER role to identify those affected files with rows on the missing pack. This information is used to remove such files and reload them from backup volumes.

Since the FLEX package is based on the MCP's CATALOG subsystem, FAMILYMANAGER is aware of and can operate on disk files, which are not currently on-line, as well as tape files.

When requested to manage a family, FAMILYMANAGER will take at most two actions by default:

- Make backups of any file with less than two good backups (except DMSII DBDATA)
- Crunch files which waste space (if the NOCRUNCH option is set)

The NOCRUNCH option can be set or reset using the DEFAULT command from the Flex Inquiry utility.

If a Flex tape, CD or CD image was marked as DESTROYED in the Volume Library or assigned as BAD BACKUP in the Statistics files, FAMILYMANAGER is meant to re-backup any of the resident files that were present on the damaged media. Unfortunately, this mechanism was only working correctly for tape media, CD and CD image volumes would be ignored. This problem is now fixed.

Also, during FAMILYMANAGER runs that create Copywrite disk images, FAMILYMANAGER will now verify that all disk containers found in the Volume Library are present on the image family. If any containers are missing, each will be marked as DESTROYED in the Volume Library and any resident files that had been backed up on the missing image will be reprocessed. FAMILYMANAGER will generate appropriate display messages when such containers are found.

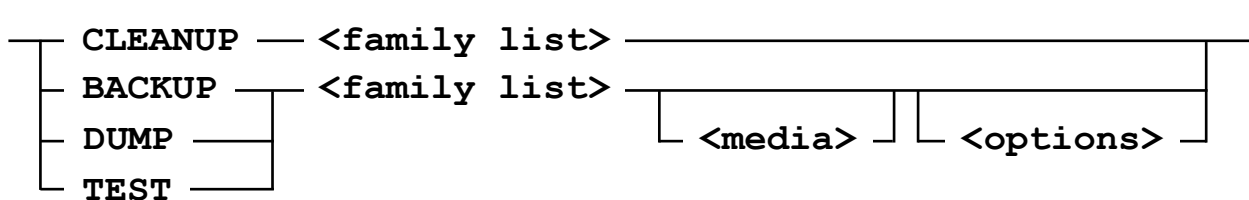
Familymanager will not CATALOG ADD any file it rules on. It will only ADD files that it has decided to backup. The call on ADD is done from the JOBBUILDER stack.

The volume library is now scanned at least one per day to check for serials which might be reused in addition to when Flex Library was restarted.

FAMILYMANAGER automatically excludes certain file kinds from being backed up at any time. For example, files such as DBDATA, JOBDESC,SYSTEMDIRECTORY etc. In the interest of file safety, all such files will be exempt from *any* Flex rule during a FAMILYMANAGER run.

Commands

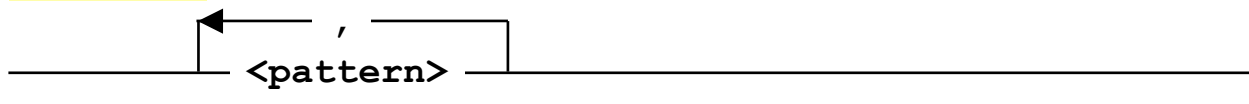
FAMILYMANAGER has an extensive syntax, allowing the site to backup and housekeep files on one or more families in a single run. The BACKUP and DUMP commands have extensive syntax allowing options and overrides to be applied to both file selection and output media.



FAMILYMANAGER can be driven entirely by information in the Rules File or in parameter syntax. Most parameters which were given by label equation are now in the Rules File. In particular, the density, length, recovery strategy (ex PROTECTION), SAVEFACTOR, and SCRATCHPOOL of the output tapes, as well as the use of &COMPARE or &VERIFY, and the restriction to a single output tape. The activation of the BIGFILES and HISTORY features is now solely dependent on the RULES OP settings for the family concerned.

Two new keywords are allowed after the familyname in the parameter. They are DUPLICATE, which causes the output tapes to be duplicated, and OFFSITE (ex BLOCKSTRUCTURE=EXTERNAL), which marks the tapes OFFSITE.

<family list>



A <family list> permits individual families or pattern-matching to be given instead of a single family. Wild-card patterns are provided to allow multiple families to be specified with a single pattern.

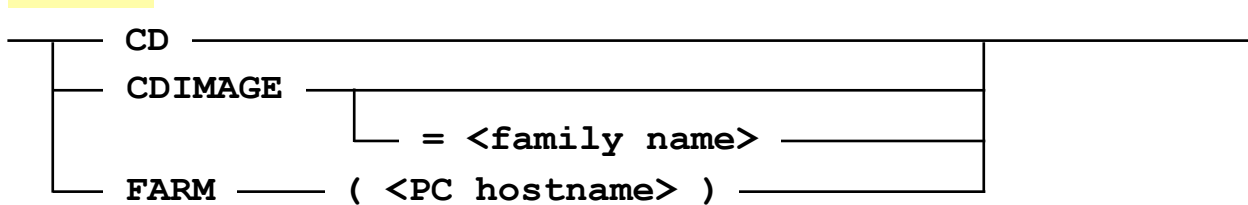
This feature is not available for backups to CDImage

For example, 'DISK=' selects DISKA, DISKB, DISKC and 'S=' selects SYSTEM, SOURCES etc. See Appendix A for a description of wild card characters.

```

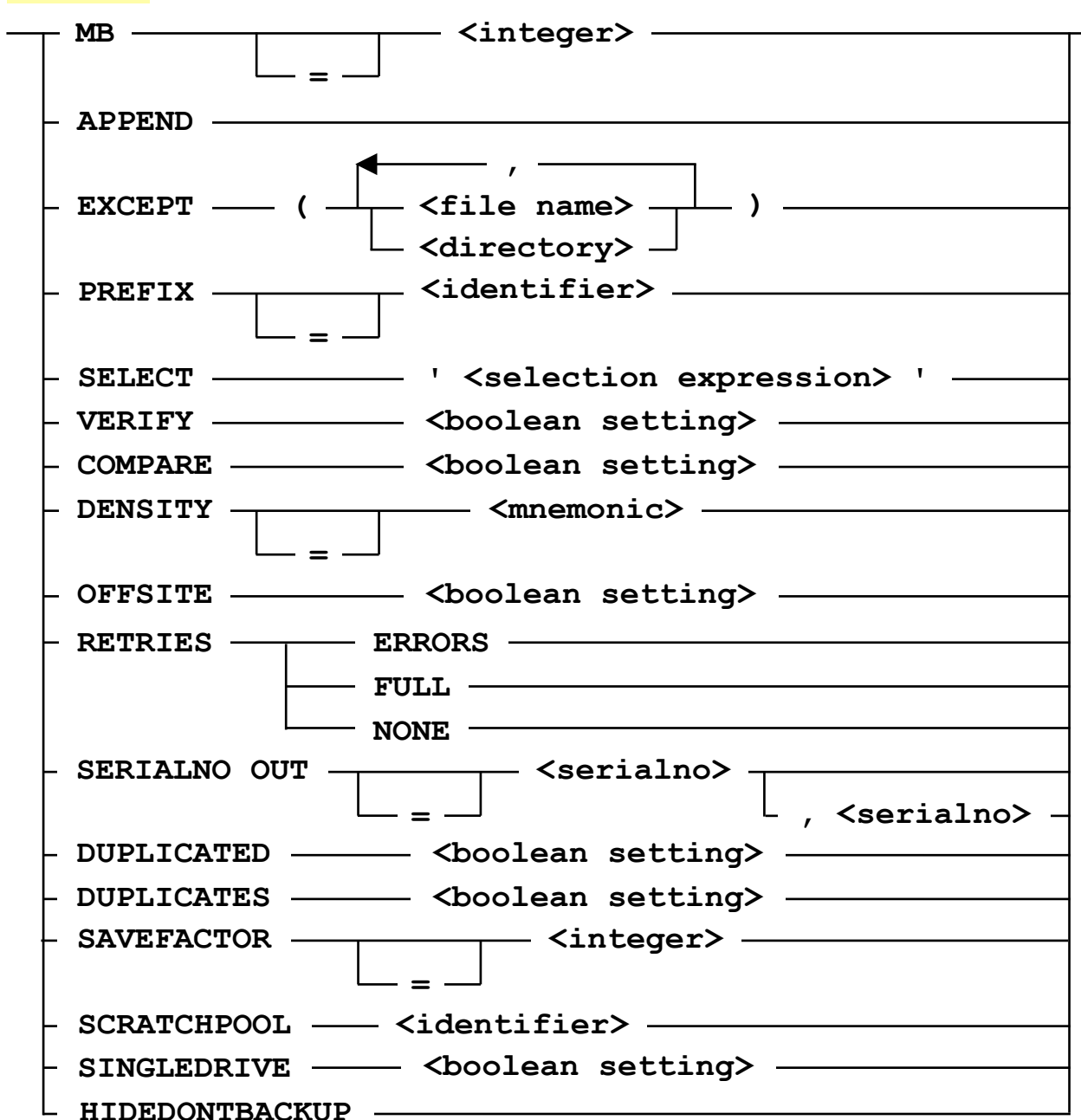
DISK=
DEV, TEST=, DISK
S=
  
```

<media>



The <media> modifier allows simple selection of output to media other than tape (default). CD initiates backup to physical CD media, CDIMAGE to on-line disk storage (or to an alternate image family) and FARM to remote disk storage on the PC system specified.

<options>



<boolean setting>

<input type="checkbox"/>	=	<input type="checkbox"/>
<input type="checkbox"/>		ON
<input type="checkbox"/>		OFF
<input type="checkbox"/>		TRUE
<input type="checkbox"/>		FALSE

BACKUP Command

The BACKUP Command examines only resident files of the specified family. It does not process LOAD rules and DESTROY or FORGET rules are applied only to resident file entries. A FORGET rule applied to a resident files is equivalent to DONTBACKUP. BACKUP is intended for the regular daily “security” backups, or in combination with a selection expression, consolidation backups which would normally be duplicated thus allowing one to be stored off site.

If any resident files are found which will be backed up ,while in a DUMP or BACKUP command, FAMILYMANAGER will CATALOG ADD them.

CLEANUP Command

The CLEANUP Command simply looks for files which can be removed or crunched from the specified family. No other file manipulation is performed and no reports are produced by this command. CLEANUP is intended as an operations aid when space becomes short on a high activity family.

DUMP Command

The DUMP command process all rule types for all file entries on the specified family. Since it is examining resident and non-resident file entries, it will take correspondingly longer than the other variants. In many cases it suffices to run FAMILYMANAGER with a DUMP command only occasionally, and to use the BACKUP Command to make the regular daily security backups.

If any resident files are found which will be backed up,while in a DUMP or BACKUP command, FAMILYMANAGER will CATALOG ADD them.

TEST Command

The TEST command behaves like the DUMP command but no actions are taken. TEST reports are generated and Jobs are generated but not started.

Media types

FAMILYMANAGER supports the usage of SCRATCHPOOL to optionally assist in the usage of scratch tapes for the normal FAMILYMANAGER WFL tapes.

By purging all FLEX tapes with a SCRATCHPOOL identifier, for example:

```
PG MT 67 SCRATCHPOOL = FLEX_POOL
```

then FAMILYMANAGER can be told to only use tapes from that SCRATCHPOOL by setting the option in the Familymanager parameter:

```
SCRATCHPOOL FLEX_POOL
```

If the above option is used, then the FAMILYMANAGER created WFL jobs will only request FLEX_POOL scratch tapes improving security (avoids accidental OU-ing to the wrong scratch tape) and responsiveness (tapes are automatically picked up even if the SERIALNUMBER (28) system option is set).

For a Boolean option where neither ON nor OFF is specified, ON is assumed.

Either DUPLICATES or DUPLICATED in the parameter options will cause duplicated backups to be created.

MB=0 option in Familymanager will override a non zero default MB

CD

FAMILYMANAGER may use CDs for its output volumes if COPYWRITE is available. CD output is selected if the DEFAULT DENSITY is 64 or if the CD is specified in <media>.

When the output media is CD, only one single source family may be specified. The settings of VERIFY and COMPARE are ignored because the CD format has its own adequate checks. The CD images are put on the pack specified by the CONFIG variable FLEX_CWIMAGE, and these images.

For CDs the backup references are put in by the MERGETAPE run after the COPY is complete. MERGETAPE hangs after the COPY waiting for the newly burnt CDROM to be mounted on-line. Once it appears, it does a COPYWRITE "ADD&CATALOG *= TO NULL" to insert the backup refs. Because CDs have no SERIALNO in their label, it is encoded in the Volume Name.

These are of the form:

CDSSSSSSSF1998333A

where SSSSSS is the SERIALNO

F indicates FAMILYMANAGER is the creator (M is MERGETAPE)

1998333 is a 7 digit Julian Date

A is the relative volume this run, First A, then B,C,...

Because it is very efficient to make duplicates on the PC that burns the CD,

FAMILYMANAGER does not make the duplicates itself for CDs.

When a CD is marked as being from a DUPLICATED run, the same SERIALNO is entered twice for the file entry.

CDs tend to be more labour intensive to make, but are much more durable so are very well suited to long-term storage. They are also extremely fast on LOADs, and have very cheap media, readers and burners.

CD Images

For DUMP and BACKUP activities to CD disk images (using the CDIMAGE modifier, FAMILYMANAGER will use the family specified by the Magus config variable FLEX_CWIMAGE as image file destination. It is possible to override this setting using the following modified '= <family name>' modifier with the CDIMAGE option.

When an alternate CD image family is specified, the CD images are generated as normal on the default image family and then automatically COPY&BACKUP by CopyWrite to the alternate family. Copywrite removes the image from the original family but leaving a non-resident file entry pointing to the alternate family. This is referred to in Metalogic software as a 'HARD LINK'. You may specify as many alternate families as required (as long as the family is volumed).

In this example, the default image family is DISK and an alternate image family is called ALTIMAGE. After creation of the image, a PD of the image on each family appears as:

```
PD *DEV/CI0017F2004142A
FILE *DEV/CI0017F2004142A ON DISK (PROMBURNERDATA)
NOT RESIDENT
FAMILY SERIAL NUMBER: 000505
ENTRY 1 (PROMBURNERDATA):
CYCLE: 1  VERSION: 0
SECURITY: PRIVATE
TIMESTAMP:  Friday, May 21, 2004 (2004142)  AT 11:52:15
LASTACCESS: Friday, May 21, 2004 (2004142)  AT 11:52:02
  BACKUP MEDIA IS: PK
  SERIAL: 000600
  BACKUP MEDIA IS: PK
  SERIAL: 000600
```

On ALTIMAGE (pack serial number 600):

```
PD *DEV/CI0017F2004142A ON ALTIMAGE
FILE *DEV/CI0017F2004142A ON ALTIMAGE (PROMBURNERDATA)
```

```

CREATION DATE= Friday, May 21, 2004 AT 11:52:01 BST
LASTACCESS DATE= Friday, May 21, 2004 AT 11:52:01 BST
ALTER DATE= Friday, May 21, 2004 AT 11:52:01 BST
ATTMODIFY DATE= Friday, May 21, 2004 AT 11:52:16 BST
BACKUP DATE= Friday, May 21, 2004 AT 11:52:01 BST
COPYDEST DATE= Friday, May 21, 2004 AT 11:52:01 BST
COPYSOURCE DATE= Friday, May 21, 2004 AT 11:52:01 BST
EXECUTE DATE= Friday, May 21, 2004 AT 11:52:01 BST
READ DATE= Friday, May 21, 2004 AT 11:52:01 BST
TOTAL SECTORS: 88,896 (5,556 PER AREA)
SECURITY = OWNER *:RWX, GROUP <none>:NO, OTHER:NO
(PRIVATE - USAGE: READ/WRITE) LOCKEDFILE
FAMILY SERIAL NUMBER: 000600
ENTRY 1 IS RESIDENT, NOT CATALOGED (PROMBURNERDATA):
CYCLE: 1 VERSION: 0
SECURITY: PRIVATE
TIMESTAMP: Friday, May 21, 2004 (2004142) AT 11:52:15
NO ARCHIVE BACKUP RECORD FOR FILE

```

If CDIMAGE is specified the Metalogic CopyWrite product is required, and files are backed up to a Copywrite CD image. If Duplicated is set, both backup references are set to the same image. The config variable FLEX_CWIMAGE is used to specify the family where the images are to be kept. The Config variable FLEX_CISERIALNO is used to specify a pattern to be used for serial numbers for images. It should be two alpha characters followed by 4 digits. Thus if FLEX_CISERIALNO=CI0000 then serialnos would have the form CInnnn. The CD images are entered in the volume library as pseudo packs so

```
PV PK CI0001
```

would give details of the image.

Flex builds a list of Serialnos to be used by scanning the Volume library for Volumes matching the pattern. Any gaps are stored as possible serianos to use.

Ex if volumes CI0000,CI0001,CI0004,CI0008 are in the volume library then CI0002,CI0003,CI0005,CI0006,CI0007 would be eligible for use.

FAMILYMANAGER will mark a CD image volume as DESTROYED if the associated image file is missing AND:

- 1. The file does NOT have any backup references or its catalog information is invalid.

- 2. The file has a Hard Link but the linked file is not present on the alternate image family.

MERGETAPE, during a CD image merge, will verify that all CD image volumes are available; a CD image volume will be omitted from any MERGE process if the associated image file is missing AND:

- 1. The CD image file does not have a valid Hard Link reference to an alternate image family.
- 2. A Hard Link reference is present but the specified family is not currently on-line or available.

When either MERGETAPE or FAMILYMANAGER check for CD image file availability, messages may be generated for any of the conditions described above:

- IMAGE <Image Name> IS NON-RESIDENT BUT HAS NO HARD LINK
- IMAGE <Image Name> NO RESIDENT FILE OR CATALOG ENTRY
- IMAGE <Image Name> HAS HARD LINK BUT FILE NON-RESIDENT
- IMAGE <Image Name> HAS HARD LINK BUT FILE IS NOT PRESENT
- IMAGE <Image Name> HAS HARD LINK BUT FAMILY OFFLINE

Lastly, any output SCRATCH volume entry added by FAMILYMANAGER or MERGETAPE into the Volume Library prior to its use will have its volume creation information updated with the current date and time.

This allows any old volumes to be safely identified if they are not subsequently used.

Also, running FAMILYMANAGER with the TEST command will produce a detailed debug listing of all files processed during this debug run.

FAMILYMANAGER will ALTER disk containers generated by Copywrite so that the LOCKEDFILE attribute is always set. This helps protect these files from accidental removal.

The default maximum size of CDIMAGE is 650 Megabytes (Million bytes). If the total size of files to be copied is greater than 650Mb then multiple copy jobs will be generated.

The MB= parameter may be used to change this value up to a maximum of 4000Mb.

If a single file is bigger than the MB limit it will be copied to an image of its own which be the size of the file. The maximum CDImage size of 4000Mb may not be exceeded.

RULES

The Rules Manager may opt to delegate his responsibility for a group of users to

one member of the group, known as the **Group Leader**. Any Group Leader can do the same thing, designating subgroups and corresponding second level Group Leaders. Thus any particular usercode might have many Group Leaders at various levels.

A Group Leader, if permitted to define and change rules for his own files, will also be able to define and change rules for a member of his group. The resulting structure resembles the organisation chart of a large corporation. In this analogy the Rules Manager is the president of the corporation. He may make rules for himself, and due to his exalted position, for any and every member of the organisation he heads. For reasons of managerial efficiency he delegates his authority over members within one division of the corporation to a division head. In large divisions the division head might further delegate authority to department heads.

Some disk pack management systems allow the enforcement of disk limits for the users of a family. A simplistic method is to allocate a fixed amount of space on a family to each usercode. When a user exceeds his segment limits on a given family, the dumping program would release disk space by dumping files to tape. This method has become outdated with the advent of the large thin film pack units. Since the total amount of available space can vary greatly if a unit is added or taken away, the use of fixed segment limits is impractical. To make the best use of the space, the site manager would constantly have to adjust the limits for each user. Furthermore, it is no longer practical to allocate a single pack to a user group, as was common in the days of 206 and 207 packs. This leads to the problems of managing space for groups of users.

FLEX FAMILYMANAGER provides a powerful way for the Rules Manager to assist in the management of the available space on a disk family. The Rules Manager does not assign fixed segments of space as in the previously described method. Instead, he nominates the percentage of available space he wishes to maintain on a family (THRESHOLD). At the beginning of each run, FAMILYMANAGER divides the remaining space into "basic shares" by allocating a portion to each user with files resident on the family. By default, if the Rules Manager does not define any groups, all basic shares are of equal size. The Rules Manager may change the relative size of a "basic share" of any user as can a GROUP LEADER for the members of his group, with the RULES SHARE command of the FLEX Inquiry. Normally the largest-least accessed files are removed first, but both the site management and the user may vary this algorithm by nominating files to be preferentially retained.

Virtually all actions on files are performed by Work Flow Language jobs generated by FAMILYMANAGER. The sole exception to this rule is that FAMILYMANAGER itself will CATALOG ADD uncataloged files.

All COPY functions are performed by LIBRARY/MAINTENANCE. By default the copies will be COPY&COMPARE&BACKUP. Options exist to separately disable either the COMPARE or the BACKUP qualification.

The backup housekeeping is such that each tape volume may be duplicated. An attempt is made to ensure no multiple reel tape volumes are produced. Two

backups will be made wherever possible. No more than two backups of a file will be made.

While FAMILYMANAGER is examining a family it can optionally collect a list of “large files”. The Rules Manager sets the threshold value a file must exceed to be considered a “large file”, on a family by family basis with the FLEX Inquiry (RULES OP BIGFILES). This list will be output as a JOBSYMBOL file (the BIGFILE file), sorted in descending file size (in segments). Each entry in the BIGFILE file contains the title, size and USEDATE of the file¹.

FAMILYMANAGER may also supply information to a site billing procedure by means of the HISTORY Feature. This option causes it to write, to a data file, a record for each usercode with resident files detailing the number of files, segments occupied by the files, both total segments and subtotals for code, symbol and data type files, the current limits in force for the user on the family in question and a timestamp. The data file may be extended indefinitely to enable year-to-date totalling.

The routine overhead of backing up files can be expensive, regardless of the method used. FAMILYMANAGER allows a major part of this overhead to be charged back to the user. This feature is optional, and is enabled by declaring FAMILYMANAGER as an MCS. A section is devoted to describing this facility.

Running FAMILYMANAGER

As most of the information required to perform its tasks is extracted by FAMILYMANAGER from the FLEX Rules Database, little is required to be supplied at run time. FAMILYMANAGER is essentially operator independent, requiring only that the particular function and the family to be ‘managed’ are supplied. This is done by means of a single string parameter. For this reason FAMILYMANAGER can be initiated from the remote terminals, the system console, as well as in Work Flow Language jobs.

FAMILYMANAGER will not operate correctly unless it detects that the usercode and accesscode under which it is running is that of the Rules Manager. A typical job looks like

```
USER=PRIV/USER ;  
ACCESSCODE=RULES/MANAGER ;  
RUN METALOGIC/FLEX/FAMILYMANAGER(<string Parameter>) ;
```

¹ Such JOBSYMBOLic files are exactly the format needed by METALOGIC's SUPERVISOR to enter a MEMO. MEMOs can be called up on the system consoles by a SUPERVISOR command. The intent here is that the operator has a list of the worst offender files on each family, which is updated relatively frequently and can be viewed from the consoles. He can then use SUPERVISOR's RELEASE command to remove files which have two safe backups.

Tape Backups

Additional optional information may be supplied to FAMILYMANAGER concerning the 'attributes' of the volumes it should use in the generated jobs, where applicable (with BACKUP and DUMP when backing up to Tape volumes). The 'attributes' which are of interest are DENSITY (BPI1250, BPI6250, BPI38000 etc.), whether duplicate backup volumes should be made, the length of the tapes to be used and the SERIAL numbers of the tapes to be used. Normally this job will be set up automatically in response to questions asked during the install phase.

These parameters may also be specified without going through INSTALL by specifying label equations. Each BACKUP or DUMP command in one run may have one independent label equation. The internal name of the file to label equate is constructed by FAMILYMANAGER using the name of the family and is

BACKUP<first 11 characters of familyname>.

For example, if a pack family named SYSTEM is to be backed up or dumped, then the internal file to be used would be called:

BACKUPSYSTEM

The label equations may be specified for each run, or they can be specified once at compile time, or they may be bound into an existing code file.

The preferred mechanism for specifying this information is via the <options> in the parameter to Familymanager.

Only certain file attributes will be interrogated by FAMILYMANAGER. They are:

ATTRIBUTE :	ATTRIBUTE INTERPRETED AS:
DENSITY	Value of DENSITY attribute to use. If not set, the density is given by the value specified at INSTALL time.
DUPLICATED	If set, two copies of each tape will be made If the output media is not tape then one copy is made but both references are set to the same volume
SIZEOFFSET	Length (in Mbytes) of tape (e.g. 550, 600, 2400, 3600). If not set, the length used is the value specified in the INSTALL.
SERIALNO	List of serial numbers of tapes to use
PROTECTION	If equal to the default value of NONE, no retry will be attempted by any COPY job in case of failure. If equal to SAVE, each COPY job will be retried if it gets a fault. If PROTECTED, the jobs will retry if the COPY job gets a fault or if any files are not copied.
BLOCKSTRUCTURE	If set to EXTERNAL, the tapes made by this run will be marked as OFFSITE in the FLEX statistics file. If DUPLICATED is also set, only the second backups will be marked OFFSITE.

ATTRIBUTE :	ATTRIBUTE INTERPRETED AS:
SAVEFACTOR	Allows the user to override the default SAVEFACTOR of 30 days for all output tapes

The default values for an attribute for backups to tape if not specified by label equations is:

```
BACKUP<familyname> (SIZEOFFSET = 2400, DUPLICATED,DENSITY =
BPI6250
, PROTECTION =
SAVE ,BLOCKSTRUCTURE=EXTERNAL) ;
```

FAMILYMANAGER now specifies the DENSITY attribute in the copy statement in the generated job. This aids sites with multiple types of tape device to always use the type of tape specified to FAMILYMANAGER. If KIND=PETAPE instead of KIND=TAPE is specified for an output cartridge (either BPI38000 or BPI1250) then the resulting waiting entry must be OU-ed, even if the system SERIALNUMBER option is reset. FAMILYMANAGER will always use KIND=TAPE for these specific densities.

SIZEOFFSET is one way of controlling how much data FAMILYMANAGER thinks that it can write to a single reel of the chosen tape medium. During the installation phase, the INSTALL utility will ask for a suggested figure for the selected tape media but can be changed using the RULES DEFAULT command from Flex Inquiry, by using the INSTALL utility or by overriding with file equation of the SIZEOFFSET attribute. FAMILYMANAGER will keep track of how much tape is being used for file backup and when the limit is reached, the current WFL COPY job will be invoked and a new job file opened to prepare the next "reel".

Suggested tape values for some tape densities are shown below, though this may vary from site to site and some experimentation may be required especially for new tape technologies:

```
BPI1250      600 feet
BPI6250      2400, 3600 feet depending on reel size
BPI38000     600 feet
```

The FAMILYMANAGER copy jobs will produce tapes with the following name format:

```
<FAMILYNAME><YYDDD><N>
```

where only the first 11 characters of the family name are used. The <YYDDD> field is a julian date and <N> is an alphabetic character representing the reel number of the dump. For example, reels 1 and 2 of a backup for the DBSPACK family would be:

```
DBSPACK97120A
DBSPACK97120B
```

If the DUPLICATED attribute is set, then two copies of each output tape will be made and the names of the output tapes change slightly with an additional letter

after the reel assignment. This is always "F" or "G":

```
DBSPACK97120AF  reel 1 of DBSPACK backup
DBSPACK97120AG  duplicated reel 1 of DBSPACK backup
```

Some users may use these letters to indicate which tape should be ONSITE and which OFFSITE. However, the action of updating the Catalog is such that the "F" tape becomes the second backup and the "G" tape the first.

If sites mark the "F" as OFFSITE then FAMILYMANAGER will "exchange" the two backup references so that the ONSITE tape is the first backup. This process can be a significant overhead for large, duplicated tape backups.

To help sites where changing their off-site rules would be very complicated, a configuration variable called "FLEX_GFIRST" can be used. If FLEX_GFIRST=TRUE then the "G" tape will be copied first thus becoming the second backup. In order to maintain consistency with MERGETAPE setting this configuration variable will cause the "M" and "N" tapes of MERGETAPE to be switched. The variable can be easily changed using the INSTALL utility:

```
U META/INSTALL FLEX_GFIRST=TRUE
```

Bigfiles and History

Further optional features of FAMILYMANAGER are enabled by dummy label equations. The BIGFILES feature is turned on and off in this way, as is the HISTORY feature. In the case of the BIGFILES feature, it is enabled if a label equation for a file whose INTNAME is related to the familyname in the following way.

```
BIG<first 14 characters of family name>
```

For example, in the case a pack named SYSTEM, the INTNAME should be BIGSYSTEM to enable this feature. No attributes except TITLE, KIND and FAMILYNAME are of interest to FAMILYMANAGER.

In the case of the HISTORY feature, it is enabled if a label equation for a file whose INTNAME is constructed as follows:

```
HIST<first 13 characters of family name>
```

For example, if the family is PACK, the INTNAME should be HISTPACK. If the file is available, then it will be extended, otherwise a new file will be created. The title and location may be specified by the user, but no other attributes are significant to FAMILYMANAGER.

FAMILYMANAGER has the ability to print reports in either upper or lower case. The Rules Manager determines the default setting with the DEFAULT LOCASE command of the FLEX Inquiry. If an installation has a mixture of lowercase capable and lowercase incapable printers then the TRAINID attribute must be used to ensure lower case files get printed on lowercase capable printers. This is done by label equating a file LYNE.

FAMILYMANAGER commands

Examples

```
DUMP SYSTEM CDIMAGE MB=1000
DUMP SYSTEM SELECT 'DAYS(USEDAY,TODAY) LEQ 7'
BACKUP SYSTEM SELECT EXCEPT ((METALOGIC)=)
CLEANUP USERS3;DUMP PROGRAMMERS
BACKUP ACCTS EXCEPT ( (HISUSER)=, (HERUSER) TEMP/= ) ;
```

The **DUMP**, **BACKUP** and **CLEANUP** commands examine selected subsets of all file entries on the specified family and take actions according to the rules defined for files on that family. Segment limits where they are defined by site management, are also applied by these commands. Both Segment limits and Rules are defined with the aid of the FLEX Inquiry (**RULES** command).

Reports are always generated by the Rule Processing commands. Summary reports for the Rules Manager are always produced. These consist of a page per usercode giving an “at-a-glance” summary of the file usage of the usercode. An overall summary for all files is also produced. Three exception reports may also appear. These reports list all files in the categories:

Files that could NOT be CATALOG ADDED,

Files which were in use

Files of "dangerous" filekinds e.g. NEWPCODE, DCALGOLCODE

If there are no files in these categories, no reports will appear. A final report for the Rules Manager shows a summary by usercode. One column of this report might need explanation - the column marked Kseg-days. This is the sum over all files of the product of the size of the file in 1000 segment units and the number of days since the last access. It is an indication of the “selfishness” of a user — the higher the product, the less the space allocated to the user has been recently used. This product is also placed into the History file if the History feature is enabled.

There is also provision made for reporting the actions taken by FAMILYMANAGER to each user. By default these reports are sent to a printer file, but the user may opt to receive his report in a disk file of filekind DATA under his usercode by email or to suppress the report completely. This option is controlled using the **RULES OP** command using the Flex Inquiry utility. Disk files are named:

FLEXREPORTSFORME/<FAMILYNAME>/<DAYOFWEEK>

For example, a FAMILYMANAGER run for the usercode META and family DBSPACK would produce a file, on Monday, called:

(META) FLEXREPORTSFORME/DBSPACK/MONDAY

Each file “actioned” by FAMILYMANAGER will appear in this report along with a description of the action taken. This description can be tailored to a large degree because whenever a rule is defined three pieces of information are required. They are the Rule type (what to do), the Rule test (when to do it) and the Rule message.

The Rule message identifies a rule and is a small program to build a text from literal strings and possibly the values of attributes of the file being actioned.

File entries which are resident on a pack family may be the subject of the following rule types:

RULE type	Meaning
DESTROY	means remove from mass storage, and ensure no valid backup exists. ZAP is a non preferred synonym
REMOVE	is a REMOVE without reference to the number of backups.
RELEASE	means remove from mass storage, after ensuring that at least one backup exists.
FORGET	means do not make backups of this file
OFFLINE	means mark this file to be first copied to a tape, then forgotten from the catalog.
DONTBACKUP	means do not make backups of this file.
RIP	means release this file last if the segment limit mechanism comes into play.

The only default actions for resident files are BACKUP and provided the FLEX default NOCRUNCH is reset (settable using the DEFAULT command from FLEX Inquiry) then eligible non-crunched files will be crunched. Neither of these actions can specified as rule actions.

Each of these rules require a <boolean expression>, written in the OPAL language, which determines whether the specified rule action applies to the current file. If the expression returns TRUE then the rule is acted upon otherwise the file is ignored. Writing OPAL expressions for rule processing is discussed in **Chapter 4: Rules**.

SELECT modifier

It is not necessary to specify a BACKUP rule, as there is an implied “hard wired” rule built into FAMILYMANAGER which backs up any file with less then two good backups. In OPAL this rule would be written:

```
TRUEBACKUPS NEQ 2
```

With the aid of a SELECT <selection expression> it is possible to backup files which already have two good backups. The <rules expression> is appended to the above condition, which could be written

```
TRUEBACKUPS NEQ 2 OR <boolean expression>
```

A <selection> such as

```
DAYS (USEDAY , TODAY) LEQ 7
```

enables FAMILYMANAGER to perform the equivalent of a Unisys *SYSTEM/ FILECOPY updated COPY, however there is no restriction on the complexity of the

<rules expression> as there is for FILECOPY. If no <rules expression> appears in the FAMILYMANAGERs string parameter it will assume one is to be found in a file. The INTNAME of the file is constructed internally by FAMILYMANAGER and is of the form CARD<first 13 characters of familyname>. The default value of KIND for this file is READER, however it may be label-equated to any convenient KIND, TITLE and location. Most common filekinds are accepted, including JOBSYMBOL, the various compiler symbolic filekinds, DATA, SEQDATA, CDATA and CSEQDATA.

The foreseen use of SELECT was to enable sites to make backups weekly to duplicated tapes of all files accessed during the week, thus allowing one of the tapes to be sent off site.

If the NOCRUNCH option is reset, FAMILYMANAGER will open the file, and immediately close it with crunch. The effects of closing with crunch are that unused space of the file is released to the system, and that the file may not be extended thereafter. Crunching files typically saves much mass storage space when used, but cannot be generally applied to data files. It should be noted that the LASTACCESSDATE of the file will be changed by the WFL job itself if a file is crunched.

Resident file entries which are eligible to be crunched, that is code and symbolic filekinds for which unused segments have been allocated are unconditionally crunched by all Rules Processing Commands, if NOCRUNCH is reset. The filekinds which are crunched by FAMILYMANAGER are discussed in Appendix B.

File entries which are non-resident on a pack family may be the subject of the following rules types:

RULE type	Meaning
DESTROY	DESTROY means ensure no valid backup exists. For non-resident files it is equivalent to FORGET. ZAP is a non preferred synonym.
FORGET	FORGET means CATALOG DELETE a file entry, thus forgetting backup references.
OFFLINE	means mark this entry to be loaded and copied to a new tape, then forgotten from the catalog.
LOAD	LOAD means make resident by COPYING from a backup volume.

There are no default actions for non-resident files. If no rule applies, no action is taken.

Files of KIND=TAPE on family TAPE are only subject to DESTROY and FORGET rules as defined for non-resident pack files.

The following table shows the types of rules enabled by each command.

Rule	DUMP	BACKUP	CLEANUP
DESTROY	YES	YES	YES
REMOVE	YES	YES	YES
FORGET	YES	YES	NO
OFFLINE	YES	YES	NO
DONTBACKUP	YES	YES	NO
RELEASE	YES	YES	YES
RIP	YES	YES	YES
LOAD	YES	NO	NO
BACKUP	YES	YES	NO

All COPY functions are performed by LIBRARY/MAINTENANCE. By default the copies will be COPY&COMPARE&BACKUP. The setting of &COMPARE is given by the DEFAULT NOCOMPARE option and &VERIFY by the DEFAULT VERIFY option.

Both the BIGFILES and HISTORY features are enabled by all Rules Processing Commands.

While examining file entries, FAMILYMANAGER may find file entries where the first backup is unsafe, but the second is satisfactory. In this case it will switch the two backup references for the offending file entry so that the second backup is the unsafe one. This action is taken so that any subsequent backup will cause the file entry to have two good backup references.

Operational Considerations

During FAMILYMANAGER's initialisation phase, the availability of the FLEX RULES file and the setup of the RULES manager usercode and accesscode is verified.

If the RULES file has not been set up or is unavailable:

'RULES file is not valid or does not exist'

If neither the RULES Manager usercode nor its alternate usercode have been setup:

'RULES Manager USERCODE is NOT assigned'

If neither the RULES Manager accesscode nor its alternate accesscode have been setup:

'RULES Manager ACCESSCODE is NOT assigned'

In any of the above circumstances, FAMILYMANAGER will emit the error and then terminate gracefully.

The commands require a cataloging MCP. As a first step in each case a copy of the Volume Library is created in an internal temporary work file. This file is used to generate internal tables and then discarded.

FAMILYMANAGER then simulates the effect of PD -1 for the specified family. This

enables it to determine all usercodes which have file entries on the family. The “basic share” of each usercode is then computed, as explained later. It then initiates two stacks. One is used to obtain information from the *SYSTEMDIRECTORY on the family, the other reads information directly from the *SYSTEM/CATALOG. The information is then merged together as a basis for application of the rules for the run.

The rulings are passed to a further stack, JOBBUILDER, when all file entries have been examined. JOBBUILDER has the task of assembling WFL jobs to implement the rulings for all user files. These jobs will be started whenever sufficient files which need backup to fill one backup tape have been determined. Each job will also contain the logic necessary to remove, crunch and catalog delete file entries. No file will be acted upon in two WFL jobs.

As part of the DUMP and BACKUP phases, the JOBBUILDER WFL jobs will COPY & BACKUP files to tape. FAMILYMANAGER will wait for these jobs to terminate successfully before invoking the next WFL. Note that each job has internal fault logic to detect library maintenance errors and will require operator intervention to confirm the re-copy. Each WFL job will set the AUTOUNLOAD file attribute to ON in the tape file-equation to ensure the automatic unload of the new tapes once the copy has terminated.

This algorithm ensures that FAMILYMANAGER will process the directories on the families as quickly as possible, and that the generated jobs may be acting on files at the same time as other files are being tested against the rules. It also makes the rule evaluation process highly processor bound.

CHECK files

In an ideal world, a system would be run down to a null mix before each run of FAMILYMANAGER so that no file could possibly be modified while it is running. Unfortunately, users of FAMILYMANAGER are increasingly having to run it during normal working hours, or at least make the backups at such times. This leads to the possibility that FAMILYMANAGER could accidentally copy the wrong generation of a file, or even a file in the process of being modified, leading to a version on tape which is unacceptable to the controlling application.

A further common problem is that users sometimes have to do non-MERGE runs of MERGETAPE to get an updated status of the state of the tapes after the FAMILYMANAGER runs.

When FAMILYMANAGER generates a COPY job it now also creates a file called a CHECK file and runs MERGETAPE after the COPY, passing the name of this file by label-equation. The file contains the titles, sizes, and timestamps of the files directed to the tape as well as general information about the tapes. MERGETAPE checks that the files were correctly copied to the backup volume and signals in an appropriate way any exceptions that might have occurred, either by a DISPLAY or in more serious cases by an RSVP. Before finishing it updates the statistics file to reflect the current state. These files may be found under the Rules Usercode on its

default family:

```
(<USERCODE>) FLEX/CHECKFILE/<VOLUME NAME>
```

where<USERCODE> is the Rules Manager usercode and <VOLUME NAME> is the full volume name of the created backup. e.g.

```
(RULES) FLEX/CHECKFILE/DBSPACK97134A
```

Or for a backup to cdimage:

```
(RULES) FLEX/CHECKFILE/CI0116F2019135A/0.
```

These changes allow FAMILYMANAGER to be run during busier working times although Metalogic still does not recommend that FAMILYMANAGER be run during prime times due to the heavy processing load which it often requires.

How Rules are Evaluated

Rules are applied on a usercode by usercode basis. The origin of the rules can be as follows:

The site manager, running under a privileged usercode and an accesscode known to FLEX as the **Rules Manager**, can set up rules for all users on a given family. These have first priority over all other sources of rules.

The Rules Manager may also create rules for a specified user on any given family. These have second priority in evaluation, that is they may not override the general rules for the family.

A Group Leader may define rules for individual members of his group. Where such rules exist, they have third priority in evaluation. A Group Leader may not define rules for users in his groups unless the Rules Manager has permitted him to define rules for himself. The Group Leader may never define rules of type LOAD for users of his group.

Finally if the Rules Manager permits, the user may make rules to manage his own files. These rules will only be evaluated if no site or Group Leader specified rules apply. Thus the user cannot override the site manager or his Group Leader although he can negotiate with the Rules Manager or his Group Leader to modify the global rules on his behalf. The user may not however, define rules of the types DESTROY or LOAD.

Having located the rules from all possible sources, FAMILYMANAGER proceeds to examine each file entry for each file under the usercode directory on the specified family in turn. If FAMILYMANAGER is processing a DUMP command, both resident and non resident file entries are examined, otherwise only the resident files are examined.

For each such eligible file entry, FAMILYMANAGER first evaluates all DESTROY rules. If any apply, the action taken will be DESTROY and all further tests are unnecessary. Note that the overall DESTROY rules for the family are evaluated first, then any DESTROY rules specified by the Rules Manager uniquely for the given usercode. If no DESTROY rules apply, then all REMOVE rules are evaluated.

Should any REMOVE rule apply no further tests are performed. The effect of this ordering is that any DESTROY rule overrides any REMOVE rule and so on.

The complete priority ordering is:

Resident Entries:	Non-resident Entries:
DESTROY	DESTROY
REMOVE	FORGET
RELEASE	OFFLINE
FORGET (=DONTBACKUP)	LOAD
OFFLINE	
DONTBACKUP	
RIP	

For resident file entries, if no rule applies, the Catalog block of the file will be examined to ascertain if this particular generation has two good backup references.

Although there may be two backup references in the catalog block, this does not guarantee that they are both “good”. A backup volume may be scratched without changing the backup references of files pointing to the backup volume. A tape volume may be marked DESTROYED in the Volume Library. The Volume identification might not conform to FAMILYMANAGER naming standards. Any of these reasons are sufficient for FAMILYMANAGER to deem a further backup necessary. If one of the backup references points to a “good” volume and the other does not, FAMILYMANAGER may exchange the backups so that the action of making a new backup will “push” the “bad” backup reference out of the catalog block.

If a <selection> was part of the command, it may cause additional BACKUP rulings to be generated.

The order in which rules are evaluated is designed to give priority to making disk space available. The second priority is to decrease the number of catalog blocks, thus freeing up backup volumes where possible.

Any action decided by interpreting the rules is called a Ruling. When all file entries have been examined all rulings are passed to the JOBBUILDER stack in one batch. This means that all files belonging to one usercode directory are processed together in one or more WFL jobs.

Files which were the subject LOAD rulings are batched together by JOBBUILDER and just prior to going to EOJ, it invokes FLEX Inquiry with a LOAD request. This is done to ensure that the minimum number of tapes are requested by the LOAD operation.

Rulings may also be generated by one more mechanism — segment limit enforcement. This topic is covered in the next paragraph.

Segment Limit Processing

The FLEX package offers the Rules Manager a powerful way of managing space on his families. It works on the concept of dynamic “basic shares” of the space which are computed each time FAMILYMANAGER runs. It also allows one user to “borrow” space from any users not using their “basic share”.

This method is more flexible and caters better to situations where the number of users or the space usage of users fluctuates over time than schemes that used fixed segment limits. Absolute segment limits have their drawbacks. They may lead to space being wastefully reserved if the amount of space occupied by a user fluctuates or the number of users with files resident on a family varies. Families with large numbers of users would tend to absorb more of the site manager’s time than is useful. In some instances, the information needed to set limits sensibly is not the province of the site manager. For this reason FLEX supports a mode of running, in which the limits become “softer” because files are not RELEASEd unless available space on the family falls below a threshold specified by the Rules Manager; and second the Rules Manager becomes less involved in the detail of managing an individual usercodes’ space usage.

The RULES THRESHOLD

To set the threshold, the Rules Manager decides the percentage of a family that he wishes be kept in reserve. This should be large enough so that the normal growth of files, due to file creation, LOAD activity and update activity can be easily accommodated. In addition, it should be realised that as a consequence of the type of algorithms the MCP uses to allocate space, a certain amount of every pack family will be unusable due to checkerboarding. It has been calculated that a realistic figure for this MCP wastage is about 16%. If the METALOGIC program JAMPACK is used to squash the family, this figure can be reduced to between 1 and 5 percent.

If no threshold is specified, FLEX assumes a default of 17%. The Rules Manager may override this default with the FLEX Inquiry RULES THRESHOLD command. For example, to set the threshold to 20% on a family called BATCH:

```
RULES THRESHOLD 20 FAM BATCH
```

Note that since the figure specified is a percentage, FAMILYMANAGER must compute the number of segments at the beginning of each run. This means that the reserve and consequently the space available to allocate to users reflects the instantaneous composition of the family. If one member of a multi-pack family is down or otherwise off line, the limits imposed will reflect the degraded status.

For example, if the family SYSTEM consisted of two 207 packs, each with about 1.1 million segments, and the threshold is set to 20 %, this means that approximately 440,000 segments are reserved, leaving 1,760,000 segments to be shared out.

Basic Shares

FAMILYMANAGER must also compute the space allocation of each usercode with files for each run. To show how this done, suppose that each usercode is entitled to an equal share of segments. If there are N usercodes (including a pseudo usercode SYSTEM-NODE for files in the system directory), the family has S segments and the threshold is T, then each usercode is entitled to $S * ((100-T)/100)/N$ segments. This number is called the “basic share”. Notice that it will change as the number of users with resident files changes. It will also be sensitive to changes in S, the total number of segments on the family. If there are 100 usercodes, on a family SYSTEM with two 207 diskpacks, and the threshold is 20%, the space to be allocated is 2,200,000-440,000 segments; a “basic share” is worth 17,600 segments.

A user can exceed his “basic share”, but his files will not be RELEASEd because of this unless FAMILYMANAGER detects that the available space, including any created by FAMILYMANAGER applying DESTROY, REMOVE, or RELEASE rules, is below the threshold percentage. So a user with more files than his “basic share” borrows some of the unused space in the “basic share” of other usercodes.

Since not all users have equal space requirements, there must be a means of changing the size of a “basic share”. This is done by changing the share number of a user with FLEX Inquiry’s RULES SHARE command. By default, the share number of a user is 1. The Rules Manager, or the Group Leader, can change the Share Number to any value between 0 and 65,535. For example:

```
RULES SHARE 50 USER META FAM SYSTEM
```

It is important to realise that Share Numbers are not percentages. They are used by FAMILYMANAGER in the following way. The Share Numbers of all users with resident files is summed. For example, if all users on BATCH except META have a default value of 1 as Share Number, the computation above is changed as follows. There are N-1 users with share number 1 and 1 with share number 50 so the sum is 50+(N-1). Now one share is worth $S * ((100-T)/100)/(50+(N-1))$. META usercode has a “basic share” of 50 of these shares, all other usercodes have a “basic share” of 1 of these shares. In our example, 1 share is worth $1,760,000/149 = 11,812$ segments, and META has a “basic share” of 602,416 segments.

Group Definition

We now address the second of the simplifying factors of GROUPMODE. If the Rules Manager had to set Share Numbers for all 100 users of SYSTEM, then the amount of effort involved is not significantly less than setting absolute limits. The concept of a group of users can greatly ease this housekeeping burden.

The Rules Manager can define groups of users with the FLEX Inquiry RULES GROUP command. For example, let us define a group on SYSTEM:

```
RULES GROUP USER SHAFT FAMILY SYSTEM OWNER META  
RULES GROUP USER NICOL FAMILY SYSTEM OWNER META  
RULES GROUP USER HURLEY FAMILY SYSTEM OWNER META
```

The Share Number of META remains 50. It is now interpreted as the number to use to compute the “basic share” of the group consisting of the four usercodes META, SHAFT, NICOL, and HURLEY. The size of a basic share is increased too, because three usercodes are no longer owned directly by the Rules Manager and are thus not counted when computing the size of a “basic share”.

By default, each of the 4 usercodes within the group is entitled to one fourth of the group META’s basic share. Should the Group Leader, usercode META, wish to change the proportion, he can use the RULES SHARE command to do so. For example

```
RULES SHARE 5 USER SHAFT FAM SYSTEM
```

or to change his own share of the groups share:

```
RULES SHARE 2 USER META FAM SYSTEM
```

In our example on SYSTEM, 1,760,000 segments may be allocated. The Rules Manager controls 96 of these usercodes directly — all with Share Number 1 except META which has Share Number 50. The worth of a single share is $1,760,000/(50+95) = 12,138$ segments. META as a group is entitled to 606,900 segments.

Within the group, the same algorithm is applied. The sum of the share numbers of usercodes in the META group is 10, so 1 “META share” is worth 60,690 segments. SHAFT usercode has a “basic share” of 5 times 60,690 = 303,450. META usercode gets 2 times 60,690 = 121,380 segments.

Thus the effects of defining the group are twofold. From the point of view of the Rules Manager the housekeeping management problem has decreased. For the META Group, control has been localised. Since Groups can be defined by META, as long as they composed only of members of the META Group, META can simplify its housekeeping by the same methods.

When groups are defined, should a member of a group not use his “basic share”, the space is lent to the group. If any other member of the group exceeds their “basic share”, they borrow first from this pool of extra space. If the group as a whole is over limit, it must borrow space from the next Group Leader up, or if none exists from the Rules Manager.

Advantages of SHARES

The advantages of SHARES are:

- The available space needed to keep the family usable is easily specified and independent of any particular pack configuration. (RULES THRESHOLD).
- The space is allocated to users in proportion to their requirement, rather than as a fixed number of segments. The parameter specifying the space allotted to a usercode, the Share Number, is largely independent of a decrease or increase in the number of users on a family. (RULES SHARE).

- The Rules Manager may define groups to both simplify his maintenance and delegate some of the decision making to the most local level. (RULES GROUP).
- Users are not inconvenienced by the penal nature of absolute limits. If space is available, files will not be made non-resident. For usercodes or pseudo usercodes (SYSTEM-NODE) which should never be released, the Share Number can be set to a large value, and any unused space will be shared between those who need it.

How Limits are applied.

If the THRESHOLD has been exceeded on a FAMILY, FAMILYMANAGER calculates a 'right' in segments for each user. This 'right' is based on the user's share and the capacity of the family. Each user is processed in turn. If, after accounting for RELEASE, REMOVE and DESTROY rules, the user is using more segments than his 'right', this user is marked for processing in a second pass. If a user is using fewer segments than his 'right' then the difference is distributed amongst the other members of his group, thus increasing their 'rights'.

If after processing all users, any user is still exceeding his 'right', a second pass is made for these users. In this second pass each user exceeding his right is processed. Resident files for the user are sorted by the product of size(in kilo segments) and age(in days). Files are then processed in that order. Each file in this list is marked for removal, until the user is within his 'right'. Any file marked for removal has its ruling altered so that it becomes non-resident. Thus a BACKUP ruling becomes a RELEASE ruling and a DONTBACKUP ruling is altered to a REMOVE ruling.

The effect of the preceding algorithm is to remove the least number of "old" files to remain within the limit, and to concentrate on the worst-offending users. Often users would like to modify this algorithm slightly by keeping some files resident regardless of the USEDAYS. To this end, FAMILYMANAGER offers a further rule type RIP, standing for Retain If Possible. If one or more RIP rules are defined, the above algorithm is modified so that any RIP file will be retained if there is any non-RIP file which can be removed.

This algorithm allows users a degree of say in which files should remain resident without allowing the segment limits to be overridden. Should a user define a rule to mark all his files RIP, the effect will be exactly the same as if he had marked none of them RIP.

One important point about RIP rules is their interaction with other rule types. DESTROY, REMOVE, and RELEASE rules already decrease the number of resident segments and this is taken into account when computing how much needs to be removed. Files whose ruling is DONTBACKUP are automatically marked RIP by FAMILYMANAGER, thus ensuring they will remain resident as long as possible. To ensure that space is not wasted keeping such files resident it is advisable to

define a REMOVE rule corresponding to every DONTBACKUP rule. If not, the long term result can be an accumulation of resident files which have no backup.

Assignment Of Serial Numbers

If a serial number list is assigned by the SERIALNO attribute to the dummy file describing attributes of the backup volume, these serial numbers will be used in any generated jobs. Serial numbers are assigned to each tape as they are required, and the order of allocation is “A-TAPE”, “A-TAPE” BACKUP, “B-TAPE”, “B-TAPE” BACKUP, where the names correspond to the names generated by FAMILYMANAGER from combining the first 11 characters of the familyname with the Julian date, and appending the letters of the (English) alphabet on the end. Examples of these tape names would be:

```
SYSTEM97032A
SYSTEM97032B
SYSTEM97032C
```

If a duplicate backup is requested, the name will consist of the first 10 characters of the familyname, followed by the Julian date, followed by an “F” for the first backup or a “G” for the second backup, followed by ascending letters of the alphabet. Examples of these tape names are:

```
SYSTEM97032FA
SYSTEM97032GA
SYSTEM97032FB
SYSTEM97032GB
```

If a serial number is missing from the list, due to a ‘hole’ or because the list was not big enough to handle all reels required, no serial number will be requested in the generated copies. If the serial number list contains more than the number of tapes used, they are ignored.

Rules Debugging Techniques

Although the technique of defining rules and segment limits is extremely flexible, it can be a complex process to define and debug a proposed set of rules.

FAMILYMANAGER allows the Rules Manager to debug a proposed set of rules in several ways.

Setting the taskvalue of FAMILYMANAGER to 1 or using the TEST command during a run places FAMILYMANAGER into debug mode. When processing a rules processing command in debug mode, FAMILYMANAGER does not initiate any WFL jobs. The jobs are built in the normal way, but instead of being started and then removed, the files are locked under the usercode directory of the Rules Manager. They will appear in the directory JOB. All DISKREPORTS for individual users will appear under the Rules Manager’s usercode in the directory FLEXDEBUGREPORTS.

By setting further bits in the taskvalue, rule types can be disabled. For example bit 1 corresponds to the DESTROY Rule type. If both bit 1 and bit 0 are set then all DESTROY rules will be evaluated, but the rulings generated by DESTROY rules will not lead to any actions on files. The bit numbers are as follows.

Bit Number:	Operation disabled if set:
0	Places FAMILYMANAGER in debug mode
1	DESTROY
2	REMOVE
3	RELEASE
4	DONTBACKUP
5	BACKUP
6	RIP
7	FORGET
8	LOAD
9	OFFLINE

Debug mode is useful when rules for all files on the family are to be debugged, but are less so when user specific rules are being defined. In this case, FLEX Inquiry is the easiest debug method. The Rules Manager can always simulate the effect of rules for one user on a family with the RULES TEST command. FLEX initiates FAMILYMANAGER as a private library and evaluates all rules for the specified usercode only. No jobs are produced by a RULES TEST — the rulings are directed to the initiating terminal and are reported within the user detail report in the normal way.

It is suggested that the rule definition process be taken slowly. Remember that the only actions taken by default are BACKUP and possibly CRUNCH so it is safe to run FAMILYMANAGER with a minimal set of rules. The further refinement of rules should be an iterative process.

Some rules are almost universal. They are typically the “policeman” rules which determine what filekinds are permitted to normal usercodes. Some examples of these may be found in the chapter on RULES.

COPYWRITE and FLEX integration

FLEX/LIBRARY can handle COPYWRITE CD Images as media for FLEX3.It supports TAPEFEET on CDImages, where 1 ft = 1 MB.

An entry point ML_GET_CD_SERIALNO allows the caller to get the next SERIALNO for a CD or CDImage.

A file may be copied by specifying a file number. The file number is specified by a # followed by an integer, and must be enclosed in single quotes. For example,

`COPY '#1' FROM TEST(CD,HOSTNAME=LOCAL,LOCKEDFILE)`

This syntax may be used for copying files from TAPE or CD, as well as CD Images and Wrapped Containers.

A LIBMAINTDIRECTORY is created if the LIBMAINTDIR attribute is set and a SERIALNO is specified.

By default the file is called LIBMAINTDIR/<volumename>/<date>/<serial> under the Usercode of the Task performing the Copy, and it is stored on the DL LIBMAINTDIR Family. The title may be changed by Label Equating FILE LIBMAINTDIR on the Copy Statement.

The LIBMAINTDIR utility command has been added. Utility commands may be executed under CANDE by

`U *METALOGIC/COPYWRITE <command>`

The LIBMAINTDIR command syntax is,

`LIBMAINTDIR <title> HOSTNAME`

will display the HostName stored in the LIBMAINTDIR given by the <title>.

`LIBMAINTDIR <title> HOSTNAME <hostname>`

will set the HostName in the LIBMAINTDIR given by <title> to the <hostname> specified.

The entrypoints LibMaintDirFileName, FindLibMaintDirHostName and ChangeLibMaintDirHostName are exported from CopyWrite.

When doing a COPY&BACKUP or COPY&CATALOG, if the Volume with the SERIALNO is already in the Volume Library, but is marked as SCRATCH, or has the FamilyName "S C R A T C H", CopyWrite will Volume Delete the Volume and use it for the Copy.

`VLUPDATE <libmaintdir title>` command in CopyWrite Utility.

If a Volume is deleted from the Volume Library, and then added again, the FAMILY CREATION DATE is set to UNKNOWN. An attempt to load a file using a backup reference which refers to this volume will fail because the FAMILY CREATION DATE is used to build the title of the LIBMAINTDIR file. The LIBMAINTDIR file stores the name of the Disk Farm Host on which the volume resides. This command opens the LIBMAINTDIR file, and uses the BOTTIMESTAMP as the CREATION DATE to update the Volume Library entry.

When copying files from a CD Image, if the <source volume> is not resident and a SERIALNO is specified, CopyWrite uses the SERIALNO to find the <familyname> in the Volume Library, and then searches for an image called <source family>/<familyname>. This matches the FLEX FamilyManager naming conventions. The search which was previously done using the current path, is now done on the family specified by the FLEX_CWIMAGE Configuration Variable. This may be overridden

by specifying the FAMILYOWNER as another FamilyName

Example:

```
Copy F from DEV(CD,SERIALNO="CI1234",LockedFile,HostName=LOCAL)
Copy F from DEV(CD,SERIALNO="CI1234",LockedFile,HostName=LOCAL,
                FamilyOwner = DISK)
```

The search is done using the FamilyName specified without Family Substitution.

If there is a mismatch between the Version of CopyWrite and the Version of ExtractArchive a warning message is displayed.

A COPY *= FROM NULL TO ...Disk Farm releases all files in the Disk Farm and deletes the Disk Farm directory from the NT File System.

A COPY <file list> FROM NULL TO ... Disk Farm replaces each selected file capsule with a Released Capsule, and the space is returned 79885000 to the NT File System.

The COPY&GO between A-Series and between A-Series and Windows waits for the Job or CodeFile to complete, before terminating. Any messages generated are displayed.

It is possible to initiate a third party copy between Disk Farms on an NT system.

The general syntax is:

```
COPY <file list> FROM <source>(PACK,HOSTNAME=<hostname>) ,
      <file list> FROM <source>(PACK,HOSTNAME=<hostname>)
      TO <destination>(PACK,HOSTNAME=<hostname>)
```

The <hostname> for the <source> and <destination> must be the same.

This capability is used by MergeTape to merge Disk Farms.

The &CATALOG modifier may be used to add an Archive Reference.

The &BACKUP modifier may be used to add a Catalog Reference.

The LIBMAINTDIR may be used to create a LibMaintDir. If the protocol version of the CopyWrites performing a file transfer between A-Series Hosts are incompatible, an AX is required to continue the Copy.

If a version mismatch occurs between an A-Series and an NT System, the Copy is terminated.

If a FAMILYOWNER of DISK is specified when doing a COPY&CATALOG, a family called DISK is assumed, and No Family Substitution occurs

After a Volume is added to the Volume Library by CopyWrite it checks if a DUPLICATE FAMILY condition could occur, and if so it terminates the Copy and deletes the Volume.

An attempt to Copy a file which has been released from a Disk Farm displays information about the released file, and skips the file

If the File CDIMAGEHARDLINK is Label Equated with a FAMILYNAME when creating a CD Image, a reference to the CD Image is created on that FAMILYNAME

This is used by FAMILYMANAGER to allow CD Images to be located on families other than that specified by FLEX_CWIMAGE.

After sending a TapeLabel Notice to Supervisor, CopyWrite Logs a Volume Directory Add Record to the System Sumlog.

If a Copy to NULL is performed, and LIBMAINTDIR is specified, a SERIALNO must be assigned to the <destination volume>.

If the SERIALNO is found in the Volume Library, the LIBMAINTDIR is created using the Volume Name, SerialNo and Creation Date of the Volume Library entry. If not found, a message is displayed and the LIBMAINTDIR is created for a volume called NULL.

If a LIBMAINTDIR is being created by copying to NULL, then the Data for all files is copied from the <source volume>, and discarded, thus checking the integrity of the <source volume>. When copying from a <source volume> which is a Disk Farm at a Remote Host, such a Copy could cause significant network traffic, just to recreate a LIBMAINTDIR.

If the SCRATCHPOOL = NODATATRANSFER is specified on the <destination volume>, then only the Disk File Headers are transferred, and the Data is skipped.

If an ADD statement specifies LIBMAINTDIR or LIBMAINTAPPEND=TOEND then a LIBMAINTDIRECTIONS file must be present.

A COPY statement with LIBMAINTDIR specified creates a new LIBMAINTDIRECTIONS.

If LIBMAINTDIR is specified and a File is released using the COPY from NULL syntax, the Released Flag ([33:1] in LMDFStatX) is set for the File Entry.

If LIBMAINTDIR is specified and an ADD&COMPARE replaces some files, the corresponding entry in the LIBMAINTDIRECTIONS is modified to match the new file

If LIBMAINTAPPEND=TOEND is specified, the files added are entered into the LIBMAINTDIRECTIONS.

A LIBMAINTDIRECTIONS can be recreated from a Disk Farm using a COPY statement like this:

```
COPY *= FROM <volume name>(PACK,HOSTNAME=<hostname>,  
    SERIALNO=<serialno>) TO  
NULL(CD,LIBMAINTDIR,SERIALNO=<serialno>,  
SCRATCHPOOL=NODATATRANSFER) ;
```

If the <serialno> exists in the Volume Library and has a valid Creation Date, the LIBMAINTDIR title is constructed using the Volume attributes

If a LIBMAINTDIRECTIONS is recreated from a Disk Farm which has 7 released files, the File Entry is marked as Released and the File Attributes are set from the

Released File Information stored in the Disk Farm

A LIBMAINTDIRECTORY can be analysed using the File Analysis utility which comes with CopyWrite for NT.

A LIBMAINTDIRECTORY can be recreated from a Disk Farm using a COPY statement like this

```
COPY *= FROM <volume name> (PACK, HOSTNAME=<hostname>,
                               SERIALNO=<serialno>) TO
NULL (CD, LIBMAINTDIR, SERIALNO=<serialno>,
      SCRATCHPOOL=NODATATRANSFER, FAMILYOWNER=<hostname>)
```

The FAMILYOWNER attribute may be used to specify the HostName which is stored into the recreated LIBMAINTDIRECTORY. This use of FAMILYOWNER is incompatible with the &CATALOG option which uses the FAMILYOWNER as the Catalog FamilyName.

MERGETAPE

Originally designed to merge backup tapes but now can merge all backup volumes.

After FLEX FAMILYMANAGER has made a backup volume, all the files on that volume are in the system catalog. However as time passes, many of the references to those files will be deleted, and, in effect, the space taken by those files on the backup volume becomes wasted. If nothing is done to prevent it, the space wasted will grow sharply over time and many more backup volumes will be needed than would be the case if they were fully occupied. This can be avoided by gathering the still useful files from relatively unoccupied backup volumes on to a smaller number of densely populated ones. Thus, the number of backup volumes is minimised and the under-utilised volumes can be released for further use.

Within the FLEX package this consolidation process is called "Merging"., and it is the primary function of MERGETAPE. It is supported by two commands. "REPORT" gives an indication of the effectiveness of different merges and a summary of the how effectively the tapes are being used, and MERGE which actually performs a merge.

Over time a backup volume can lose all of its backup references: as files are CATALOG PURGED, either manually by a user or through a RULE in FAMILYMANAGER; or as new generations of a file push older ones over the generation limit; or more than two backups of a file entry are made. Also, MERGETAPE can force this process to happen by merging.

MERGETAPE second major function is to mark these tapes available for re-use, which is called "Releasing". Released backup volumes can be seen in the MERGETAPE report entitled "*Volumes no longer in use as Backup*", in FLEX Inquiry in the LISTVL and PV reports, and, if available, in the TAPECONTROL database. Releasing is done by the explicit "RELEASE" command, or implicitly at the end of a merge.

MERGETAPE has been designed to run with minimal operator intervention and conducting a merge should not require anything other than very basic operational skills. For example, . a hard copy report of all input and output messages is kept by the program to enable post-mortems, and dynamic recovery from situations like missing tape volumes, or missing damaged, or incorrect versions of loaded files, is built in.

MERGETAPE is also the primary component of a small tape library system, built in to Cataloging and FLEX, which allows FLEX users to manage their tape library without recourse to tape library systems like TRIM unless they have quite complex tape management requirements. Like all tape management systems, FLEX needs a database of tape information. These data are kept in the system Catalog file and a file called the Statistics file maintained by MERGETAPE. OPAL has access to the data in the Volume Library through the VL attributes and to the Statistics file data via the VS attributes. See HELP ATT VS= or HELP ATT VL= for more details. FLEX Inquiry will show all the information about a volume using the PV command, which is

also used for manual update of the Statistics file data.

Running MERGETAPE

MERGETAPE comes on the FLEX release as the code file *METALOGIC/FLEX/MERGETAPE, although its name can be changed to any name desired. It must be run under a privileged usercode (PU). There is a single String parameter and MERGETAPE can be simply run from CANDE (R or U commands), MARC, or WFL. When the MERGETAPE run does not involve Merging, no interaction is involved. When Merging the questions are normally sent to a remote terminal, but can also be sent straight to the operator via ACCEPTs.

Examples

The following job releases any backup volumes made available since the last run of MERGETAPE

```
?BEGIN JOB;  
USER=PRIVI/PASS;  
RUN *METALOGIC/FLEX/MERGETAPE ("RELEASE") ;  
?END JOB.
```

The following job runs a merge with interactive questions sent to the station which started the job.

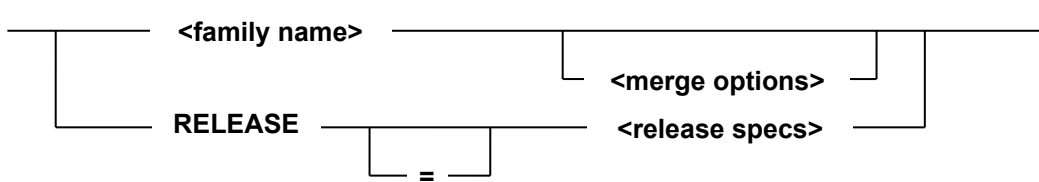
```
?BEGIN JOB CATALOG/MERGETAPE/SYSTEM/RUN;  
OPTIONS=(*,BDBASE);BDNAME=LISTING/MERGETAPE;  
JOBSUMMARY=UNCONDITIONAL;  
STRING S;  
S:= "MERGE PACK CDIMAGE TIMEOUTS WORKPACK=AUDITPK,LOADS=10";  
DISPLAY(S);  
RUN *METALOGIC/FLEX/MERGETAPE ON BATCH(S);  
    STATION=MYSELF(SOURCESTATION);  
?END JOB.
```

Note` that the use of JOBSUMMARY and BDNAME task attributes ensures that the reports are not lost. Setting the STATION task attribute for MERGETAPE tells the program where to open its remote file, which is needed for the conversation with the operators during a merge. Setting STATION this way means the operator may log onto any desired terminal, start the job, and interaction will take place from that terminal.

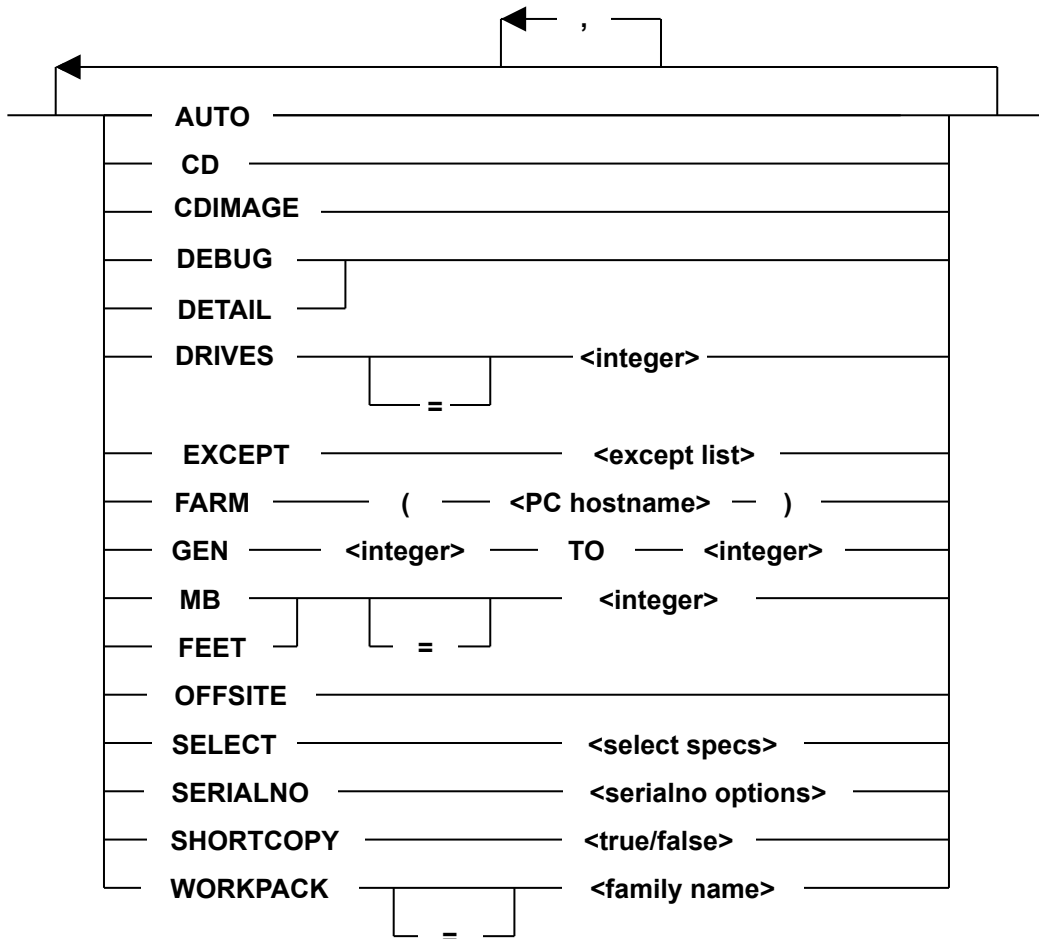
An example of a more complicated job to handle all aspects of the merging process is supplied on the tape as *EXAMPLES/FLEX/WFL/MERGETAPE. This job has the advantage that the operators do not need to know a privileged usercode to run the merge. The job can be started once from a privileged usercode, and it will wait in a queue to be FS'ed by the operator. Once it is running, it starts another copy of itself that will wait in the queue for the next merge that needs to be run. The operator will be prompted through ACCEPTs for the <familyname> to be merged or analysed,

and the LSN of the remote station that will be used to run the merge.

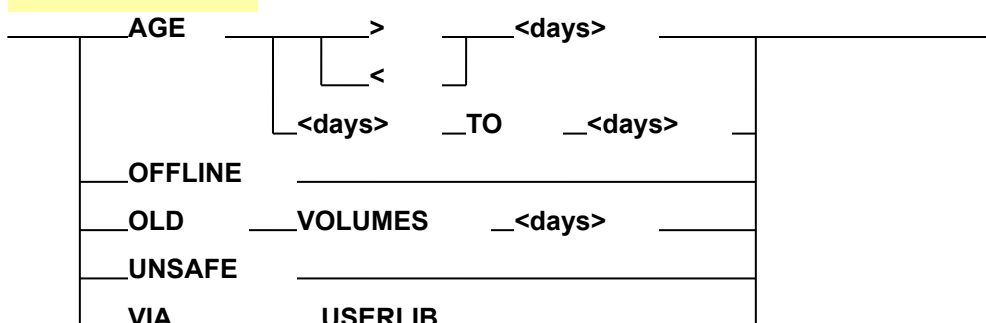
Syntax



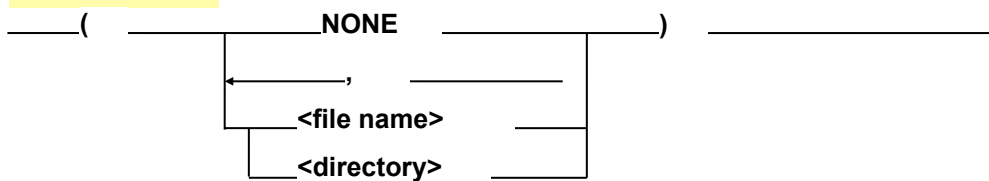
```
<merge options>
```



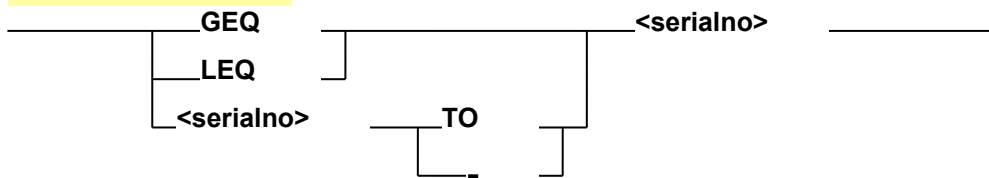
```
<select specs>
```



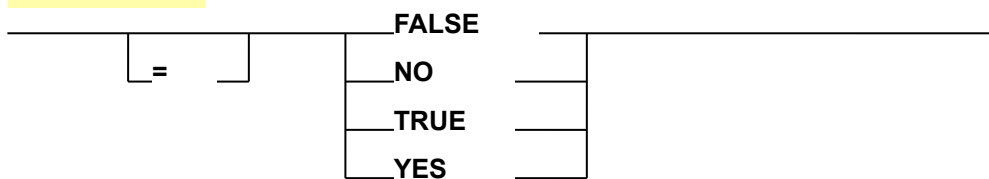
<except list>



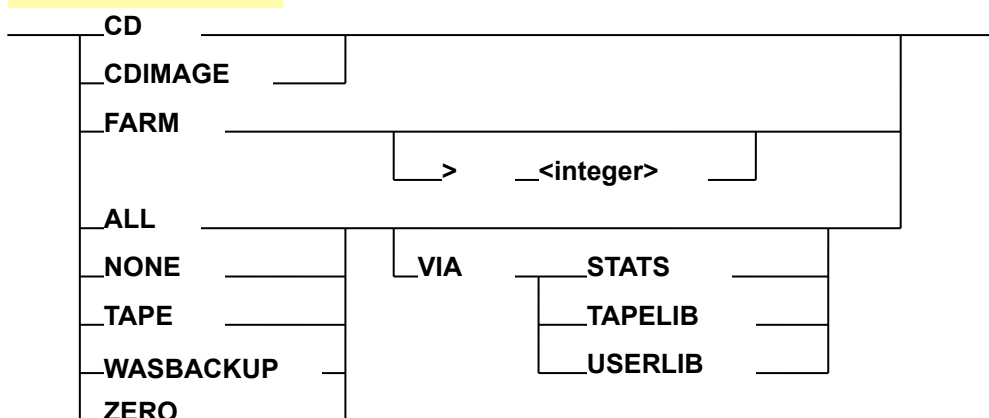
<serialno options>



<true/false>



<release specs>



Due to a limitation of the WFL compiler, it would be possible for MERGETAPE to cause a WFL FAULT if the size of the generated WFL COPY job was greater than 65535 words. Although this size of WFL job is very unusual, but because of this restriction, MERGETAPE will automatically split WFL jobs if this limit is reached. The dialog file will show the reason as 'CLOSED DUE TO ARRAY'.

All MERGETAPE runs must be run under the Rules Manager usercode. It is not necessary to run MERGETAPE with the Rules ACCESSCODE.

Tape Merges

A work pack is required when merging tape volumes.

Previously Mergetape would fix up all backup references and tidy up the work pack after the final copy was complete. This could mean that if Mergetape failed after the first pair of tapes had been created then much of the benefit of those

copies could have been lost. Backup references are now updated after each pair of tapes, and the relevant files removed from the workpack at that time.

A new question is now asked, after the question for the name of the Operator. The new question asks

'Do you wish full details of the merge to be reported?'

If Mergetape is run from a WFL Job, any syntax errors in copies will create a summary file. An example Job to run Mergetape for a family specified as a parameter, with defaults of Merge as the workpack, 2 tape drives and full details, is listed below.

```
BEGIN JOB MERGE (STRING FAM) ;  
  
FAMILY DISK=DISK ONLY ;  
  
RUN (META)METALOGIC/FLEX/MERGETAPE ON DEV(FAM &  
"WORKPACK=MERGE,DRIVES=2,DETAIL=YES") ;  
  
STATION=MYJOB (SOURCESTATION) ;  
  
END JOB
```

Entering START MERGE("DEV") from a cande station would run a merge for family DEV with all questions being asked to the Cande station.

Mergetape is run in the job generated by Familymanager. It checks that files were backup up correctly and update the statistics file.

A problem may occur if any files have changed between the time of the Familymanager run and the time of the copy to tape and was then removed before the Mergetape run. Mergetape will display an Accept message saying that some files had bad references and will DS if no accept is entered in 45 seconds. The DS is to draw attention to the fact there were suspect files on the tape. Such files will be flagged with the error " was modified after FMGR run and removed".

The Question ..Do you wish full details of the merge to be reported? is intended to display more details of files and tapes being merged. If YES is answered this extra information is sent to the Screen or ODTs and written to the printer file Dialogue. If NO is answered it is written only to Dialogue. This information can be useful in diagnosing bugs in Mergetape and in establishing why fewer backup volumes are released than expected.

After entering the number of volumes to be merged, a list of the serial numbers of the volumes request is displayed.

If the CD image COPY&CATALOG.. TO NULL procedure (used to update catalog backup references after a merged CD image has been created) generated any file errors., MERGETAPE will display an appropriate error message and continue; this enables unattended AUTO runs for image merges to complete regardless of errors. MERGETAPE will CATALOG PURGE any expired LIBMAINDIR for Disk Farm volumes that have been released during a merge run. MERGETAPE will also attempt to get the PC hostname for a Disk Farm volume from TRIM if the

associated LIBMAINTDIR is unavailable or does not have the host name present.

MERGETAPE will automatically partition copies into multiple volumes according to WFL library maintenance limits or according to the default length assigned to the backup medium used. After the copies are complete, MERGETAPE will automatically update the backup information of the file copied during the merge.

When merging tapes, if any of these are found to be subsequently non-resident during the alter phase, MERGETAPE will generate a job to copy the files from the work family; these are called "phantom" files.

<merge specs>

MERGETAPE accepts additional keywords to allow unattended operation of any merge. The keywords are documented as follows:

CD

Find eligible CD volumes only; both CD and CD images will be used for merging. A workpack is required.

CDIMAGE

Finds eligible CD images only; only CD images will be merged. No workpack is required.

If neither CD or CDIMAGE is specified, only tape volumes will be merged.

DETAIL

If YES, provides additional file and volume information in MERGETAPE's diagnostics printout.

DRIVES

Specifies the number of tape/CD drives available for loads and merges. Not applicable for CD image.

LOADS

Specifies the number of CD/CD images/Tapes to be merged for this family.

FEET/MB

For tapes, this value specifies the maximum 'Tape feet' currently in use for a volume, to be used to select eligible tapes for merging. For CD and CD image merging, this value is referred to in MegaBytes. Max value=255.

SHORTCOPY

If YES, then MERGETAPE will automatically proceed with merge jobs that create small tape/CD volumes. MERGETAPE will consider a tape as 'short' if only 20% of the volume will be used. Not relevant for CD images.

WORKPACK

For tape and CD merges, WORKPACK allows user selection of a disk family to be

used for loading files for merging. Not relevant for CD images.

AUTO

Tells MERGETAPE that this will be an automated batch run and various options will be automatically assigned unless overridden by other keywords. If used, AUTO should be the first keyword after the family to be merged.

AUTO Defaults

DRIVES

set to 1

NOSHORTCOPY

set to FALSE

DETAIL

set to FALSE

LOADS

set to 10 volumes

LIMIT

set to 25 MB/tape feet

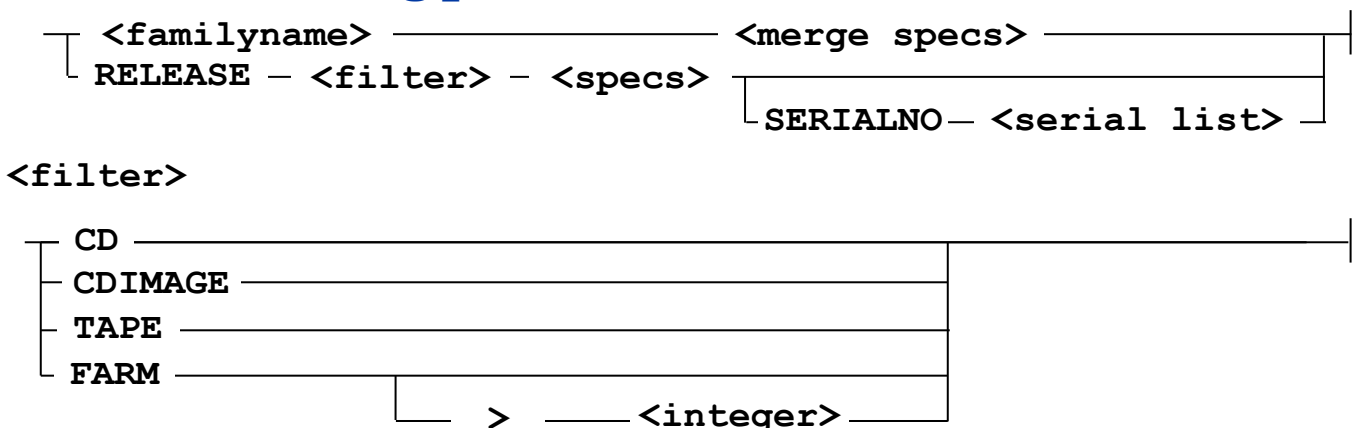
CDImage Example

In the case of CD image merging, it is simple to set up a run of MERGETAPE to merge images without operator intervention:

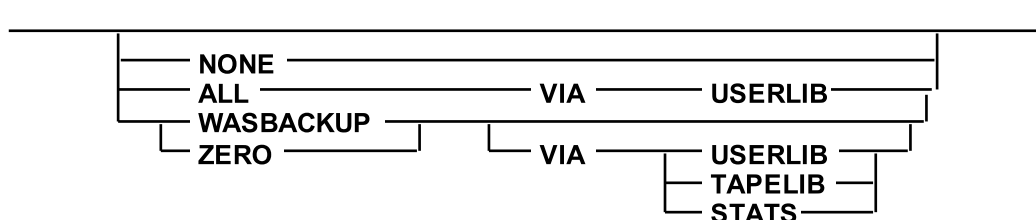
```
RUN *METALOGIC/FLEX/MERGETAPE("DEV AUTO CDIMAGE MB=20")
```

Where MB and DRIVES are both set, MERGETAPE will use the setting which releases the minimum volumes.

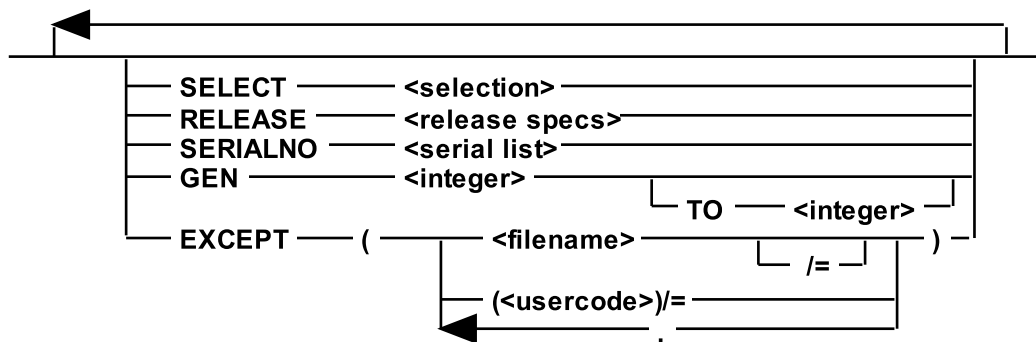
Syntax of the string parameter



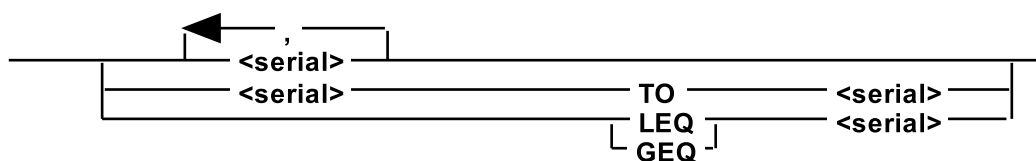
< specs >



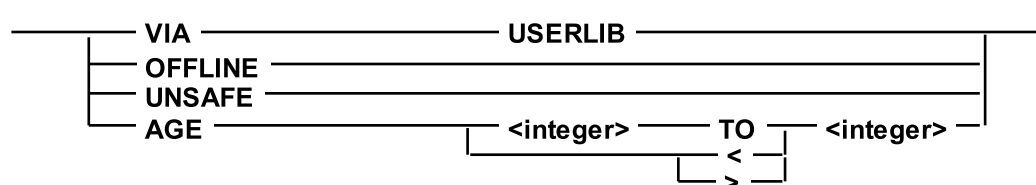
<selection/release>



<serial list>



<selection>



<merge specs>

The keywords are documented as follows:

CD

Find eligible CD volumes only; both CD and CD images will be used for merging. A workpack is required.

CDIMAGE

Finds eligible CD images only; only CD images will be merged. No workpack is required.

If neither CD or CDIMAGE is specified, only tape volumes will be merged.

DETAIL

If YES, provides additional file and volume information in MERGETAPE's diagnostics printout.

DRIVES

Specifies the number of tape/CD drives available for loads and merges. Not applicable for CD image.

LOADS

Specifies the number of CD/CD images/Tapes to be merged for this family.

FEET/MB

For tapes, this value specifies the maximum 'Tape feet' currently in use for a volume, to be used to select eligible tapes for merging. For CD and CD image merging, this value is referred to in MegaBytes (1000000 bytes). Max value=255.

SHORTCOPY

If YES, then MERGETAPE will automatically proceed with merge jobs that create small tape/CD volumes. MERGETAPE will consider a tape as 'short' if only 20% of the volume will be used. Not relevant for CD images.

WORKPACK

For tape and CD merges, WORKPACK allows user selection of a disk family to be used for loading files for merging.

AUTO

Tells MERGETAPE that this will be an automated batch run and various options will be automatically assigned unless overridden by other keywords. If used, AUTO should be the first keyword after the family to be merged.

```
ERGETAPE("DEV AUTO CDIMAGE MB=20")
```

Where MB and DRIVES are both set, MERGETAPE will use that setting which releases the minimum volumes.

RELEASE Command

RELEASE causes volumes to be marked as released for later re-use. RELEASE is implicit in merges and because a RELEASE clause is part of <selection/release> the desired release behaviour can be added on to any MERGETAPE run.

e specified <familyname>. Merges normally need interactive input from the operator.

<filter>

This permits MERGETAPE to release volumes of the appropriate type that have zero references and are marked as RELEASED or WASBACKUP. The RELEASE FARM variant will only operate if a threshold value is given.

< specs >

NONE

means that no volumes will be released.

WASBACKUP

releases volumes that are known to have had backups previously

ZERO

means release volumes that have zero backup references, EVEN IF they have NEVER had backup references before.

BEWARE! Indiscriminate use of RELEASE ZERO can lead to MERGETAPE releasing user tapes that are made by COPY rather than COPY&BACKUP. RELEASE ZERO can be useful in conjunction with limited SELECT criteria or when TAPECONTROL has the final say on a tapes destiny.

VIA

This clause indicates what action MERGETAPE is to take to release a volume.

STATS

says that the only action is to mention the tape in the printed reports and set the RELEASED bit in the statistics file.

TAPELIB

says to effect a VIA STATS release plus to call TAPECONTROL via the OPALTAPELIB Library to update the database PGOK flag and NOTES field. The NEW released flag will not appear in the reports if the tape was already PGOK in the database. The tape will also not be released if its PGNEVER flag is on. This allows users of TAPECONTROL and FLEX/3 to benefit from their mutual communication. TAPELIB defaults to STATS if DEFAULT+TAPECONTROL has not been input to

Default

If just RELEASE is specified without qualification, the method employed is: RELEASE WASBACKUP VIA TAPELIB or STATS depending on TAPECONTROL availability.

USERLIB

allows the installation to more precisely control the tapes to be released. Volumes are passed to a procedure called USERRELEASE, which like USERSELECT is in the FLEXMERGELIB Library. The tapes passed are given by the tapes which would normally be released by the WASBACKUP or ZERO keyword. Thus RELEASE WASBACKUP VIA USERLIB will only pass the tapes with zero backups that MERGETAPE knows used to have backup refs. As already mentioned above, care must be taken when using the RELEASE ZERO VIA USERLIB construct. The result returned from the USERRELEASE tells MERGETAPE what the VIA should be for the release, or if the tape should not be released at all. VIA USERLIB is intended to be used by sites wishing to link MERGETAPE release action to tape library systems other than TAPECONTROL. It may also be useful to implement more stringent local checking.

ALL VIA USERLIB Allows the site to use MERGETAPE to pass all the tapes to USERRELEASE. However, the procedure will cause MERGETAPE to fault if it tries to release a tape which has backup references. A Boolean parameter gives the information as to whether the tape can be released. ALL VIA USERLIB gives a simple way for a site to write a procedure to make reports on every tape in the Volume Library. The specification of USERRELEASE can be found in the source code of the USERLIB example on the FLEX release tape (FLEX/MERGETAPE/USERLIB).

If the SERIALNO clause is present, only tapes specified in the <serial list> are candidates to be released. The <serial list> clause is described later.

<selection/release>

RELEASE

adds the RELEASE command specified after the usual REPORTMERGE processing. The release phase is always quite separate from any other processing. If no RELEASE clause is present in a REPORT command, no release pass is done and the statistics file will not be updated. If omitted for a MERGE command, a RELEASE WASBACKUP is assumed.

GEN

restricts the merge to specified generations (minimum 0, maximum 6). This can be useful in merges if specialised recovery tapes are needed, for example a set of

volumes for off site storage containing all generation 0 files.

EXCEPT

allows particular files or directories to be excluded from a merge or report, which also implies the exclusion of the tapes on which they reside. This clause is normally only used when the files are only to be excluded for one run. For more permanent exclusions a file called the EXCLUDEFILE is used. This allows one file or directory per card in a TEXT file format CANDE flat file. MERGETAPE itself will sometimes add files to the EXCLUDEFILE when it finds a file which is more than half the length of the tape on which it resides.

SERIALNO Some sites use a specific range of serial numbers for FLEX tapes and do not wish FLEX to take actions outside those ranges. Other sites wish to merge certain arbitrary sets of tapes at certain times. The SERIALNO clause is used to specify the serial numbers to be processed.

<serial list>

The GEQ, LEQ, and TO forms allow broad ranges of tape serial numbers to be specified. For example to restrict MERGETAPE to all tapes with serial numbers beginning with G0,

```
SERIALNO G00000 TO G09999
```

<serialno> can be single or double quoted i.e. 'G00000' or "G00000".

If the tape list form is used, a maximum of 511 serial numbers can be specified. Note that if a simple tape list is given, the list will override any excludes, either in the EXCLUDEFILE or in the EXCEPT clause.

The SERIALNO clause must appear as the last clause in the input parameter. If no SERIALNO clause is given, the range used, as before, is

```
"A      " TO 999999
```

Example:

```
U *METALOGIC/FLEX/MERGETAPE MERGE X RELEASE ZERO SERIAL C TO  
C99999
```

SELECT

SELECT Specifies a <selection> to modify the method used to select tapes for a merge

<selection>

UNSAFE

Volumes are unsafe if they are scratch, marked VOLUME DESTROYED in the Volume Library, or known to have bad references. Using UNSAFE as a selection will place those tapes at the highest priority for merging. This is feasible because there are normally other tapes from which the affected files can be loaded.

OFFLINE

One of the FAMILYMANAGER Rule Types is called OFFLINE. It has the effect of placing a file title in a file called the OFFLINE file, along with its TIMESTAMP and LASTACCESS. When an OFFLINE merge is performed, MERGETAPE reads the OFFLINE file and merges just those files. When the merge is completed, the files are deleted from the Catalog. This allows little used files to be placed on special tapes, which can be kept for long periods in special storage, without having to keep references to these files in the system Catalog.

AGE

selects tapes created within a specified age range. The <integer> is the number of days in the past. Apart for the selection by age the merge is otherwise the same and the tapes are prioritised as usual by their estimated tape utilisation. However, exceptionally, the AGE > form prioritises by age first. Thus older tapes are always selected before younger ones within the specified range. If the effect of an AGE > is desired without the change in prioritisation, it can be simulated by changing AGE > x to AGE x TO 5000.

VIA USERLIB

Allows the site to take complete control of the selection of tapes for a merge. When this option is used, all the tapes in Volume Library with backup references will be passed to a procedure USERSELECT in the FLEXMERGELIB library. The return value of USERSELECT determines how the tape will be treated. The specification of USERSELECT can be found in the source code of the USERLIB example on the FLEX release tape (FLEX/MERGETAPE/USERLIB).

MERGE 'Command'

Although there is no implicit merge command any string not starting with Release will perform a merge. Note that the <selection/release> documented above is also valid in the MERGE and contains most of the options for merging. The remaining ones in <merge specs> concern the practical aspects of a merge.

OFFSITE

Many installations use off site storage for their tapes to allow disaster backup in case of physical destruction of the tape library. As a result, FLEX can keep track of a tapes on or off line status in the statistics file. A tape can be marked OFFSITE by using the FLEX Inquiry PV command. The OFFSITE clause in MERGETAPE specifies that the second backup tape (the first mounted) is to be marked OFFSITE at the end of the merge.

TIMEOUTS and Interactive Merge

When merging, MERGETAPE is interactive. It can be run using DISPLAYs and ACCEPTs at the console or via a datacom file. Most of the questions are routine and many sites always answer them in the same way. If the TIMEOUTS option is not specified, all questions will cause MERGETAPE to wait until it gets a response. When the TIMEOUTS option is specified, most questions wait for a short time (typically 30 seconds), then the merge proceeds with a sensible default value.

Workpack name? = WORKPACK

The Workpack is the family used to COPY files to, when they are not resident on the original family. It should not normally be the original family because that reduces the efficiency of the merge. In general it should be a family that is unlikely to have any file title conflicts with the original pack, and one that has a considerable number of sectors available.

How many tapes shall we merge? = LOADS

The number of tapes to mount for loading files.

Number of tape drives available to MERGETAPE? = DRIVES

The number of tape drives available. This answer governs the number of simultaneous LOADs that can be fired off. If the number of drives is 1, MERGETAPE will not issue a duplicate destination COPY to make the new tapes, but fire off two sequential single destination COPYs.

Do you wish to go ahead with this COPY? = SHORTCOPY

If the output tape to be made will be less than 20% full, this question appears. The default answer is YES so the tape is made. However to avoid the wastage, say NO, and re-run the Merge with a larger number of LOADS.

All questions asked by MERGETAPE of the operator will prompt for the type of response required. For example², here is some of the dialogue taken from an actual run :

² See the sample merge at the end of the section

```
12:43 MERGETAPE Version 39.390.77, 12:43 Friday January 22,1993
12:43 Parameter: MERGE SYSTEM SHORTCOPY)
12:43 ..Shall we use XSYSTEM as the workpack ? (Enter YES or NO)
YES
12:43 ..Begin scanning SYSTEM
12:44 ..6806 volumes found in the Volume Library
..Now at generation 0 of *D/DOC.

12:46 ..How many tapes shall we merge (0-255) ? (INTEGER)
10
```

This excerpt illustrates some of the general techniques used. Most output of the program is prefixed by 2 or more periods (..). All questions which require an operator response end with a question mark (?). Those that require a string accept any string of characters, excepting a totally blank string. Those that require a number returned will append a prompt to the question. A blank or null response is useful because it will suspend a TIMEOUTS question. Once a blank input has been given, MERGETAPE will wait indefinitely for the answer to the question. This also works before the question is posed, so if the operator needs to leave for a while and knows that an exceptional response needs to be given to the next question, he may transmit a blank and then walk away, knowing that MERGETAPE will stay waiting on the answer.

Many questions require only a YES or NO response. In error recovery situations MERGETAPE offers a choice between taking the recovery option or terminating; if the operator chooses to terminate he will be requested to re-input his decision by prompt messages which emphasise that what he has entered implies termination. If an unexpected response is entered, the question is repeated.

MERGETAPE reports

All reports use the MCP paging file attributes, hence a simple label equation of the file LINE will allow reformatting for a different page size, for example FILE LINE(PAGESIZE=44). By default, the reports will appear in upper and lower case. If the site does not have printers capable of lower case, it may change the default (for all reports created by the FLEX package) by means of the FLEX Inquiry command DEFAULT LPLOCASE OFF. Those sites that have a mixture of printer types, only some of which are capable of printing lower case, should associate a value of TRAINID with the file LINE. This can be done by label equation at compile time or run time.

MERGE dialog report

A report is produced each time a merge is run that contains the text of the dialogue between the operator and MERGETAPE. This can be used for training new operators, or for tracking down problems with the merged volumes. See Fig. 3-2 for an example.

Volumes released by MERGE command

If a merge releases volumes, a report is produced that shows each volumes released with its SERIALNO, TITLE, KIND, creation date, savefactor, and family structure. These volumes will also be listed by SERIALNO and TITLE in the MERGE dialogue report. If the METALOGIC tape library, TAPECONTROL, is in use at your site, the released volumes will be marked **PGOK** (purge authorised, may be used as scratch) in the tape library, with a note indicating that it was released by MERGETAPE on today's date.

Summary Report

This report is intended to be an “at a glance” summary of the state of the “virtual disk system”. It compares statistics gathered by MERGETAPE within the current run with those from the prior run of MERGETAPE. Any large variance should be a cause of, if not alarm, a search for an explanation. It was intended for the Operations Manager or a similar level of management. The numbers shown on the example for the number of resident file entries, non-cataloged file entries, file entries without backup, and within the table comparing resident to non-resident file entries by generation should remain relatively constant after a period of stabilisation. See Fig. 3-4 for an example.

As an example, one installation used a CATALOGLEVEL of 6, implying 7 generations. Approximately 50% of the files of generation 0 are non-resident. This can be interpreted as a “Virtual Disk Factor” of 2. The family in question contains eight thin-film diskpacks, and would probably have needed seventeen thin-film diskpacks to place all files on-line. This gives an indication of the worth of adopting a virtual disk system.

More extensive and flexible reports can be generated using the SELECT and REPORT commands of FLEX Inquiry. Some of theses are pre-defined, and may be accessed using the STD command. See the FLEX Inquiry manual for more details.

Merging Process

The Merging Process involves four distinct phases. First, there is a scanning phase, where all of the files on the pack and their backup volumes are considered. Secondly, there is the load phase, where files to be merged will be loaded to the workpack.(if not CDImage) Thirdly, there is dumping phase, where two new backups of these files are made. For CDImages files are copied directly to a cdimage and both backup references changed to point to that image. Lastly, the family is re-scanned to determine which volumes are no longer in use as backup and can therefore be released (release phase).

In practice, the time required to merge all files on a given family is too long to be practical. For this reason MERGETAPE allows you to select a limited number of backup volumes and do a partial merge. From the point of view of the tape librarian,

a merge is done to release volumes. For this reason the default merge selects volumes with a low number of files. This partial merge gives the maximum return for the minimum effort.

All non CDIMAGE merges work by copying files from backup volumes to a work pack. Even if a merge is terminated abnormally, the state of the resident files on the original family will be the same after the merge as before.

The following pages present a simple MERGE with emphasis on how the merge process works.

Scanning phase

This phase is where MERGETAPE examines the family selected. If run by a REPORT command, this is the only phase executed. In this phase, MERGETAPE is essentially performing a PD for each file on the family and building lists of all files in certain categories. If run in Merge Mode, MERGETAPE will first check that it is connected to a datacom terminal. Here is a sample of the terminal interaction in this phase:

```

u *metallogic/flex/mergetape dev cdimage
#RUNNING 9188
#?
10:43  -- MERGETAPE Log: 10:43 AM Wednesday, May 29,2019 --
10:43  METALOGIC Flex MergeTape Version 60.600.01
10:43  Compiled 13:07:36 on 12/03/2019
10:43  Request: DEV CDIMAGE
10:43  ..Do you want DEBUG info reported? (YES or NO)
no
10:43  ..Merging CD Images only from CDIMAGE
10:43  Family DEV will be merged (Media=CDIMAGE)
10:43  ..100 Volumes found in the Volume Library
10:43  ..Begin scanning Catalog
10:43  ..There are 3 volumes using less than 50 MBytes.
10:43  ..There are 4 volumes using less than 100 MBytes.
10:43  ..There are 7 volumes using less than 150 MBytes.
10:43  ..There are 12 volumes using less than 200 MBytes.
10:43  ..There are 14 volumes using less than 250 MBytes.
10:43  ..There are 16 volumes using less than 400 MBytes.
10:43  ..How many volumes shall we merge (0, or 3-16)? (INTEGER)
7
.
.
.
.

```

```

.
.
10:46  ..Volume PI0022 DELETED ok from Tape Library
10:46  ..Volume CDIMAGE_A [PD0026] will be released
10:46  ..Volume CDIMAGE_A [PD0033] will be released
10:46  ..Volume CDIMAGE_A [PD0034] will be released
10:46  ..Volume CDIMAGE_A [PD0031] will be released
10:46  ..Volume CDIMAGE_A [PD0032] will be released
10:46  ..Volume PI0001F2019148A [PI0001] will be released
#9188 FLEXLIB:PK506 REMOVED *DEV/PI0001F2019148A ON CDIMAGE
10:46  ..Disk file *DEV/PI0001F2019148A ON CDIMAGE successfully removed
#9188 FLEXLIB:PK506 ARCHIVE PURGE NOT DONE; NO RECORD(S) PRESENT: *DEV/PI0001F2019148A ON
CDIMAGE
#9188 PK506 *DEV/PI0001F2019148A NOT PURGED (NOT FOUND) CDIMAGE
10:46  ..Deleting PI0001 from Volume Library
#9188 VOLUME FAMILY PI0001F2019148A (PK) [PI0001] DELETED
10:46  ..Volume PI0001 DELETED ok from Tape Library
10:46  ..Volume CDIMAGE_A [PD0038] will be released
10:46  ..Volume S C R A T C H E D [HP0005] will be released
10:46  ..three (3) volumes released
#9188 MERGE:PK500 REMOVED *METALOGIC/FLEX/RULES/CATALOG/STATISTICS ON DEV
#9188 MERGE:PK500 REMOVED (FLEX)TEMPORARYFILE/MERGETAPE ON DEV
10:46  ..End of scanning the Catalog
#ET=2:55.0 PT=2.6 IO=3.3

```

In this example, a Merge has been requested for the family named DEV for CDIMAGE volumes. This dialogue is in fact from a live run, shown in the example in During the scan phase, MERGETAPE requires that the specifies where extra debug data should be displayed, then simulates a PD on every file on the requested family (e.g. DEV), and will build its lists. Every 3 minutes it will put out a message indicating its progress. The process of scanning a family has been made as fast as possible so it is unusual to get more than one or two such progress messages. A sort is done on the files selected to eliminate multiple generations of a single file from being merged in the same run. This restriction is essential, since the system disk directory will not allow access to two files of the same name on the same family (even if they are of different generations).

Merge phase

Having handled all exceptions, MERGETAPE now proceeds to the Merge Phase, asking about the number of LOADS.

This question is to determine how much effort the operator can spend on this merge, i.e. how many Volumes should be processed. Under ideal conditions, essentially all volumes specified will be released, i.e. made available for re-use. If 0 is entered for the number of tapes, MERGETAPE will not process any input volumes, and will not make any output tapes; however, any volumes that are no longer in use as backup will be released.

If a tape merge is being done then the operator is then presented with a 'picking list' of the volumes MERGETAPE would need, in the order it would use them, to perform a merge. If MERGETAPE is being run on a hard copy terminal, a tear-off list is produced; if run at the ODT or a screen terminal, a printed list can be generated. The list might look something like:

```
..Volumes needed to LOAD files on family SYSTEM
.. [B15175] [B15580] [G01090] [G01092] [G01179]
.. [G01076] [G01185] [G01186] [G01189] ...
..Total volumes to LOAD from: 10
```

The importance of the "picking list" is that MERGETAPE will wait for tapes to be mounted in that order. Tapes mounted out of sequence will not be used out of sequence. The reporting mode of MERGETAPE can be used in a limited way to provide the picking list before a merge is run. If the report is done too far in advance, the tapes requested by the subsequent merge may be completely different from those in the report, since the situation can be very dynamic.

Loading Files (Tape Merge)

This section covers the process of loading to a work pack. Files are copied to the workpack using standard WFL and LIBRARY/MAINTENANCE.

The operator is asked how many simultaneous copies he wants. MERGETAPE will handle anything up to 20 copies simultaneously, each copy being the responsibility of a separate stack. Each one is named so that a MX or A ODT command identifies the tape volume required. These stacks wait for the tape to mounted, ready the unit if required (ODT RY command), initiate the library maintenance, handle the restart recovery if required and rewind the tape when it is no longer required:

```

..I will handle RY and RW for you
..Number of tape drives available to MERGETAPE?  (INTEGER)
5
..All files backed up on [G01179] are already on the workpack
#BOT 3964  (META)COPY/TO/XSYSTEM/FROM/TAPEPE/B15175
...
#BOT 3965  (META)COPY/TO/XSYSTEM/FROM/TAPEPE/G01090

```

The operator should mount the first 5 volumes from the “picking list”. On the terminal, MERGETAPE is providing prompts for operator. These prompts are one of the many levels of recovery in this phase. In addition the status of all copies attempted will be reported. The following is an extract of this output:

```

..If problems with [G01090] SYSTEM83086G3, enter '3904 AX
HELP' at SPO
...
..Do you have a problem with TAPEPE SYSTEM86046G3 ?  (Enter
YES or NO)
YES
..Tape is: 1. Unavailable, 2. SCRATCH, 3. Other ?  (Enter
INTEGER number)
1
..Tape noted as unavailable
.....Abandoning attempts to load from TAPEPE [G01092]
SYSTEM86046G3
..Cannot complete copy from volume
SYSTEM86046G3 (TAPEPE,SERIALNO="G01092")
..Copy from volume SYSTEM82203A (TAPEPE,SERIALNO="G01185")
completed OK

```

In the above example, both a successful copy and an unsuccessful copy were shown.

This is an example of the most common operational problem involved in merging backup volumes: there is a non-resident file which is supposedly backed up on a tape volume. When a copy is attempted, the file is not on that tape. This situation may arise in a number of ways, but the most common is when a backup volume is scratched and reused. In our example, the operator has correctly entered an “AX NO” at the invitation to attempt a recopy. Recovery of this file is attempted at a later stage.

There are however cases where the correct response is “AX YES”. The most common is when a tape parity occurs. The operator may then dismount the tape, clean the drive or remount the tape elsewhere, and retry the copy.

Loading Files (CDIMAGE Merge)

For a CDIMAGE Merge files are Copied directly to the new output volumes

Recovery when Loading Files

As we have seen above there are cases where files are not loaded. This section explains how recovery of these files is attempted. Here is the output to the terminal:

```
..We have a problem
..10 files were not found on the volumes we just loaded
..Here are your recovery options
..1. Ignore these files and continue
..2. Terminate this merge
..3. Attempt reload from second backups
..Which recovery option? (Enter  INTEGER number)
3
..8 Files have no second backup
..Here are the additional 2 volumes needed
..Additional volumes needed for LOAD of family SYSTEM
..G00702 TAPEPE B69LOG2057 #001 OF 001 (1 FILE)
..G00104 TAPEPE SYSTEM82195E #001 OF 002 (1 FILE)
....Cannot complete copy from
B69LOG2057(TAPEPE,SERIALNO="G00702")
..Copy from volume SYSTEM82195E(TAPEPE,SERIALNO="G00104")
completed OK
```

The recovery process means attempt to use the second backup for the missing files. In the example there were 10 files which could not be loaded by the first pass. MERGETAPE keeps track of the files not yet loaded, and attempts the same process but now using the second backups. Sometimes there will be no second backup, as shown for eight of the files above. For the remaining two, a new “picking list” is produced, and the whole cycle is repeated. Even then only one of files is recovered. For the nine problem files, no further recovery can be made. MERGETAPE then offers the operator a further choice:

```
..We have a problem
..4 files were not found on the volumes we just loaded either
..Here are your recovery options
..1. Ignore these files and continue
..2. Terminate this merge
..Which recovery option? (Enter  INTEGER number)
1
```

At this point if our operator had not elected to continue, the program would have demanded a “tell me 3 times” response. The irrecoverable files are ignored, and the incorrect backup references will remain. These must be handled manually.

Creating merged tapes

Having loaded as many files as possible to the workpack, MERGETAPE now copies

them to new tape volumes to make two new backups for each file. More than one backup volume may be created, as there are limits on the number of files that can be copied within one WFL COPY statement. MERGETAPE will attempt to create single reel volumes.

```
..We will need at least two scratch tapes shortly
#BOT 3006 (META)WFLCODE ON SYSTEM
#BOT 3007 *LIBRARY/MAINTENANCE
#3007 SYSTEM90023MA/FILE000 REQUIRES VOLUMED PEXT #1
#3007 VOLUME FAMILY (MT) [G04081] DELETED
#3007 VOLUME FAMILY SYSTEM90023MA (MT) [G04081] #1 ENTERED
#3007 SYSTEM90023NA/FILE000 REQUIRES VOLUMED PEXT #1
#3007 VOLUME FAMILY (MT) [G03809] DELETED
#3007 VOLUME FAMILY SYSTEM90023NA (MT) [G03809] #1 ENTERED
(APLS)UTIL/SNVC COPIED AND BACKED UP FROM XSYSTEM TO
SYSTEM90023MA
(APLS)UTIL/SNVC COPIED AND BACKED UP FROM XSYSTEM TO
SYSTEM87023NA
...
#EOT 3007 *LIBRARY/MAINTENANCE
..Backups altered for (META)OBJECT/FLEX/LIBRARY ON SYSTEM
.. Timestamp 01/09/87 @ 04:57:13 Generation 1 (was
generation 1)
...
..Backups altered for (APLS)UTIL/SNVC ON SYSTEM
.. Timestamp 01/07/87 @ 01:30:13 Generation 3 (was
generation 3)
..Backup information for 67 file entries changed, it was NOT
changed for 2 entries
```

When all tapes have been created, MERGETAPE now verifies that the files are actually backed up on the new backup volumes. For each file on the workpack which has two good backup references, it gathers the kind and serial number of the backups, and then alters the backup information of the identical file on the family which is being merged. Finally it DESTROYs the entries on the workpack(unless it is the original pack as well).

Release phase

All that remains to complete the merge is to re-scan the pack to determine which volumes can be released.

If Metalogic TAPECONTROL is installed then each of the released volumes will be marked as PGOK (purge OK), and a note made that it was released by MERGETAPE on today's date. There are many other subtle but powerful links between FLEX and TAPECONTROL.

Error Messages from MERGETAPE

MERGETAPE requires a privileged usercode

MERGETAPE must be run from a privileged usercode. It is not sufficient to PP the code file.

MERGETAPE must run on a Cataloging MCP

MERGETAPE stores the results of its work in the catalog.

Familyname expected

A <familyname> was not found in the remainder of the <string parameter>

Invalid FAMILY name

The string parameter did not specify a valid large system familyname

Parenthesis expected

A left parenthesis did not follow the key word EXCEPT

Error in exception list

The <file list> was incorrect

Right Parenthesis expected

The <filelist> following EXCEPT was not terminated with a right parenthesis

UNSAFE ... expected

A valid key word specifying which type of <selection condition> was not found (after SELECT)

<integer> expected

The number of days for OLD tapes must be an integer

unknown SELECT criteria

The key word following SELECT was not one of AGE, UNSAFE etc.

Merging Tapes

If the system option 45(UseCatDefault) is set then any non library maintenance tape created gets a backup reference from the psuedo family TAPE. Mergetape keeps track of these references in the Statistics file.

If Tapemanager is used these noted references will prevent these tapes from being purged. Some time ago a configuration option FLEX_FAMTAPEREL was added to allow Mergetape to release tapes which had once had references on Family Tape but now had no references. The meaning of this option has now been extended so that Mergetape will only count references to Family Tape when this option is true. The option can be inspected and changed using U META/INSTALL CONFIG. The option is in the Full Flex section

Merging from Disk farms

MERGETAPE supports the merging of files from CopyWrite Disk Farms.

—— <Family> —— Farm — (<Farm Host>) —————|

The use of <Farm Host> is mandatory since, at this time, it is not possible to merge farms which may be on multiple PC hosts.

MERGETAPE will now automatically correct average MB values for both CD image and Disk Farms if they are found to be incorrect. MERGETAPE requires these values to determine the weighting of eligible volumes to be used in the merging process. Previously, Disk Farms values would always be 0 or 1 which affected volume selection. For Disk Farms, MERGETAPE will automatically search for the volume's LIBMAINTDIR file and calculate the average disk space by extracting the size of each file. If the Disk farm LIBMAINTDIR file is missing, MERGETAPE will determine the hostname of the Disk farm from the TRIM Tape Library subsystem and attempt to re-create the LIBMAINTDIR using a CopyWrite COPY.

MERGETAPE expects to find LIBMAINTDIR files under the RULES Manager usercode on the DL LIBMAINTDIR family.

During the merging process, if MERGETAPE determines that a Disk Farm is eligible for deletion (i.e. the Disk Farm volume has 0 references), an ADD *= FROM NULL request is initiated requesting CopyWrite to purge the farm at the specified host. When the delete is successfully completed, MERGETAPE will automatically delete any associated Tape Library entry, remove the LIBMAINTDIR and delete the Volume Library entry.

All the above activities are logged in MERGETAPE's dialog print file. This is an example of Disk Farm volume DF0020 being deleted from PC host META1:

12:26 ..Volume Library and TRIM entries will be removed

12:26 ..Volume DEV_A [DF0020] will be released

12:26 ..Disk Farm DF0020 will be deleted from META1

12:26Removing old LIBMAINTDIR for DF0020

12:26(FLEX)LIBMAINTDIR/DEV_A/20040512/DF0020 ON DISK
successfully removed

12:26Deleting DF0020 from Volume Library

The WFL deletion of an extinct Disk Farm would typically look like:

#BOT 14081 (IPP)WFLCODE

#BOT 14082 (IPP)FILE/TRANSFER

#BOT 14083 (IPP)FILE/TRANSFER/SERVICES

#14083 CopyWrite:Metalogic CopyWrite Version 50.500.03 was compiled at 16:41:8 on the 18th May 12004 (MCP 49.189.8737)

#13802\14085 BOT (IPP)SESSION/"Mix# 14083"/"10.0.0.2"/

META1/IPP/COPYWRITE

#14083 CopyWrite:Release ALL 4 FILES in SPARE_A

#14083 CopyWrite:File Release Completed. Volume Release Check Pending

#14085 CopyWrite:[META1] Disk Farm C:\ASeriesBackup\DELL8500MCP\
SPARE_A[DF0017] Removed

#14085 CopyWrite:[META1] Deleting 4 Files in C:\ASeriesWaste\
SPARE_A[DF0017]_21_May_2004_155812

#14085 CopyWrite:[META1] Deleted C:\ASeriesWaste\
SPARE_A[DF0017]_21_May_2004_155812\FILE000.dir

#14085 CopyWrite:[META1] C:\ASeriesWaste\
SPARE_A[DF0017]_21_May_2004_155812 Deleted

#13802\14085 EOT (IPP) (IPP)SESSION/Mix# 14083/10.0.0.2/META1/
IPP/COPYWRITE

#EOT 14083 (IPP) (IPP)FILE/TRANSFER/SERVICES

#EOT 14082 (IPP) (IPP)FILE/TRANSFER

#EOT 14081 (IPP) (IPP)WFLCODE

If Disk Farms volumes hold very large files which drop out of the catalog on a regular basis, it is possible that disk space shortages may become an issue on PC hosts. This can occur because such files will never be released until the volume is merged or drops to zero reference by natural means.

To help minimize this problem, a modifier to the RELEASE command is available:

—— Release —— Farms ————— |
 Threshold

The RELEASE FARM command will analyze all existing Disk Farms in the Volume Library searching for large files (or older generations of those files) that have dropped out of the catalog. By default, MERGETAPE will use a threshold file size of 5 Megabytes (5000 KB) to select files but this may be raised or lowered by using <KB Threshold> to specify a lower limit in KiloBytes.

For example:

RELEASE FARM 8000

The above command would look for expired files that have now dropped out of the catalog and are larger than 8 Megabytes. MERGETAPE uses each Disk

Farm's LIBMAINTDIR file to select eligible files so this must file be resident or the volume will be excluded. Further, the family associated with each eligible file must be online and available or the release will be skipped.

When CopyWrite receives a request to release a file from a Disk Farm, the file is not removed but replaced by a 'RIP' capsule which is essentially a null file, occupying minimal disk space. This means that a reference to the file remains in the Disk Farm but the disk space has been returned.

At this time, neither MERGETAPE nor CopyWrite 'know' that a released file has been processed before so multiple RELEASE FARM runs may re-release the same file. This problem will be fixed in a later version of Copywrite by marking the file as released in the associated LIBMAINTDIR. A RELEASE FARM command is much faster than a Disk Farm merge and can easily return significant disk space.

In practice, a combination of merging and RELEASE is envisaged as the best way to housekeep Disk Farms. As with CD images, MERGETAPE can be run silently, without operator intervention, to merge Disk Farms by PC host, at scheduled times. For example, to merge family DEV with farms on PC host META1:

```
RUN *METALOGIC/FLEX/MERGETAPE("DEV FARM (META1) AUTO")
```

This run uses automatic settings that controls Farm selection, by default, using less than 25 MBytes to a maximum of 10 volumes. These values can be readily overridden by run-time modifiers.

Merging CDs

FLEX can use CDRoms as output media using the Metalogic product CopyWrite. CDs are used for output media from MERGEs if the DEFAULT DENSITY is 64. It will take the family to create the CD images from the CONFIG variable FLEX_CWIMAGE, and the HOSTNAME to send the images to from the variable FLEX_CWHOST. These can be overridden by label equating the FAMILYNAME and HOSTNAME of a file called CDIMAGE.

The operator is also prompted to confirm the family.

Unlike for tape output, all the output volumes are generated in parallel, but that is not when the backup refs are updated. MERGETAPE waits for the CDs to be mounted, before it updates the backup refs. Because a CD does not have a SERIALNO in its label, it has to be placed in the Volume Name instead. The CDs have Volume Names of this form:

CDSSSSSSM19983333A

where SSSSSS is a SERIALNO generated by FLEX itself,

M means made by MERGETAPE (F for FAMILYMANAGER)

1998333 is a 7 digit Julian Date

A is the output volume identifier of this run (first A, then B,C)

A file entry backed up to CD will appear to have 2 identical PACK backup references, although the SERIALNO will be non-numeric, something impossible for a real PACK.

CDs can also be used for LOADs and MERGETAPE can also RELEASE them when they are no longer in use for backup.

Although the implementation does not disallow it, it is strongly recommended that CDs be given a serial range completely dis-joint from any existing tapes. FLEX itself allocates the SERIALNOs based on a starting value in the CONFIG variable FLEX_CWSERIALNO.

The starting value should have 1 or 2 alphabetic characters in it at either or both ends, with at least 4 contiguous numerics, e.g.

A00000, E0000E, CD1000 etc

FLEX will then add one to the numeric field each time it needs to allocate a SERIALNO.

During this large implementation, some small problems were dealt with including fixing GEN merges, and allowing new TAPE densities.

Merging CD Images

When a CD image file is released by MERGETAPE, the Volume Library entry is deleted, the image disk file is removed and any Tape Library entry is deleted. Normal CD volumes will remain in both the Volume Library and METATAPELIB after release, as with tapes.

MERGETAPE will automatically ALTER merged Copywrite disk images to assign the LOCKEDFILE attribute; this change helps to protect the file against accidental removal.

The option, MB, assists batch merges of CD and CD images. If specified, MB controls the megabyte threshold value used by MERGETAPE to automatically select those CD/CD images which are eligible for merging. For example, MB=25 will only select tapes with less than 25 megabytes of active files in the CD or image.

Configuring FLEX

FLEX DEFAULT command

Following on a recent expansion of the DEFAULT command in FLEX Inquiry, FAMILYMANAGER can now be driven entirely by information in the Rules File or in parameter syntax. Most parameters which were given by label equation are now in the Rules File. In particular, the density, length, recovery strategy (ex PROTECTION), SAVEFACTOR, and SCRATCHPOOL of the output tapes, as well as the use of &COMPARE or &VERIFY, and the restriction to a single output tape. The activation of the BIGFILES and HISTORY features is now solely dependent on the RULES OP settings for the family concerned.

Two new keywords are allowed after the familyname in the parameter. They are DUPLICATE, which causes the output tapes to be duplicated, and OFFSITE (ex BLOCKSTRUCTURE=EXTERNAL), which marks the tapes OFFSITE.

Install utility CONFIG menu

Both FAMILYMANAGER and MERGETAPE will now use a default SCRATCHPOOL when creating output tapes, if one is assigned to the MAGUS config variable FLEX_SCRPOOL. Using the SCRATCHPOOL attribute in run-time file equation will override this setting.

The variable may be set up as usual using the INSTALL utility:

```
U META/INSTALL FLEX_SCRPOOL=MYSCRATCH_POOL
```

Further, the PV OFFSITE command will now update the Volume Library for that tape volume, as if a 'WFL VOLUME OFFLINE...' command had been issued for that tape, but only on MCP versions greater than 41.0 or with the configuration variable SYS_OFFSITEOK set to TRUE.

When dumping files to CD using CopyWrite, certain configuration variables are required. If they are not set up the generated copy fails with syntax errors. The variables are now checked during initialisation, and Familymanager aborts if they are not present.

FLEX_CWHOST should be set to the host name of the system where the CD image file should be sent to be burned.

FLEX_CWIMAGE is the name of the family where CD image files are stored.

FLEX_CDSEIALNO is a six character string representing the next serial number to be used for CD output.

If the configuration variable FLEX_NOESTORE is present then MERGETAPE will

not use the WFL RESTOREADD command when loading tapes to disk.

This helps to avoid the problem of using older tapes which are incapable of supporting this feature. This only affects sites that are currently dumping multiple families to a single tape ,

Migration

When a site switches from one tape subsystem to another e.g. half-inch cartridge (BPI38000) to Exabyte (BPI11000), then the default density for FAMILYMANAGER and MERGETAPE must be changed to the new density. This causes problems for MERGETAPE since it will consider the feet of tape values extracted from the FLEX's STATISTICS file to be misleading when evaluating the "weighting" of eligible tapes for a merge.

To assist with such a situation, MERGETAPE will search for a specific configuration variables called FLEX_CHANGEDATE and FLEX_OLDDEN. These variables allow the customer to "tell" MERGETAPE the date that hist tape library was converted and the density of any tapes created prior to that date. If both variables are present and correctly set up, MERGETAPE will use the information to adjust the tape feet weighting of eligible tapes during a merge as though they had been created with the new tape subsystem.

The variables may be set using the INSTALL utility:

```
U META/INSTALL FLEX_OLDDEN=6250
```

```
U META/INSTALL FLEX_CHANGEDATE=1997123
```

files to be WFL ADDED from a specific source tape could have caused the To avoid confusion, the value for the FLEX_CHANGEDATE variable should MERGETAPE was incorrect causing the same problem to occur. This use a 7-digit julian date.

Flex Recovery Features

Many of the techniques described here to handle recovery situations are discussed in the Unisys DISK SUBSYSTEM ADMINISTRATION guide; these notes are intended to supplement the information provided in that guide.

FLEX recovery techniques assume that a valid Catalog exists. There are a number of cases where FLEX can help. In all cases of a lost physical pack, the first action should be to take a safe copy of the current Catalog with a COPYCAT ODT command. Then a new pack should be RC'ed with the same name and SERIAL as the one lost. The MCP will ask:

```
OK TO RE-ENTER FAMILY INTO CATALOG?
```

The response should **always** be OK because the MCP can get into some quite confusing states otherwise. This RSVP is only present to guard against mistakes in the RC command. Do not be disturbed by any CATALOG DELETED messages. These messages come from the files that were not backed up on the family when it crashed.

Once the new pack is RC'ed and OK'ed, the main cases can be dealt with as follows.

Case 1. You have lost one member of a multi-pack Family:

We assume that you still have a directory. If you have lost the base pack, then duplicated directories would have solved the problem. If there is no directory, go to case 2.

The technique to use here is the FAMILYMANAGER DISCARD command. The problem you are resolving is that some files were partly on the lost family member. There is no effective Unisys supplied technique to identify which files. Having identified the affected files, they must be removed. If the file had backup references in the Catalog, it can be restored by a LOAD. If it was not backed up it cannot be recovered by FLEX. The DISCARD feature is documented in the FAMILYMANAGER chapter.

Case 2. You have lost a whole disk family:

If the family was the DL Catalog family, go to case 4.

FLEX Inquiry has the ability to read the catalog copy taken at the time of the problem, determine which files were resident, and to LOAD them from backups. Files which had no backup references cannot be recovered by FLEX. If the copied catalog is called SYSTEM/CATALOG/001 ON SAVEPK, the syntax for recovery is:

```
MAINT ALT CAT SYSTEM/CATALOG/001 ON SAVEPK  
LOAD (*)= ON <family> : LINEAR,REO
```


See Chapter 3 of the FLEX Inquiry manual, under the OPTION command for a discussion of the LINEAR option.

Case 3. You have lost the Halt-Load Family:

Apart from the inevitable halt-load, this case is identical to case 2 and provides no significant difficulties unless a Cold-Start is required, in which case care must be taken to restore the settings of the CATALOGING Option and the DL CATALOG family.

Case 4. You have lost the DL CATALOG Family:

It is not possible to recover this family to the state just before the crash. The closest point is to the state at the last backup copy of the catalog. Backup copies should be taken by the Odt **COPYCAT** command every few hours, e.g. every 6 hours. Users of Metalogic SUPERVISOR can schedule this quite easily. Also at least once a day a Catalog should be put on tape and sent off site.

When the machine is halt-loaded after the DLCATALOG family is lost, it will ask for a new DL CATALOG family. It is normally best to reset Cataloging and halt-load again to avoid the overhead of re-building the Catalog. Try to do the minimum on the system until the new pack is made available. It can often save time to turn off un-needed packs until Cataloging is restored.

When the RC has been done, COPY the latest saved Catalog onto the new pack, set OP CATALOGING, and the DL CATALOG back to that family, then halt-load. OK all RSVPs and wait for the rebuilds to finish. Power back up any packs that were powered off earlier. They too will re-build. Using the last saved Catalog, recover the DL CATALOG family as in case 2.

Case 5. You have lost the entire system

RC the packs that you wish to re-build. Set Cataloging and the DL CATALOG Family. COPY the latest available Catalog to the DL CATALOG Family and also the usual family used as the destination for the COPYCAT. Set CATALOGING and Halt-Load. Run META/INSTALL to restore the FLEX software on the usual families. Run FLEX Inquiry under the Rules Manager usercode and input:

```
MAINT ALT CAT SYSTEM/CATALOG/001 ON SAVEPK  
LOAD (*) = : LINEAR,REO
```

assuming that SAVEPK is the usual COPYCAT destination pack.

Case 6. You have lost the entire system and tape library

This is similar to Case 5, except new keys will be needed for the backup system for FLEX. Also the tapes available will be all the off site tapes. It is normally better to mark the old on site tapes destroyed before proceeding with the LOAD. This can be done from CANDE as follows:

```
U FLEX
LISTVL SIMPLE TAPE ONSITE:F X
Q
G X
REP COL 1 /WFL VOLUME DESTROYED/
DO
U FLEX
```

and continue as in case 5.

Copyright

Copyright © 1979-2019 Metalogic S.à R.L., Luxembourg.
All rights reserved.

Last revision: May 2019

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.