PERKIN-ELMER

OS/32 FASTBACK

Reference Manual

48-063 F00 R00

The Perkin-Elmer Corporation, Data Systems Group, 2 Crescent Place, Oceanport, New Jersey 07757

TABLE OF CONTENTS

CHAPTERS (Continued)

CHAPTERS (Continued)

APPENDIXES

APPENDIXES (Continued)

TABLES

PREFACE

This manual describes the new OS/32 fast disk backup and restore utlity - OS/32 Fastback R00-01, program number 03-345.

OS/32 Fastback contains a very efficient disk-to-tape backup capability, and is thus aimed at the operator who needs to take fast security backups of large amounts of data.

Users of Fastback should be familiar with OS/32, Perkin-Elmer's multi-tasking operating system for its 32-bit processors. An understanding of the Multi-Terminal Monitor (MTM) is also desirable.

Chapter 1 of this manual gives a general overview of disc backup procedures, and of Fastback in particular. Chapter 2 outlines how to load and start Fastback. Chapter 3 gives a detailed explanation of Fastback commands and their meanings. Chapter 4 describes the extended file selection masking available with Fastback. Chapter 5 explains the operation of Fastback and includes timing and tuning information. Chapter 6 discusses Fastback list output and error messages. Chapter 7 outlines Fastback internal failure codes and what to do if you encounter one of them. Finally, the Appendixes to the manual contain command summaries, end of task codes, linkage procedure, logical unit usage, and information regarding Fastback comparisons/compatibility with other products.

NOTE

> It is strongly recommended that those users who wish to obtain the best possible performance from Fastback should read Chapter 5 in order to understand how Fastback operates and the optimisations which may be effected to maximise its performance.

OS/32 Fastback R00-01 executes as a user-task (u-task) under OS/32 Revision 6.2.

For information on all 32-bit manuals, see the 32-Bit Systems User Documentation Summary.

CHAPTER 1
OS/32 FASTBACK OVERVIEW

1.1 INTRODUCTION

In computer terminology, the word "backup" generally has two meanings. The first meaning refers to making a copy of some specific data, usually copying the data from on-line storage (such as a disc) to off-line storage (such as a magnetic tape). The actual copy procedure is called the backup operation. The second meaning of "backup" refers to the result of the backup operation - that is the actual copy of the data. This copy may be called a "backup copy", or simply a "backup".

Backup copies of data are normally made for one of two reasons. Firstly, as a security measure, in case the original data is lost, and secondly as an archiving operation, in order to file away old data and release the on-line storage space it occupied, for further use.

Backup copies of data may be done by reading and writing logical records from/to files through the access mechanisms provided by the computer's host operating system. Perkin-Elmer's OS/32 Backup utility provides a backup/restore capability which operates with this "logical file/logical record" mechanism. This method of operation also enables OS/32 Backup to be readily usable for copying files from one storage medium to another. However, the use of logical access methods can impose limits on the throughput that may be required for backup operations. This lack of throughput may pose problems for different types of users - especially those users who have large on-line systems on which regular security backups are required.

OS/32 Fastback is designed to meet the requirement for a fast disc-to-tape backup utility. In order to fulfill this requirement, Fastback copies data directly from the physical input disc device with minimal use of the operating system's logical access paths. However, the tradeoff for the fast physical backup comes in the tape-to-disc restore operation. Restore operations using Fastback require on-line disc work space, and proceed at a somewhat slower rate than backup operations. This is because a restore operation involves a physical to logical record mapping, in order to reconstruct the logical files which contain the data.

Thus OS/32 Fastback is ideal for archiving "old" data and for taking end-of-day security copies of volatile data.

In general, Fastback is intended to be used for backing up large amounts of data from disc to magnetic tape, in situations where the backed up data is seldom needed to be restored.

1.2 GENERAL DESCRIPTION

The OS/32 Fastback utility program provides OS/32 users with a comprehensive backup/restore facility, for use in an OS/32 or MTM environment.

OS/32 Fastback provides four different functions:

- disc to magnetic tape backup operations,
- magnetic tape to disc restore operations,
- verification of previously created Fastback tapes against files currently held on disc, and
- listing of files contained on previously created Fastback tapes, or listing of files which may be backed up from input discs.

Each of the four functions may operate in one of two possible modes:

- selective, in which only certain files are selected, or
- non-selective, in which all files are processed.

In addition, Fastback has the following features:

- support for all current Perkin-Elmer disc and tape unit offerings,
- high performance backup capability,
- easy to use commands that may be entered interactively as responses to prompts, specified in a file, or included as arguments to the OS/32 START command,
- all non-interactive commands identified by simple keywords,
- comprehensive file selection facilities including an extensive filename masking ability, the ability to rename files, and individual and global since/before date/time file selection,

- options that will: verify data transferred, specify action to take when errors are encountered, specify the size of the buffers to be used, etc.,
- a detailed listing of the files backed up or restored, and
- messages that are self-explanatory.

1.3 OS/32 FASTBACK REQUIREMENTS

The following resources are required for Fastback R00-01 execution:

- a Perkin-Elmer 32-bit computer system running under OS/32 R06.2,
- one disc device,
- one magnetic tape device, and
- approximately 126kb (including default segment size increment) of main storage over and above OS/32.

In addition, disc workspace is required for restore and verifyonly operations. This is necessary to contain the tables required for the physical to logical mapping procedure mentioned earlier. Chapter 5 discusses Fastback operation, including the various amounts of memory and disc space which may be required.

CHAPTER 2
GETTING STARTED

2.1 WHEN TO RUN FASTBACK

Fastback may be run whenever the user requires a backup or restore operation, etc., to be performed. The frequency of such operations is dependent on the user's operational environment. For example, a user running a large on-line system may require regular backups of different parts of his system in order to maintain a hierarchy of backups. In such a case, daily backups of transactions occurring during business hours may be undertaken (if the on-line system is running DMS/32, then backing up the DMS/32 log file would provide this level of security). Less frequent backups of the entire database may also be made (such as backing up an entire DMS/32 database, perhaps once per week). In a program development environment, regular backups of new work may be performed. For both of these environments, Fastback is an ideal tool for backing up the large quantities of data involved.

It should be realised that Fastback is optimised for backing up data, not restoring it. Hence the user who wishes to backup and restore (i.e. copy) a limited number of (small) files may find that OS/32 Backup will better meet his requirements, in that the total backup and restore time taken may well be similar, and OS/32 Backup will require less main memory and work space than Fastback.

2.2 LOADING FASTBACK

2.2.1 Loading Fastback from the System Console

This command loads Fastback from the system console.

Format:

```
LOAD taskid ,fd ,segsize increment
```

Parameters:

taskid	is a 1- to 8-character alphanumeric string specifying the name of the task after it is loaded into a foreground segment in main storage.
fd	is the file descriptor of the device or file which contains the task image load module of Fastback. If this parameter is omitted, the default is "taskid.TSK".
segsize increment	is a decimal number in kb specifying the additional main memory workspace to be added to the module's impure segment. This value, if given, overrides the WORK=n option which may have been specified when the task was linked.

NOTE

Fastback will generally execute faster if it is allocated more memory. For information on the choice of optimum segment size increment see Chapter 5.

2.2.2 Loading Fastback from an MTM Terminal

This command loads Fastback from an MTM terminal.

Format:

LOAD [taskid] ,fd ,segsize increment

Parameters:

taskid	is a 1- to 8-character alphanumeric string specifying the name of the task after it is loaded. This parameter is ignored by MTM.
fd	is the file descriptor of the device or file which contains the task image load module of Fastback.

segsize increment — is a decimal number in kb specifying the additional main memory workspace to be added to the module's impure segment. This value, if given, overrides the WORK=n option which may have been specified when the task was linked.

NOTE

Fastback will generally execute faster if it is allocated more memory. For information on the choice of optimum segment size increment see Chapter 5.

2.3 STARTING FASTBACK

After Fastback has been loaded, the START command may be used to begin execution of the program. The format of this command is identical in both the OS/32 System Console and MTM environments.

The START command which should be used for Fastback, has the general format:

$$\text{START}\ \left[,\text{param}_1\ \left[,\text{param}_2\ \left[\ldots\left[,\text{param}_n\right]\ldots\right]\right]\right]$$

Depending on the parameters used in the START command, Fastback will commence operation in one of three possible command entry modes: interactive, batch or immediate.

Note that once Fastback has validated all commands and is about to commence the requested operation, if the commands have been entered other than via the system console, the following message is logged to the console:

OS/32 FASTBACK Rnn-nn <function> OF <pack> STARTING

where <function> is either BACKUP, RESTORE or VERIFYONLY and <pack> is the volume name of the disc pack involved in the operation.

2.3.1 Starting Fastback in Interactive Mode

This command is used to start Fastback in interactive mode so that the Fastback commands can be entered in a conversational manner.

Format:

START [,COMMAND [=] [idev:] [,LIST [=] [fd]]]

Parameters:

COMMAND= idev: specifies the input device from which commands are to be entered and must be an interactive device. If this parameter is omitted or idev: is omitted, the default command entry device is CON:. Note that throughout this manual all references to the device name CON: in fact refer to the name of the system console. (The name of the console may be other than CON:.)

LIST= fd specifies the output device or file to which all listing information is to be output. If this parameter is omitted or fd is omitted, the list device may be specified during the interactive command input phase. If a file is specified to receive the list output, the file must already exist. The list output is appended to any existing data in the file.

Functional Details:

When started in interactive mode, Fastback carries on a question (i.e. a prompt) and answer dialogue with the user. Interactive mode is most suited to inexperienced and/or infrequent users of Fastback. A detailed explanation of the interactive mode prompts, and their possible answers, may be found in Chapter 3 of this manual.

After Fastback is started, this message is displayed:

OS/32 FASTBACK Rnn-nn

where nn-nn gives the revision and update level of Fastback. The first prompt message for the dialogue is then displayed.

Examples:

```
START
```

implies that the system console or MTM terminal is to be used as the command device

```
ST ,LI=PR:, C=CRT1:

ST ,LIST FASTBACK.LST
```

2.3.2 Starting Fastback in Batch Mode

This command is used to start Fastback in batch mode so that the Fastback commands are read from a file or device. These commands are fully described in Chapter 3.

Format:

START [,COMMAND [=] fd_1 [,LIST [=] [fd_2]]]

Parameters:

COMMAND=	fd_1 specifies the input file or noninteractive device from which commands are to be read.
LIST=	fd_2 specifies the output device or file to which all listing information is to be output. If this parameter is omitted or fd_2 is omitted, the list device may be specified (using the LIST command) by one of the commands read from the batch file. If this parameter is omitted from both the START command and the batch command file then PR: is used as the default list device. If the assignment to PR: fails then CON: is used as the secondary default. If a file is specified to receive the list output, the file must already exist. The list output is appended to any existing data in the file.

Functional Details:

When started in batch mode, Fastback reads commands from the specified command file or device. These commands are documented in Chapter 3.

After Fastback is started, this message is displayed:

```
OS/32 FASTBACK Rnn-nn
```

where nn-nn gives the revision and update level of Fastback. Commands are then read from the batch file or device.

Examples:

```
START ,COMMAND=SCRT:FASTBACK.CMD

ST ,LI=PR:, C=CR:
```

Where:

CR: is a card reader.

2.3.3 Starting Fastback in Immediate Mode

This command starts Fastback in immediate mode. This mode is identical to batch mode (in that the commands to Fastback are identical), except that no command device is specified. All the commands required by Fastback must be passed to Fastback as arguments of the START command.

Format:

START ,cmd_1 ,cmd_2 [,cmd_3 [,cmd_4...]]

Parameters:

cmd_1, cmd_2, etc.,	are the Fastback commands. These are discussed in Chapter 3.

Functional Details:

In immediate mode all commands to Fastback must be passed via the START command. It is not possible to pass some commands to Fastback via the START command and the remainder from a command file. Thus for example, the following START command will be rejected:

```
START ,BACKUP,DISC=DSC1:,COMMAND=SCRT:FASTBACK.CMD
```

Immediate mode is best used when the commands to Fastback can be passed via the START command. If the required commands cannot be passed via the START command, then batch mode must be used.

Whenever Fastback is started in immediate mode, any specified SELECT command referencing an "immediate" operand must contain only one select entry; i.e., multiple entry select immediate operands are not allowed. See Section 4.1.

After Fastback is started, this message is displayed:

```
OS/32 FASTBACK Rnn-nn
```

where nn-nn gives the revision and update level of Fastback. The commands passed in the START command are then processed.

Examples:

```
START ,BACKUP,DISC=DSC1:,TAPE=MAG1:,LIST=PR:

ST ,RESTORE,TAP=MAG2:,DI=DSC2:,LIST PRIN:

ST ,DISPLAY,TAPE MAG1:,LI=PR:,SEL=(-.-/194)
```

2.4 STOPPING FASTBACK

Once Fastback has processed all its initial commands, it commences operation. If the user desires to halt Fastback after it has begun execution, he may use the SEND command of OS/32 to issue a STOP command, which instructs Fastback to terminate in an orderly manner.

Format:

```
SEND STOP
```

Functional Details:

This command, when received by Fastback, instructs the program to make an "orderly" shutdown. To effect its shutdown, Fastback will close any open logical units and delete any workfiles currently being used.

Regardless of the initial command mode (i.e., interactive, batch, or immediate), reception of the STOP command will cause Fastback to terminate.

NOTE

This command is mainly intended for use when Fastback is performing a restore operation. Given that a restore is taking place, the use of this command is preferred to simply cancelling Fastback because it allows the program to delete any intermediate files which it may have created on an output disc.

CHAPTER 3
COMMAND ENTRY

3.1 INTRODUCTION

Certain commands must be given to Fastback to specify the function to be performed, the mode of that function, and any required options. As discussed in Chapter 2, these commands can be entered either interactively in response to a series of prompts, or from a file or input device, or as part of the operating system START command.

In this chapter each command is described. Section 3.2 discusses the commands used in the batch and immediate modes of command entry, and Section 3.3 describes the prompts and responses for interactive mode.

3.2 BATCH AND IMMEDIATE MODE COMMAND ENTRY

These two methods of command entry are provided for experienced users who are familiar with the command syntax. Batch mode entry is used either when the number of options to be selected is too great to be entered as part of the operating system START command, or when it is convenient to set up a fixed command file that will be used repeatedly.

When commands are entered as arguments to the operating system START command, each command is separated by a comma. (Note that some commands may have multiple parameters separated by commas, but these are enclosed in parentheses.) When commands are entered from a batch file, each record read from the file can contain multiple commands and these can be separated either by commas or semi-colons.

NULL commands are allowed and are ignored; i.e., a START command of the form

ST ,arg_1,arg_2,,,,arg_6

is valid. Similarly a line read from the batch file can validly have the format

cmd_1,cmd_2;;;cmd_3,,cmd_4;,cmd_5

In general, blanks are ignored; i.e., a command may be preceded or followed by blanks. The following are all equivalent:

cmd_1,cmd_2,cmd_3

cmd_1 , cmd_2 ; cmd_3

cmd_1, cmd_2, cmd_3

In addition, many of the commands have the form:

KEYWORD = value

In these cases, any blanks preceding or following the equals sign are ignored and the equals sign is not necessary if at least one blank separates the keyword from the value. Thus the following formats are all valid and equivalent:

KEYWORD value

KEYWORD=value

KEYWORD =value

KEYWORD = value

KEYWORD= value

In addition, where the value itself has some default setting, the following formats are all valid and equivalent:

KEYWORD

KEYWORD=

KEYWORD =

KEYWORD=default

Comments can be entered by preceding the comment by an asterisk. The comment is taken to extend to the end of the command line. Thus, for example, in the line

$cmd_1,cmd_2,*$ SET OPTIONS 1, 2 AS REQUIRED

all characters to the right of the asterisk are taken to be a comment even though the comment may contain commas or semicolons. This means that on a given line no command will be recognised if it follows a comment since it will be taken to be part of the comment.

A line read from a batch file may be terminated by a carriage return character.

The batch file is read until one of the following conditions is encountered:

- End of file status (X'88') is returned.
- End of medium status (X'90') is returned.
- The command 'END' is recognised. Any commands following the END command on the line containing it will not be processed.
- An erroneous command is detected.

The individual commands are described in the following sections. The commands can be divided into three classes: those which specify the function to be performed; those which set the mode of the function; and those which set options and values. Only certain combinations of commands are allowed. For example, it is not valid to specify more than one function or mode. When all commands have been entered and individually validated, a consistency check is performed before initiating the requested operation.

Table 3-1 shows which commands are allowed for each possible function and mode. An M indicates that the command is mandatory, an O that it is optional, and a blank that it is invalid for the given function.

TABLE 3-1 COMMAND FUNCTIONS AND MODES

COMMAND	BACKUP	RESTORE	VERIFYONLY	DISPLAY
BACKUP	M			
BEFORE	O	O	O	O
BUFFERSIZE	O			
DISC	M	M	M	O
DISPLAY				M
ERRORPOOL	O	O	O	
KEEPFILES		O		
NOPROTECT	O			
ONERROR	O	O	O	
POSITIONING	O	O	O	O
RESTORE		M		
SELECT	O	O	O	O
SINCE	O	O	O	O
TAPE	M	M	M	O
TEMPLATE		O		
VERIFY	O	O		
VERIFYONLY			M	
WORKSPACE		O	O	

Note that the commands SINCE and BEFORE are mutually exclusive, as are the DISC and TAPE commands if the DISPLAY function is requested.

Note that any error messages generated whilst the commands are being processed are logged to the system console (or MTM terminal) and not to the list device. These messages are documented in Chapter 6, but in general consist of a plain language message followed by the erroneous command line with an indication of the error position.

3.2.1 BACKUP

The BACKUP command is used to select the backup function.

Format:

BACKUP

Parameters:

None

Functional Details:

This command informs Fastback that a backup operation is to take place. The operation will involve backing up a disc pack to a magnetic tape.

The (input) disc specified via the DISC command must be currently ready and marked on protected, unless the NOPROTECT command has been specified, in which case the input disc may be marked on unprotected. It is preferable, however, for the disc to be marked on protected whenever possible.

Use of a core directory (CDIR) on the input disc does not increase the speed of the backup operation.

The (output) magnetic tape device specified via the TAPE command must be currently online, write-enabled, and positioned according to the user's requirements. Tape positioning may be specified through the use of the POSITIONING command.

The tape should not be positioned such that Fastback will encounter the end of the tape before it has finished writing its header labels. Fastback must be able to write a complete set of header labels on the tape. Also, whenever possible, the tape should be positioned such that the "directory" file, which is the first file Fastback writes to tape, will fit entirely on the first tape of the backup. (This file is relatively small, usually occupying less than 300 K bytes.) This is because restore operations require two passes of the "directory" file; hence, if the file crosses tapes, then tape swaps will be required early in the restore operation.

It is recommended that backup operations be commenced at the start of a reel of tape (i.e., from BOT), simply for operational convenience.

Examples:

BACKUP

BA

BACK

3.2.2 BEFORE

The BEFORE command sets a global time limit on the age of files which are to be processed. Files will be processed if their date and time last written (i.e., updated) is older than the date and time specified by the BEFORE command.

Format:

BEFORE[=][date] [,time]

Parameters:

date	is the date component of the time limit.
time	is the time component of the time limit.

Functional Details:

The BEFORE command sets a global time limit value. This value is used in one of two slightly different ways, depending on whether or not Fastback is operating in select mode.

If the SELECT command is not entered, then BEFORE sets a time limit which will determine which files on the input medium are to be processed. Thus, in this case, BEFORE will cause the program to operate in a "pseudo select" mode where all files on the input medium will be considered for processing, but will be processed only if their date and time last written is older than the time limit specified by the BEFORE command.

If the SELECT command is encountered, then BEFORE sets a default global time limit for files matching the specified select criteria. Thus if a file on the input medium is to be processed, it must match a select entry and its date/time last written must be earlier than the specified global BEFORE time limit. Individual select entries may contain their own BEFORE (or SINCE) commands which will override the global BEFORE time limit. If both SELECT and BEFORE commands are used, then if no files on the input medium match the select criteria, no files will be processed. That is, files which fail to match select criteria will not be processed simply because they meet a global BEFORE time limit.

The BEFORE command (and the SINCE command - see Section 3.2.15) may specify a date, a time, or both, which will be used in the time limit comparison.

The date component of the time limit may be given in any of the following formats:

dd/mm/yy	- e.g.	25/12/83
mm/dd/yy	- e.g.	12/25/83
dd/month-mnemonic/yy	- e.g.	25/DECEMB/83
month-mnemonic/dd/yy	- e.g.	DECEMB/25/83

If the all-numeric date format is used (i.e., dd/mm/yy or mm/dd/yy), Fastback will interpret the date according to the currently sysgened date option. For example, if Fastback encountered a date of 25/12/83, then it will interpret the date as referring to the 25th of December 1983, providing the operating system was sysgened with European date format. However, if the operating system was sysgened with U.S. date format, then Fastback will declare the given date to be invalid, and will generate an explanatory error message since the 25th month does not exist. On the other hand, if Fastback encountered a date of 4/10/83, it would not generate an explanatory error message since this date would be an acceptable input for both the European date format and the U.S. date format.

Be aware of the date format used when the system was sysgened and be consistent when using the date component of the SINCE command.

If the month-mnemonic date format is used, Fastback will be able to interpret the desired date, regardless of the sysgened date format.

The minimum abbreviations for month-mnemonics are as follows:

JANUARY	FEBRUARY	MARCH	APRIL
MAY	JUNE	JULY	AUGUST
SEPTEMBER	OCTOBER	NOVEMBER	DECEMBER

If a BEFORE command is given without an explicit date following, the date defaults to the current date (i.e., the date of the day on which Fastback begins execution).

The time component of the time limit parameter may be expressed as:

hh:mm:ss - e.g. 15:20:55

hh:mm - e.g. 15:20

Thus the seconds field is optional and is ignored, if specified.

If the time field of a SINCE or BEFORE command is omitted, a default time of 00:00:00 (i.e., the start of the given day) is used.

Thus, if a SINCE or BEFORE command is found without both a date and time, then a date/time of 00:00:00 on the current day is assumed.

Examples:

```
BEFORE=18/APRIL/80,18:00:00

BEFO = JUNE/30/79                (default time setting)

BEFOR 9/28/59                    (U.S. date, default time)

BE,10:27                         (before 10:27:00 am "today")

BE                               (default date and time,
                                  i.e., 00:00:00 "today")
```

```
 -------------
| BUFFERSIZE |
 -------------
```

3.2.3 BUFFERSIZE

The BUFFERSIZE command allows the user to specify the desired buffer size, which determines the size of the data blocks transferred to/from tape.

Format:

$$\underline{\text{BUFFERSIZE}}[=]\left[\left\{\begin{matrix}\text{nn}\\ \text{16}\end{matrix}\right\}\right]$$

Parameters:

nn	is a decimal number determining the buffer size in kilobytes. The only allowed buffer sizes are 8, 12, 16, 20, 24, 28 or 32kb. The default buffer size is 16kb.

Functional Details:

The BUFFERSIZE command is used by backup operations to set the desired buffer size. All other operations reject this command, since the other tape related operations (i.e., restore, verifyonly and display) use information coming from the tape to select the buffer size.

If this command is omitted, a default buffer size of 16kb is used.

As the buffer size is increased:

- The tape usage efficiency increases
- The amount of data lost when an error occurs increases
- The reliability of the tape's cyclic redundancy checking decreases

Hence the user should try to set the buffer size as large as possible, bearing these points in mind. It is recommended that the selected buffer size should not exceed 16kb for phase encoded (1600 BPI) and NRZI (800 BPI) tape drives. Buffer sizes up to 32kb may be used for GCR (6250 BPI) tape drives.

Examples:

```
BUFFERSIZE=16

BUFFER      = 12

BUFF 32

BU = 24
```

3.2.4 DISC

The DISC command specifies the disc pack which is to be backed up, restored, verified, or displayed.

Format:

$$\underline{DISC}[=]\begin{Bmatrix} devn: \\ voln \end{Bmatrix}$$

Parameters:

devn:	is the device name of the disc drive containing the pack to be processed (i.e., backed up, restored, verified or displayed).
voln	is the volume (i.e., pack) name of the disc pack to be processed.

Functional Details:

A disc device mnemonic (devn:) is distinguished from a volume identifier (voln) by the presence of a colon.

The specified disc must be currently ready and marked on to the system. It must be marked on protected if a backup operation is to be performed, unless the NOPROTECT command was specified. Restore operations will execute faster if the target disc is marked on unprotected with the CDIR option.

During a restore operation the specified disc, which is the output disc, is also the default location for the building of Fastback's various control tables. This may be avoided by the use of the WORKSPACE command.

Examples:

```
DISC=DSC1:

DISC  SYS
```

3.2.5 DISPLAY

The DISPLAY command selects the display function.

Format:

DISPLAY

Parameters:

None

Functional Details:

This command informs Fastback that a display operation is to take place. The operation will involve displaying details of those files on the input medium that match any specified select criteria. Omission of any select criteria will cause the display of details pertaining to every file on the input medium. File details are displayed upon the list device.

This function may be used to display both the contents of previously created Fastback tapes, and the contents of disc packs which the user may wish to backup. The display function may also be used to check the suitability of complex select lists, by displaying those files on the input medium that match a given list.

Whenever the display function is invoked, either the TAPE or DISC command must be given. It is impossible to use both the TAPE and DISC commands at the same time since the display of files cannot be done from both input devices simultaneously.

If the input device is a tape, then tape positioning may be accomplished through the use of the POSITIONING command.

If the input device is a disc, then marking on the disc with the CDIR option does not increase the speed of the display operation.

Examples:

```
DISPLAY

DISPL

DISP
```

3.2.6 END

The END command is used to indicate the end of the commands being read from the batch command file or device.

Format:

END

Parameters:

None

Functional Details:

When this command is recognised, no attempt is made to decode any remaining commands in the current command line and no further records are read from the batch command file or device. The program then performs a consistency check on the commands entered and if no errors are found, commences the execution of the requested function.

Note that the END command is not valid if entered as part of the START command. If passed to Fastback as a START argument, the END command will be rejected.

Examples:

END

EN

3.2.7 ERRORPOOL

The ERRORPOOL command is used to specify the number of errors which Fastback can accumulate for analysis. Fastback accumulates errors relating to unreadable disc sectors and tape blocks, and nonverification of copied data.

Format:

ERRORPOOL[=] [{ entries }]
             [{ 64   }]

Parameters:

entries	is a decimal number giving the maximum number of entries that the pool can contain. Each entry requires one fullword of memory. The default number of entries is 64.

Functional Details:

This command is used by the backup, restore, and verifyonly operations. It is rejected by display operations.

If this command is not specified, then the default ERRORPOOL size is 64 entries, which requires 256 bytes.

Because Fastback processes data in physical order, when an error occurs the program cannot inform the user on which (logical) file the error occurred. The ERRORPOOL command is used to accumulate data relating to errors, so that when Fastback has completed the required operation, it can analyse the errors and report which (logical) files and records are affected.

If the error pool overflows, Fastback will only count further errors and will not be able to identify which files/records are affected by these errors.

NOTE

The user should not normally have to use this command, since the default error pool size is sufficient to hold a reasonably large number of errors. This command may need to be used only where the error pool has overflowed during an operation. The user may then wish to increase the size of the pool and rerun the operation. The user should realise that a large number of error entries may indicate a more serious problem. For example, a large number of defective sector error entries may indicate a disc hardware problem.

Examples:

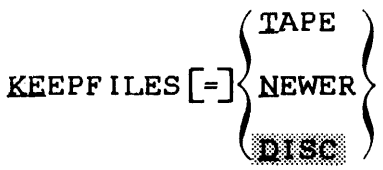
```
ERRORPOOL=64

ERRORP 128

ERRO   = 24

ER =     32

ER 64
```

```
 -------------
| KEEPFILES   |
 -------------
```

3.2.8 KEEPFILES

The KEEPFILES command is used to specify what action Fastback is to take if, when attempting to restore a file, it encounters a file with a duplicate name on the output disc.

Format:



Parameters:

TAPE	specifies that, when duplicate filenames are encountered during a restore operation, the original file on the output disc is to be deleted and replaced by the file coming from the input tape.
NEWER	specifies that, when duplicate filenames are encountered during a restore operation, whichever of the files is the newer will be retained.
DISC	specifies that, when duplicate filenames are encountered during a restore operation, the original file on the output disc is to be retained and the tape file is to be skipped. This is the default option.

Functional Details:

This command applies only to restore operations, and will be rejected if specified for a backup, verifyonly, or display operation.

When the NEWER option is invoked, the test for which file is newer is done on the basis of each file's date and time last written (i.e., updated). Should both files have identical ages, the KEEPFILES command reverts to the default option. The file on the output disc will be retained and the tape file will be skipped.

If Fastback decides that a disc file is to be replaced by a tape file (e.g., because of a duplicate filename, and KEEPFILES = TAPE), then the disc file will be deleted just before Fastback renames the appropriate intermediate file to the target (i.e., duplicate) name. Thus, if any error occurs during the execution of Fastback up to this point, the original file will still reside on the disc (i.e., the original disc file is not lost).

A consequence of this approach is that whenever KEEPFILES = TAPE is specified, the user should ensure that the output disc has sufficient space to hold all its current files, plus those that are to be restored. If this is not the case, Fastback will run out of disc space while allocating and prefilling its intermediate files. If the user believes that a disc has insufficient space to hold its current files and all Fastback's intermediate files, then some existing files on the disc should be deleted.

If the user knows that no filenames on the output disc will duplicate the names of the files to be restored, then he should specify the KEEPFILES=TAPE option to Fastback. This allows Fastback to operate slightly faster than would otherwise be the case. This is because if DISC or NEWER are specified to the KEEPFILES command, then Fastback must do a file existence check (i.e., attempt to assign any existing file of the given name) for every file it restores.

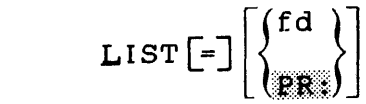
Examples:

```
KEEPFILES=DISC

KEEP = T

KE NEW

KE =D

KEEPF NE
```

3.2.9 LIST

The LIST command is used to specify the file or device to which messages are output. If this command is omitted then PR: is used as the default list device unless it cannot be assigned, in which case CON: (the device name of the system console) is assigned as the list device.

Format:



Parameters:

fd	is the file descriptor of the file or device to be used as the list device.

Functional Details:

The LIST command may be entered either as part of the operating system START command or, if batch command entry is used, as one of the commands read from the batch file. However, if batch command entry is used, then the LIST command cannot be specified as both part of the START command and one of the batch commands.

If the file descriptor specifies a disc file, then the file's account may be specified as a numeric account number or by account identifiers of P, G, or S.

If a file is specified as the list device, then it must currently exist. Output to this file will be appended after any existing data.

Note that nothing is output to the list device until all commands have been processed and validated. Thus, for example, if errors occurred whilst reading commands from a batch file, then the resulting error messages would not be output to the list device but to the system console (or MTM terminal).

Examples:

```
L PR:

LIST = D300:FBACK.LST/123

LIS FBACK.LST/P
```

| NOPROTECT |

3.2.10 NOPROTECT

The NOPROTECT command informs Fastback that the user desires to backup a disc that is not marked on protected.

Format:

NOPROTECT

Parameters:

None

Functional Details:

This command may be specified if the operation selected is a backup. If any other operation is selected, the NOPROTECT directive will be rejected.

By default, if this command is omitted, the input disc to be backed up must be marked on protected.

This command informs Fastback that the user desires to backup a disc that is not marked on protected. If the user has specified VERIFY, then, in order for Fastback to verify an unprotected disc, Fastback must be loaded with a segment size increment sufficient to hold a complete bit map for the disc plus space for the I/O buffers, error pool, etc. Fastback cannot verify an unprotected "live" disc unless it has sufficient memory for a complete disc bit map, etc. If NOPROTECT is in effect and Fastback has insufficient memory to verify the backup tape, it will generate warning messages to this effect and will not verify the backup tape. In this case it is strongly recommended that the user run a verifyonly operation after the backup is complete.

For more information on Fastback's use of a bit map, please refer to Section 5.2.1, which contains a description of how Fastback's backup operation works.

The table of segment size increments given in Section 5.5 (tuning information) outlines segment size increments required to backup different types of discs using various buffer sizes. To ensure that Fastback is loaded with a segment size increment that is sufficient to support verifying an unprotected disc, with given buffer sizes, etc., please refer to the maximum performance segment size increments given in Table 5-4. Loading Fastback with these segment size increments for each type of disc allows it sufficient memory to construct a full disc bit map.

If a backup operation is performed with NOPROTECT and VERIFY, such that Fastback determines that it cannot run a backup verify pass, then the options NOPROTECT and VERIFY=YES will be printed out on the listing created by Fastback. This is because these are the options requested by the user. However, when a restore, verifyonly, or (tape) display is run with the created tape, the options read from the tape labels will be printed as NOPROTECT and VERIFY=NO. This is because these options more correctly indicate the state of the data on the tape (i.e., the tape may not have been verified).

WARNING

WHENEVER POSSIBLE THE INPUT DISC SHOULD BE MARKED ON PROTECTED. THE ONLY TIME WHEN NOPROTECT SHOULD BE USED IS WHEN IT IS OPERATIONALLY IMPOSSIBLE TO MARK A DISC ON PROTECTED IN ORDER TO BACK IT UP. SUCH MIGHT BE THE CASE WITH SYSTEMS WHICH RUN "LIVE" 24 HOURS PER DAY.

Because Fastback backs up data in physical sector order, some peculiarities exist in verifying unprotected "live" discs. For example, if Fastback backs up a file called FRED, and then someone copies and deletes FRED before Fastback begins its verify pass, Fastback may well verify FRED because the sectors on disc which the original copy of FRED occupied have probably not yet changed, even though a whole new copy of FRED has been created. Thus Fastback will have copied the original version of FRED to tape. Similarly, if someone deleted FRED and then reused the released disc space before Fastback began to verify the backup tape, Fastback will find a large number of nonverifies.

Thus, backup nonverifies on an unprotected disc do not necessarily imply that hardware/software is operating improperly.

Examples:

NOPROTECT

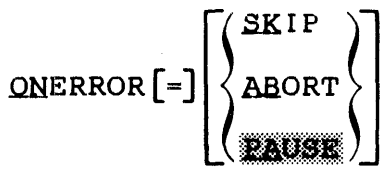
NOP

NOPRO

3.2.11 ONERROR

The ONERROR command specifies the action taken by Fastback if certain errors occur.

Format:



Parameters:

SKIP	this option requests Fastback to log an explanatory message whenever an error occurs, and then continue operation, if at all possible.
ABORT	this option requests Fastback to log an explanatory message whenever an error occurs, and then terminate.
PAUSE	this option requests Fastback to log an explanatory message whenever an error occurs, and then pause. The user may then attempt remedial action. This is the default option, should the ONERROR command be omitted.

Functional Details:

The ONERROR command may be specified for any operation.

If this command is omitted, then PAUSE is the default error action.

If ONERROR = PAUSE is specified, then whenever an error situation arises, Fastback will inform the user of the error and then pause. When the program is continued, the event which caused the error is retried.

If the SKIP option is selected, then Fastback will continue after an error occurs, whenever possible. For example, if the user has specified ONERROR=SKIP for a restore operation and the output disc fills up with files, Fastback will continue to restore whatever files it can. On the other hand, if ONERROR=PAUSE has been specified, then Fastback will pause, and the user will have to delete some files from the disc to create free space before continuing Fastback.

The ABORT option is intended for use in situations where Fastback is to execute unattended. If ABORT is specified, then Fastback will terminate on any error condition - even simple I/O errors, such as those on the list device. This option is intended for those situations where Fastback is to terminate if it cannot continue, rather than pausing whilst it has control of system resources. For example, if Fastback is run under command substitution system (CSS) control, and an error occurs, then it may be preferable for Fastback to terminate and allow the CSS to continue, rather than pause and hence suspend the CSS.

NOTE

Certain errors will not cause Fastback to pause or terminate regardless of the specified ONERROR action. The primary situation where this may occur is in the reading of data blocks from disc during backup operations. If Fastback gets (SVC 1) status X'84xx' returned by a disc read, it will locate the unreadable sector(s), add it/them to the error pool, and inform the user. Any data required from such sectors is skipped. At the end of the operation, Fastback will inform the user which files were affected (if any).

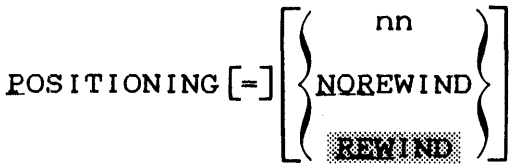
Examples:

```
ONERROR=PAUSE

ONERR = AB

ON =SKI

ON SK

ONE=     PA
```

3.2.12 POSITIONING

The POSITIONING command is used to control the initial and final positioning of the magnetic tape mounted on the specified TAPE device.

Format:



Parameters:

nn	specifies the number of filemarks which are to be skipped before the operation commences. Note that zero (0) is not acceptable - the same thing can be achieved by NOREWIND. Fastback will rewind the tape device and then forward it over the given number of filemarks. The tape will not be rewound again upon completion of the requested operation.
NOREWIND	specifies that no rewinding of the tape device is to be performed. Moreover, no other initial or final tape positioning is to be performed by the program. That is, all tape positioning is to be done by the user.
REWIND	specifies that the tape device is to be rewound before the requested operation begins and after the operation is complete. This is the default option.

Functional Details:

The POSITIONING command may be used for any operation, except display operations which involve a disc device.

If this command is omitted, REWIND is the default option.

If REWIND is specified for a backup or restore operation with the verify option enabled, Fastback will rewind the tape before the operation commences, again before the verify pass commences, and finally when the whole process is complete.

If nn is specified for a backup or restore operation with the verify option enabled, then Fastback will rewind the tape device and forward it over nn filemarks both before the operation commences and again before the verify pass is performed. At the completion of the whole operation the tape will not be rewound.

Whenever possible, the tape should not be positioned too close to the end of a reel when beginning a backup operation. The reasons for this are outlined in the description of the BACKUP command.

NOTE

> Multiple Fastback backups may be placed on a single tape. For example, backing up three discs to one tape, using the NOREWIND parameter to the POSITIONING command for each backup. Each disc backup contains six filemarks (one of the six filemarks marks the end of each backup). If you later wish to restore/verifyonly/display the nth backup on a multi-backup tape, use the POSITIONING command and specify (n-1)*6 filemarks to skip. Thus in our above case of three backups on single tape, if you wish to restore the second backup on the tape, perform a Fastback operation with POSITIONING = 6 filemarks (i.e., (2-1)*6).

Examples:

```
POSITIONING=NOREWIND

POS = 3

P REWIND

PO 7

P=REW
```

3.2.13 RESTORE

The RESTORE command is used to select the restore function.

Format:

RESTORE

Parameters:

None

Functional Details:

This command informs Fastback that a restore operation is to take place. The operation will involve restoring a previously created Fastback tape to a disc pack.

The (input) tape device specified via the TAPE command must be currently on-line, positioned according to the user's requirements, and preferably write protected. Tape positioning may be specified through the use of the POSITIONING command.

The (output) disc pack specified by the DISC command must be currently ready, on-line, and marked on unprotected.

If a disc other than the output disc is specified to the WORKSPACE command, then that disc must also be ready, on-line, and marked on unprotected.

The speed of the restore operation will be increased if the pack is marked on with the CDIR option.

If the user believes that the output disc may fill up during a restore operation, then Fastback should be started with WORKSPACE=some-other-disc and ONERROR=SKIP. Such use of the WORKSPACE command helps ensure that Fastback's control table files will not run out of space, while use of ONERROR=SKIP will cause Fastback to continue to restore whatever files it can if the disc should fill up.

If the disc fills up and ONERROR=PAUSE (the default ONERROR action), then Fastback will pause. In this case, it is up to the user to create some free space on the disc - presumably by deleting some files - before continuing Fastback.

Note that in the extraordinary circumstance where Fastback's control table files occupy a significant amount of disc space and the disc fills up during a restore, you should be able to restore your files by doing several selective restores. Each selective restore will use less disc space for control tables than would a single large restore operation, hence more disc space will be available for the files being restored.

WARNING

THE INTERMEDIATE FILES THAT FASTBACK CREATES DURING RESTORE OPERATIONS, SHOULD NEVER BE DELETED WHILE THE RESTORE OPERATION IS STILL RUNNING.

This is because Fastback uses bare-disc writes to overwrite the prefilled dummy data in (indexed-type) intermediate files, with the "real" data being restored from tape. Hence, if an intermediate file is deleted and its disc space is claimed by another file that is being written while the Fastback restore is active, then the contents of those sectors may be overwritten by Fastback restoring data from tape. Thus the file(s) which claimed the sector(s) formerly belonging to the intermediate file may have some of its/their contents destroyed.

If the file(s) that claimed the freed sectors happens to be one of Fastback's control files, then, if overwriting occurs, a Fastback internal failure may result.

For more information on Fastback intermediate files, please refer to Chapter 5 of this manual.

Examples:

RESTORE

RESTO

RES

3.2.14 SELECT

The SELECT command establishes the mode in which Fastback is to operate. In the select mode only those files on the input medium which match certain specified criteria are processed, whilst in the non-select mode all files on the input medium are processed unless either of the SINCE or BEFORE commands have been specified.

Format:

SELECT[=] { fd | @ | immed-entry }

Parameters:

fd	is the file descriptor of a file or device from which Fastback is to read select information. The file descriptor may have a numeric account number field, or may use account identifiers of P, G, or S. These latter three identifiers, when encountered, are understood to refer to the current private, group, and system accounts, respectively, when Fastback is operating in the MTM environment. Outside of the MTM environment, these identifiers refer to the system account (account zero).
@	indicates to Fastback that select criteria is to be read from the current command device/file after all commands have been read. This parameter is invalid if all commands were passed to Fastback as arguments of the OS/32 or MTM START command (since, if Fastback were "driven" by START arguments there would be no such thing as a current command device/file from which to read select entries).
immed-entry	is a single select entry enclosed in parentheses. (The format of an "immediate" select entry is discussed in Chapter 4.)

Functional Details:

If the SELECT command is omitted, Fastback defaults to operating in the non-select (i.e., process all files on input medium) mode, unless either of the (global) SINCE or BEFORE commands have been specified.

Select operations may be requested for any of the four possible functions: backup, restore, verifyonly, and display.

In the select mode, the program only processes those files on the input medium that match the given select criteria. Files may be selected for processing on the basis of their file descriptor or both their file descriptor and date/time last written (i.e., updated).

For a complete explanation of file selection criteria, please refer to Chapter 4.

Examples:

```
SELECT=SYS:Fastback.SEL/194

SEL FB.SEL/P

SELEC (-.FTN/-)

SELE=(-.TSK/194,SINCE=28/JULY/81,16:53:00)
```

3.2.15 SINCE

The SINCE command sets a global time limit on the age of files which are to be processed. Files will be processed if their date and time last written (i.e., updated) is equal to, or more recent than, the date and time specified to the SINCE command.

Format:

SINCE[=][date][,time]

Parameters:

date	is the date component of the time limit.
time	is the time component of the time limit.

Functional Details:

The SINCE command sets a global time limit value. This value is used in one of two slightly different ways, depending on whether or not Fastback is operating in select mode.

If the SELECT command is not entered, then SINCE sets a time limit which will determine which files on the input medium are to be processed. In this case, SINCE will cause the program to operate in a "pseudo select" mode where all files on the input medium will be considered for processing, but will be processed only if their date and time last written are equal to, or more recent than, the time limit specified by the SINCE command.

If the SELECT command is encountered, then SINCE sets a default global time limit for files matching the specified select criteria. Thus if a file on the input medium is to be processed, it must match a select entry and its date/time last written must be later than, or equal to, the specified global SINCE time limit. Individual select entries may contain their own SINCE (or BEFORE) commands which will override the global SINCE time limit. If both SELECT and SINCE commands are used, then, if no files on the input medium match the select criteria, no files will be processed. That is, files which fail to match select criteria will not be processed simply because they meet a global SINCE time limit.

The SINCE command (and the BEFORE command - see Section 3.2.2) may specify a date, a time, or both, which will be used in the time limit comparison.

The date component of the time limit may be given in any of the following formats:

dd/mm/yy	- e.g.	25/12/83
mm/dd/yy	- e.g.	12/25/83
dd/month-mnemonic/yy	- e.g.	25/DECEMB/83
month-mnemonic/dd/yy	- e.g.	DECEMB/25/83

If the all-numeric date format is used (i.e., dd/mm/yy or mm/dd/yy), Fastback will interpret the date according to the currently sysgened date option. For example, if Fastback encountered a date of 25/12/83, it will interpret the date as referring to the 25th of December 1983, provided the operating system was sysgened with European date format. However, if the operating system was sysgened with U.S. date format, then Fastback will declare the given date to be invalid and will generate an explanatory error message since the 25th month does not exist. On the other hand, if Fastback encountered a date of 4/10/83, it would not generate an explanatory error message since this date would be an acceptable input for both the European date format and the U.S. date format.

Be aware of the date format used when the system was sysgened and be consistent when using the date component of the SINCE command.

If the month-mnemonic date format is used, Fastback will be able to interpret the desired date, regardless of the sysgened date format.

The minimum abbreviations for month-mnemonics are as follows:

JANUARY	FEBRUARY	MARCH	APRIL
MAY	JUNE	JULY	AUGUST
SEPTEMBER	OCTOBER	NOVEMBER	DECEMBER

If a SINCE command is given without an explicit date following, the date defaults to the current date (i.e., the date of the day on which Fastback begins execution).

The time component of the time limit parameter may be expressed as:

hh:mm:ss - e.g., 15:20:55

hh:mm - e.g., 15:20

Thus the seconds field is optional and is ignored, if specified.

If the time field of a SINCE or BEFORE command is omitted, a default time of 00:00:00 (i.e., the start of the given day) is used.

Thus if a SINCE or BEFORE command is found without both a date and time, then a date/time of 00:00:00 on the current day is assumed.

Examples:

```
SINCE=30/DECEMBER/80,11:23:57

SINCE september/9/42          (default time setting)

SINCE = , 9:11                (since 09:11:00 am "today")

SI 29/7/82                    (European date with default time)

SINC                          (default date and time setting
                               i.e., 00:00:00 "today")
```

```
-------------
|    TAPE    |
-------------
```

3.2.16 TAPE

The TAPE command specifies the tape device containing the tape to be used in the backup, restore, verify, or display operation.

Format:

TAPE[=]{devn:}

Parameters:

devn:	is the device name of the tape unit containing the tape to be processed in the backup, restore, verify, or display operation.

Functional Details:

The specified tape unit must be on-line, and the appropriate tape mounted and positioned according to the user's requirements. Further tape positioning may be accomplished through the use of the POSITIONING command.

The tape may be write protected if the operation to be performed is either a restore, a verifyonly, or a display.

NOTE

The current revision of Fastback uses predefined values in most fields of the volume and header labels, etc., that Fastback writes to the tape during backup operations. Hence the tape serial number will always appear as zero (0), and the tape expiration date as 31st December 1999, etc. Thus the contents of these fields may not currently be specified by the user.

Examples:

```
TAPE=MAG1:
T = MAG1:
TA  MAG2:
```

```
 -------------
|  TEMPLATE   |
 -------------
```

3.2.17 TEMPLATE

The TEMPLATE command allows the user to specify the names of the intermediate files which Fastback will create during restore operations.

Format:

TEMPLATE[=][short-fd]

Parameters:

short-fd	is a short file descriptor consisting of a two-character name field, a standard three-character extension, and an account identifier. A volume name must not be specified in the short-fd and will be rejected if included.

Functional Details:

For an explanation of intermediate files, please refer to Section 5.2.2.

The TEMPLATE command determines the names of the intermediate files created during restore operations. The command is rejected for all other operations.

If this command is omitted, Fastback will generate intermediate files with names of:

FBxxxxxx.INT/act	where xxxxxx represents a hexadecimal number in the range 000000 to FFFFFF, and act represents the number of the account under which Fastback is operating. (The hexadecimal numbers will be consecutive, starting from zero.)

Through the TEMPLATE command, the user has the ability to specify the initial two characters of the name field, the extension, and the account number. Account identifiers of P, G and S are accepted, as are numeric account numbers.

Intermediate files are allocated, by default, on the account under which Fastback is currently executing. Intermediate files are always allocated with protection keys of (hexadecimal) FBAC.

If, when Fastback attempts to allocate an intermediate file, it discovers that a file of the same name exists, the program will increment the count part of the name field by one and try again.

Examples:

```
TEMPLATE=RI.PAT/31

TEMPL =   BP.XXX          (defaults to private account)

TEMP  IN./P               (blank extension)

TE=XX                     (blank extension, private account)

TE  NR/194                (blank extension)
```

```
 -------------
|    VERIFY   |
 -------------
```

3.2.18 VERIFY

The VERIFY command informs Fastback that once the desired backup or restore operation is accomplished, a verify pass is to be performed to check the data copied to the output medium.

Format:

VERIFY

Parameters:

None

Functional Details:

This command may be specified only if the selected operation is either a backup or restore. This command is rejected if any other function is invoked.

By default, if this command is omitted, a verify pass will not be performed.

Each reel of tape involved in a backup or restore operation is verified before the next reel of tape is requested.

If, at the time of a backup or restore operation, the user is aware that he wishes to verify the copied data, it is more efficient to specify the VERIFY command for the selected operation rather than running a later VERIFYONLY pass. This is especially true for backup operations since a verify pass, like the actual backup pass, can be done using physical record processing techniques. A VERIFYONLY pass, however, always executes like a restore, with physical to logical record mapping taking place, and thus takes somewhat longer to perform.

Examples:

VERIFY

VERI

VE

3.2.19 VERIFYONLY

The VERIFYONLY command is used to select the verifyonly function.

Format:

VERIFYONLY

or

VO

Parameters:

None

Functional Details:

This command informs Fastback that a verifyonly operation is to take place. The operation will involve verifying a previously created Fastback tape against an existing disc pack. Verifyonly operations function in a similar manner to restore operations, thus the magnetic tape device is always considered to be the input device.

The tape device specified via the TAPE command must be currently online, positioned according to the user's requirements, and preferably write protected. Tape positioning may be specified through the use of the POSITIONING command.

The disc pack specified by the DISC command must be currently ready, online, and marked on either protected or unprotected.

Marking the pack on with the CDIR option does not increase the speed of the verifyonly operation.

If, at the time of a backup or restore operation, the user is aware that he wishes to verify the copied data, it is more efficient to specify the VERIFY command for the selected operation rather than running a later VERIFYONLY pass. This is especially true for backup operations since a verify pass, like the actual backup pass, can be done using physical record processing techniques. A VERIFYONLY pass, however, always executes like a restore, with physical to logical record mapping taking place, and thus takes somewhat longer to perform.

Examples:

VERIFYO

VO

3.2.20 WORKSPACE

The WORKSPACE command instructs Fastback that, for restore and verifyonly operations, it is to build its control structures on the specified disc pack rather than on the output disc.

Format:

WORKSPACE[=]{devn: / voln}

Parameters:

devn:	is the device name of the disc drive containing the pack which is to be used to hold Fastback's data structures.
voln	is the volume (pack) name of the pack on which Fastback is to build its control structures.

Functional Details:

This command affects restore and verifyonly operations. It is rejected if specified for any other operation.

A disc device mnemonic (devn:) is distinguished from a volume identifier (voln) by the presence of a colon.

The disc identified by this command must be currently ready, online, and marked on unprotected.

If this command is omitted, workspace is obtained from the output disc.

The user may obtain a small performance improvement in restore and verifyonly operations by using the WORKSPACE command to position Fastback's data structures on another disc. This other disc should be on its own controller and selector channel if the maximum improvement from the WORKSPACE command is to be obtained.

Examples:

```
WORKSPACE=DSC1:

WORKS = DSC2:

WORKSPA SYS

WO  SCRT
```

3.3 INTERACTIVE COMMAND ENTRY

This method of command entry is provided for inexperienced and occasional users who are not familiar with (or cannot remember) the various command keywords and parameters.

The interactive command entry mode is invoked when the program is started with an interactive device as the command device. That is, the START command has one of the following formats:

```
START ,COMMAND=idev: [,LIST=fd]

START [,LIST=fd]
```

where idev: is the device name of an interactive device, and the second format results in CON: being used as the interactive command device.

When started in this mode Fastback outputs a series of prompts requesting various parameters, values, or yes/no responses. If the response to a given prompt is not valid, an error message is output and the prompt is redisplayed so that a valid response can be entered.

The majority of prompts will accept a carriage return as indication that a default value is to be used. In these cases the default response is indicated in the prompt message by a number sign character (#).

Example:

```
ACTION ON ERROR (#PAuse, SKip or ABort) ?
```

where PAUSE is the default. Note that the default value can be explicitly selected if desired, that is, the response PAUSE is valid in the above case. However, #PAUSE is not.

The prompt also shows the minimum abbreviations of the allowed keyword responses in upper case with the remainder of the keyword in lower case (provided that the terminal being used supports lower case).

The dialogue is conducted in such an order that mandatory (i.e., non-defaultable) parameters are requested first, except for the prompt for the NOPROTECT command, which is output following the FUNCTION..... prompt if the backup function is requested. Once these have been input, the user can elect to default the remaining parameters by using the response

```
!GO
```

In this case the dialogue is terminated and the requested operation commences immediately, bypassing the remaining questions. Note that the !GO response is not allowed until all the mandatory parameters have been obtained. Once this has been done, the !GO response can be given to any prompt, thus defaulting the remainder. If !GO is entered before all mandatory parameters have been obtained, an error message and indicator will be generated.

If the user wishes to change his response to a previous prompt, he can cause the complete dialogue to be restarted by entering the response

!RESTART

to any prompt.

Two other special responses are recognised and can be input for any prompt. These are

!PAUSE

and

!STOP

The !PAUSE response causes the program to be paused. When it is continued, the current prompt message is redisplayed. The !STOP response causes the program to be terminated in an orderly fashion with EOT code 250.

The following sections give the prompt messages used and the allowable responses. The first prompt requests the function to be performed. Since the dialogue varies depending on the selected function, separate sections are used to describe the conversation for each possible function.

Note that in these sections the functional details of each response are not documented since they have been given in Sections 3.2.1 through 3.2.20, and these should be consulted if any clarification is required.

3.3.1 Dialogue for the Backup Function

The first prompt is

```
FUNCTION (BAckup, REStore, DISPlay, VERIFYOnly or VO) ?
```

to which the response must be

```
BACKUP
```

to select the backup function.

If the backup function is requested, the dialogue continues with the prompt

```
ALLOW BACKUP OF UNPROTECTED DISC (#No or Yes) ?
```

The reply to this question sets the state of the NOPROTECT command. If the user wishes to backup an unprotected disc, reply Yes. If No is specified/defaulted, the input disc will have to be marked on protected.

The disc to be backed up then needs to be specified, so Fastback outputs the prompt:

```
DISC (voln or devn:) ?
```

where voln is a valid volume name, or devn: is the name of the disc drive containing the intended pack. The user should respond with either of the allowed identifiers. For example:

```
SYS             (volume name)
DSC1:           (disc drive name)
```

Fastback now requests the tape device to be identified, by prompting:

```
TAPE (devn:) ?
```

where devn: represents a tape drive name.

The user should respond by entering the device name of the tape unit.

The next prompt requests the mode in which the backup is to be performed:

```
FILE SELECTION (#ALL, @=idev:, select-fd or (select-entry)) ?
```

If all files are not to be processed then Fastback will obtain selection criteria from one of the sources identified by the user's response.

The user may respond with:

@	which requests the select list to be read from the command device/file once the main dialogue is completed.
select-fd	which is the file descriptor of a file or device containing a select list. See Chapter 4.
(select-entry)	which is an "immediate" select entry (i.e., a single select entry enclosed in parentheses).

Note that default responses can be made to both the above and all remaining prompts. Thus the special response !GO can be used to terminate the dialogue and commence execution.

The user is then asked if a verify pass is to be made of the copied data.

```
VERIFY (#Yes or No) ?
```

The user should reply either Yes or No.

Following this, the program solicits information regarding the positioning of the tape device. The following prompt is displayed:

```
TAPE POSITIONING (#REWind, NORewind or nn files to skip) ?
```

If the default is not selected, then the response must be one of the following:

REWIND

NOREWIND

nn

where nn is a decimal number.

If the operator had previously requested all files to be processed, then the following prompt is now output:

```
SELECT FILES BY AGE (#No or Yes) ?
```

On the other hand, if the user had previously indicated that file selection was desired, then this prompt is output:

```
DEFAULT FILE SELECTION AGE REQUIRED (#No or Yes) ?
```

If a Yes response is entered to either of these questions, then age data is requested using the prompt:

```
SInce date,time  or  BEfore date,time  ?
```

The response must be one of the following:

```
SINCE     date   ,time

          or

BEFORE    date   ,time
```

where the date and time format are given in Section 3.2.15.

The next prompt is:

```
ACTION ON ERROR (#PAuse, SKip or ABort) ?
```

The user should respond to this question by entering one of the indicated options. These options are explained in Section 3.2.11.

The buffer size is set in response to the following prompt, to which the reply must be either the default or one of the indicated values:

```
BUFFER SIZE IN KB (#16, 8, 12, 20, 24, 28 or 32) ?
```

The next prompt is:

NUMBER OF ENTRIES IN ERROR POOL (#64 or nn) ?

If a default is not used, a decimal number must be entered.

If the list device was not specified in the START command the following prompt is displayed:

LIST DEVICE (#PR:, @=idev:, or FD) ?

The default (null) response will select PR: as the list device. A response of @ will select the user's terminal as being the list device. Alternatively any required file or device can be selected by entering its file descriptor. Note that if a file is specified it must currently exist and that the listing information will be appended to any existing data in the file.

At this point all required data has been entered and Fastback will commence the requested operation.

3.3.2 Dialogue for the Restore Function

The first prompt is

FUNCTION (BAckup, REStore, DISPlay, VERIFYOnly or VO) ?

to which the response must be

RESTORE

to select the restore function.

If the restore function is requested, the dialogue continues with the prompt

DISC (voln or devn:) ?

where voln is a valid volume name, or devn: is the name of the disc drive containing the intended pack. The user should respond with either of the allowed identifiers. For example:

SYS	(volume name)
DSC1:	(disc drive name)

Fastback now requests the tape device to be identified, by prompting:

```
TAPE (devn:) ?
```

where devn: represents a tape drive name.

The user should respond by entering the device name of the tape unit.

The next prompt requests the mode in which the restore is to be performed, and is as follows:

```
FILE SELECTION (#ALL, @=idev:, select-fd or (select-entry)) ?
```

If all files are not to be processed, then Fastback will obtain selection criteria from one of the sources identified by the user's response.

The user may respond with:

@	which requests the select list to be read from the command device/file once the main dialogue is completed.
select-fd	which is the file descriptor of a file or device containing a select list. See Chapter 4.
(select-entry)	which is an "immediate" select entry (i.e., a single select entry enclosed in parentheses).

Note that default responses can be made to both the above and all remaining prompts. Thus the special response !GO can be used to terminate the dialogue and commence execution.

The user is then asked if a verify pass is to be made of the copied data:

```
VERIFY (#Yes or No) ?
```

The user should reply either Yes or No.

Following this, the program solicits information regarding the positioning of the tape device. The following prompt is displayed:

```
TAPE POSITIONING (#REWind, NORewind or nn files to skip) ?
```

If the default is not selected, then the response must be one of the following:

REWIND

NOREWIND

nn (where nn is a decimal number)

If the operator had previously requested all files to be processed, then the following prompt is now output:

SELECT FILES BY AGE (#No or Yes) ?

On the other hand, if the user had previously indicated that file selection was desired, then this prompt is output:

DEFAULT FILE SELECTION AGE REQUIRED (#No or Yes) ?

If a Yes response is entered to either of these questions, then age data is requested using the prompt:

SInce date,time or BEfore date,time ?

The response must be one of the following:

SINCE date ,time

or

BEFORE date ,time

where the date and time format are given in Section 3.2.15.

The next prompt is:

ACTION ON ERROR (#PAuse, SKip or ABort) ?

The user should respond to this question by entering one of the indicated options. These options are explained in Section 3.2.11.

In order to ascertain what is to be done if duplicate filenames are encountered, Fastback then asks the question:

```
IF FILENAME CONFLICT, KEEP (#DISC, Tape or NEwer) FILE(S) ?
```

The default action is to retain the original disc file and skip the tape file. Alternatively, the Tape option would cause the original disc file to be deleted and replaced by the file coming from the input tape, whereas the NEwer option would only delete and replace the disc file if it was found to be older than the tape file.

The user may direct Fastback to build its required data structures on some disc other than the (default) output disc. This may be done by the response given to the following prompt:

```
USE WORKSPACE ON (#ovol, voln or devn:) ?
```

where ovol is the volume name of the output disc pack.

The default location for restore workspace is on the current (output) disc. However, by specifying another disc volume or drive, the user can move the work area on to another pack.

The program then gives the operator the option of specifying the names of the intermediate files that Fastback will create on the output disc. The prompt for this is:

```
INTERMEDIATE FILENAME TEMPLATE (#FB.INT/0 or fn.ext/act) ?
```

The user may either retain the default file descriptor or specify a "short" file descriptor. Account identifiers of P, G, and S are accepted. Given the answer to this question Fastback will generate intermediate filenames of the form discussed in Section 3.2.17.

The next prompt is:

```
NUMBER OF ENTRIES IN ERROR POOL (#64 or nn) ?
```

If a default is not used, a decimal number must be entered.

If the list device was not specified in the START command the following prompt is displayed:

```
LIST DEVICE (#PR:, @=idev:, or FD) ?
```

The default (null) response will select PR: as the list device. A response of @ will select the user's terminal as being the list device. Alternatively any required file or device can be selected by entering its file descriptor. Note that if a file is specified it must currently exist. The listing information will be appended to any existing data in the file.

At this point all required data has been entered and Fastback will commence the requested operation.

3.3.3 Dialogue for the Display Function

The first prompt is

```
FUNCTION (BAckup, REStore, DISPlay, VERIFYOnly or VO) ?
```

to which the response must be

```
DISPLAY
```

to select the display function.

If the display function is requested, the dialogue continues with the prompt

```
DEVICE (voln or devn:) ?
```

where voln is a valid volume name, or devn: is the name of a disc or tape drive. The user should respond with either of the allowed identifiers. For example:

```
SYS             (volume name)

DSC1:           (disc drive mnemonic)

MAG1:           (tape drive mnemonic)
```

The next prompt requests the mode in which the display is to be performed, and is as follows:

```
FILE SELECTION (#ALL, @=idev:, select-fd or (select-entry)) ?
```

If all files are not to be processed, then Fastback will obtain selection criteria from one of the sources identified by the user's response.

The user may respond with:

@	which requests the select list to be read from the command device/file once the main dialogue is completed.
select-fd	which is the file descriptor of a file or device containing a select list. See Chapter 4.
(select-entry)	which is an "immediate" select entry (i.e., a single select entry enclosed in parentheses).

Note that default responses can be made to both the above and all remaining prompts. Thus the special response !GO can be used to terminate the dialogue and commence execution.

Following this, the program solicits information regarding the positioning of the tape device. The following prompt is displayed if the device specified to the DEVICE command was a magnetic tape unit:

```
TAPE POSITIONING (#REWind, NORewind or nn files to skip) ?
```

If the default is not selected, then the response must be one of the following:

REWIND

NOREWIND

nn (where nn is a decimal number)

If the operator had previously requested all files to be processed, then the following prompt is now output:

```
SELECT FILES BY AGE (#No or Yes) ?
```

On the other hand, if the user had previously indicated that file selection was desired, then this prompt is output:

```
DEFAULT FILE SELECTION AGE REQUIRED (#No or Yes) ?
```

If a Yes response is entered to either of these questions, then age data is requested using the prompt:

```
SInce date,time  or  BEfore date,time  ?
```

The response must be one of the following:

```
SINCE     date    ,time

          or

BEFORE    date    ,time
```

where the date and time format are given in Section 3.2.15.

If the list device was not specified in the START command, the following prompt is displayed:

```
LIST DEVICE (#PR:, @=idev:, or FD) ?
```

The default (null) response will select PR: as the list device. A response of @ will select the user's terminal as being the list device. Alternatively any required file or device can be selected by entering its file descriptor. Note that if a file is specified, it must currently exist. The listing information will be appended to any existing data in the file.

At this point all required data has been entered and Fastback will commence the requested operation.

3.3.4 Dialogue for the Verifyonly Function

The first prompt is

```
FUNCTION (BAckup, REStore, DISPlay, VERIFYOnly or VO) ?
```

to which the response must be

```
VERIFYONLY

 or

VO
```

to select the verifyonly function.

If the verifyonly function is requested, the dialogue continues with the prompt

```
DISC (voln or devn:) ?
```

where voln is a valid volume name, or devn: is the name of the disc drive containing the intended pack. The user should respond with either of the allowed identifiers. For example:

```
SYS             (volume name)

DSC1:           (disc drive name)
```

Fastback now requests the tape device to be identified, by prompting:

```
TAPE (devn:) ?
```

where devn: represents a tape drive name.

The user should respond by entering the device name of the tape unit.

The next prompt requests the mode in which the verifyonly is to be performed, and is as follows:

```
FILE SELECTION (#ALL, @=idev:, select-fd or (select-entry)) ?
```

If all files are not to be processed, then Fastback will obtain selection criteria from one of the sources identified by the user's response.

The user may respond with:

@	which requests the select list to be read from the command device/file once the main dialogue is completed.
select-fd	which is the file descriptor of a file or device containing a select list. See Chapter 4.
(select-entry)	which is an "immediate" select entry (i.e., a single select entry enclosed in parentheses).

Note that default responses can be made to both the above and all

remaining prompts. Thus the special response !GO can be used to terminate the dialogue and commence execution. Following this, the program solicits information regarding the positioning of the tape device. The following prompt is displayed:

TAPE POSITIONING (#REWind, NORewind or nn files to skip) ?

If the default is not selected, then the response must be one of the following:

REWIND

NOREWIND

nn

where nn is a decimal number.

If the operator had previously requested all files to be processed, then the following prompt is now output:

SELECT FILES BY AGE (#No or Yes) ?

On the other hand, if the user had previously indicated that file selection was desired, then this prompt is output:

DEFAULT FILE SELECTION AGE REQUIRED (#No or Yes) ?

If a Yes response is entered to either of these questions, then age data is requested using the prompt:

SInce date,time or BEfore date,time ?

The response must be one of the following:

SINCE date ,time

or

BEFORE date ,time

where the date and time format are given in Section 3.2.15.

The next prompt is:

```
ACTION ON ERROR (#PAuse, SKip or ABort) ?
```

The user should respond to this question by entering one of the indicated options. These options are explained in Section 3.2.11.

The user may direct Fastback to build its required data structures on some disc other than the (default) output disc. This may be done by the response given to the following prompt:

```
USE WORKSPACE ON (#ovol, voln or devn:) ?
```

where ovol is the volume name of the output disc pack.

The default location for restore workspace is on the current (output) disc. However, by specifying another disc volume or drive, the user can move the work area on to another pack.

The next prompt is:

```
NUMBER OF ENTRIES IN ERROR POOL (#64 or nn) ?
```

If a default is not used, a decimal number must be entered.

If the list device was not specified in the START command, the following prompt is displayed:

```
LIST DEVICE (#PR:, @=idev:, or FD) ?
```

The default (null) response will select PR: as the list device. A response of @ will select the user's terminal as being the list device. Alternatively any required file or device can be selected by entering its file descriptor. Note that if a file is specified it must currently exist. The listing information will be appended to any existing data in the file.

At this point all required data has been entered and Fastback will commence the requested operation.

CHAPTER 4
FILE SELECTION

4.1 GENERAL INFORMATION

When running Fastback, the user can request that either all files on the input medium be processed or that only certain files be selected. That is, Fastback can operate in either normal or select mode. In select mode the files to be processed can be selected either by name, by date/time last written, or a combination of the two.

If the file selection is to be done purely on the basis of date/time last written, then this may be achieved by using the SINCE or BEFORE commands as described in Sections 3.2.15 and 3.2.2.

If file selection by name is required, then the SELECT command (see Section 3.2.14) must be used. This being the case, the decision as to whether or not to process each individual file on the input medium is made by comparing each file's file descriptor against a set of selection criteria (called a select list) supplied by the user. Note that, in the case of a verifyonly operation, the input medium is always considered to be the tape device.

The select list is made up of one or more select entries. In general, these are read in from the file or device specified by the SELECT command. However, if only one select entry is required, this can be specified in parentheses as an argument to the SELECT command.

Example:

```
SELECT = (-.TSK/194)
```

This is referred to as the immediate select feature, and an entry specified in this manner is called an immediate select entry.

When the select list is read from a file or device, each input record may contain multiple select entries provided that each entry is separated by semicolons.

Example:

```
SEL-.LST/- ; EXAMPLE.*A*/1-      ;entry-.sel/194
```

An entry cannot be continued across multiple records.

A select entry may take the form of a comment. These are denoted by a leading double-quote (") character, and, once this character is recognized, Fastback treats the remainder of the current select input record as a comment.

Example:

```
COMMENT.EX1/-   ;" First select record - this is a comment

" This whole select record is a comment.

-.-/194         ;" Next select entry ignored ==> ; -.-/200
```

However, comments are not allowed in immediate select entries.

The reading of the select list is terminated when one of the following conditions is met:

- The characters /* or ./ appear in the first two columns of a record.
- End of file (X'88') status is returned.
- End of medium (X'90') status is returned.

Each select entry must contain a "mask" against which the file descriptors of the files on the input medium are compared. Select entries may also contain an exclusion indicator, a date/time limit, a reprotect command, and a rename command.

The rename and reprotect commands are used to specify the new names and keys for any files which match the selection criteria.

The date/time limit is used to provide for files which match the mask, an override to any (global) date/time limit set by the BEFORE or SINCE commands.

The use of the exclusion indicator is best explained as follows. Each file on the input medium is tested against each successive select entry in turn. If the file matches the selection criteria (i.e., both the mask and any date/time limit), it is said to be "included" unless the select entry contains the exclude indicator in which case it is said to be "excluded". Thus, as the file is checked against each successive select entry, whenever matching occur the file may switch from "included" to "excluded" or vice versa. The file will be processed if, after it has been compared with the last select entry, it is still "included".

It should be apparent from the above discussion that the ordering of the select entries is significant if exclusion indicators are being used. If a complex select list is being used, the user can quickly check its suitability by using the DISPLAY function before running the actual backup or restore operation.

4.2 SELECT ENTRY FORMAT

Each select entry has the general format:

[!] mask [,date/time limit [,reprotect [,rename]]]

where the character ! is the exclusion indicator.

The other parameters are discussed in the following sections.

Apart from the exclusion indicator and the mask, the parameters are keyword parameters and may be specified in any order following the mask.

4.2.1 Select Masks

The mask in the select entry has the same general format as a standard file descriptor. That is, it has the general format:

voln:filename.ext/act

Where:

voln	is the volume name. A volume name may either be specified or omitted from the mask. See Section 4.2.2.
filename	is the name of the file. The filename field should always be specified in the select mask.

ext — is the filename extension. This field may be omitted from the select mask and, if so, the extension field of the mask defaults, allowing any three valid characters equivalent to a "-" character in the extension field.

act — is the number of the account to which the file belongs. Fastback will also accept account specifiers of P, G and S. When running in the MTM environment, these are understood to refer to the current private, group, and system accounts, respectively. Outside of the MTM environment, Fastback will assume that these account identifiers refer to account zero (the system account). This field may be omitted from the select mask, and, if so, the account number defaults to the current private account. The current private account is the system account when Fastback is executing outside of the MTM environment.

Both upper and lower case alphabetic characters may be used in file descriptor masks.

Also, space characters are ignored whenever they are encountered within the select mask. For example:

FASTBACK.TSK/0 is equivalent to FAS T BACK . TSK /0

FAST-.TSK/- is equivalent to F A ST - .TSK/ -

Whilst the mask can consist simply of a standard file descriptor (e.g., FASTBACK.TSK/0), it may also contain certain special characters which are used to represent various ranges of alphanumeric characters. By using these special characters, a single mask can be used to obtain a positive match against a number of filenames and thus select multiple files. These special characters are discussed below.

- The character - is used to specify that the remaining part of the field in which it occurs may contain any valid characters.

Examples:

MASK	MATCHES
-.TSK/112	any file with extension .TSK and account number of 112.
F-.OBJ/-	any file with extension .OBJ whose name starts with F.
XYZ.A-/1-	all files named XYZ, whose extension field begins with A, provided that the account number is between 10000 and 19999.

- The character * is used to specify that any valid character (including space) is allowed in the position occupied by the * character within the file descriptor.

Examples:

MASK	MATCHES
**XYZ.TSK/112	any file with a name in the range A0XYZ through ZZXYZ, with the extension .TSK and account number 112.
ABC.***/23	all files named ABC in account 23, irrespective of extension. This is equivalent to ABC.-/23.
XYZ.TSK/11*	all files with name and extension of XYZ.TSK residing on accounts 110 through 119.
AL*.TSK/92	all files under account 92 with extension .TSK, whose name can be simply AL or any name in the range AL0 through AL9 or ALA through ALZ.

- The character % is used to specify that any valid alphanumeric character except space is allowed in the position occupied by the % character within the file descriptor.

Examples:

MASK	MATCHES
AL%.TSK/92	any file under account 92 with extension .TSK, whose name is in the range AL0 to AL9 or ALA through ALZ.

NOTE

A file whose name is simply AL is not acceptable. Such a file would be acceptable if the * mask character had been used instead of the % character.

ID-.%%%/-	all files whose names begin with ID and do not have a blank extension.
XYZ.TSK/11%	all files with name and extension of XYZ.TSK residing under accounts 110 through 119.

- The character $ is used to specify that any alphabetic character is allowed in the position occupied by the $ character within the file descriptor. Note that the space character is not accepted as an alphabetic character.

NOTE

This character cannot be used in the account number field, since it represents all alphabetic characters. The only alphabetic characters allowed in this field are P, G, and S. Fastback recognizes that these characters imply specific account numbers.

Examples:

MASK	MATCHES
PRH.T$K/3	any file with name PRH existing on account 3, whose extension is in the range .TAK through to .TZK.
PMB$$$$$.TXT/-	any file with the extension .TXT under any account, whose name consists of PMB followed by 5 alphabetic characters.

- The character # is used to specify that any numeric character is allowed in the position occupied by the # character within the file descriptor. Note that the space character is not accepted as a numeric character.

NOTE

This special character is illegal if it is used as the first character of the volume or name fields of the file descriptor. These fields must begin with an alphabetic character, and thus an implied numeric character is unacceptable.

Examples:

MASK	MATCHES
JTG.SC#/G	all files under the current group account whose names are JTG and whose extensions are made up of the characters SC followed by any character in the range 0 to 9.
J#G.SC2/-	any file with an extension of SC2, existing under any account, and whose name consists of a J and a G separated by a single digit. Thus the name field may take any value in the range J0G to J9G.
#TG.SC2/-	is illegal since an implied numeric character is unacceptable as the first character of the name or volume field of the file descriptor.

4.2.2 Volume Name Masking

The volume name may, if desired, be specified in the select mask. If it is specified, then it may contain any of the special characters discussed in Section 4.2.1.

If the input medium is a tape, volume names in select masks are totally ignored.

If the input medium is a disc, any volume names in select masks must match the name of the input disc pack.

Explicit volume names in select lists may be used to ensure that individual select lists refer to specific disc packs. Thus the user could always be certain that a correct select list is being used for a backup or disc display operation.

Examples:

VOLUME MASK	REMARKS
SYS:	must match the name of the input disc pack if the operation being performed is a backup or a display of a disc.
S$S*:	must match the name of the input disc pack if the operation being performed is a backup or a display of a disc.
-:	is valid for all operations.

4.2.3 Account Number Masking

As part of each file descriptor mask, Fastback will accept either a specific account number, an account number mask, or an account identifier of P, G, or S.

If a specific account number is used, the program will process only those files which belong to the given account, and whose names and extensions match the appropriate fields of the file descriptor mask.

An account number mask can be used to process files which belong to a whole set, or family, of accounts.

If the account field is omitted from a select file descriptor mask, Fastback will default the account number to the current private account. This will be zero when the program is running outside the MTM environment.

Account numbers, when specified, may contain leading zeroes and/or imbedded spaces. Account numbers may contain up to three significant digits.

Examples:

MASK	MATCHES
-.-/1**	any file on the input medium whose account is in the range 100 through 199 inclusive.
-.***/####7	any file on the input medium whose (5 digit) account number ends with a 7 (e.g., 007, 017,...,147,...,247, ...,1007,...,65527).
-/%	all files on the input device with a single digit account number (in the range 0 to 9 inclusive).
FASTBACK.TSK/2-	any file on the input medium with a name and extension of FASTBACK.TSK residing on any account from 20000 to 29999 inclusive.
XYZ.-/$	is invalid since the $ character cannot be used to imply P, G, or S in the account number field.
RKI-.FTN/7****	is invalid since the maximum allowed account number is 65535; a 5 digit account specifier beginning with a 7 cannot be used.
nerang.***/6554#	is invalid since the maximum allowed account number is 65535; a 5 digit account number specifier beginning with 6554 (or greater) cannot be used.

4.2.4 Complex Masks

The masking abilities outlined in the preceding sections may be combined to form very powerful file descriptor masks, as shown below. Caution should be exercised when constructing complex masks since the resulting matches may not be readily apparent. Hence it may be preferable to use a number of simple masks rather than a single complex one.

If the user is unsure of the effect that the desired select list will have, use the display function to check the results that will be obtained from the use of the given select list.

Examples:

```
J$M.$$$/#           DEK.%%%/%#4          F$S%B*C-.$$$/19-
!-.HLP/-            -:DISCVOLN.-/***     $#%-.$%#/1*4
!****:%%%%.-/##1    !$%%%:EXAMPLE/-      -/-
lower.cas/***       !-/***               -.-/P
!-/G                -.***/s              -/**4
```

4.2.5 Date/Time Limit

The date/time limit parameter is used to specify a test on the age of each file to further qualify the select mask. The age of the file is determined by the date/time it was last updated, and the time limit parameter may indicate those files last updated either BEFORE or SINCE the given date and time.

Format:

```
 (  >          )
 ( SINCE  [=]  )
 (             )  [date] [,time]
 ( BEFORE [=]  )
 (  <          )
```

Parameters:

date	is the date component of the time limit.
time	is the time component of the time limit.

Functional Details:

A date/time limit contained within a select entry overrides any active global time limit (as specified by a SINCE or BEFORE command) for those files whose file descriptors match the mask in the select entry.

The SINCE (or >) date/time parameter is used to indicate that those files whose names conform to the given mask, and whose date/time last written is equal to or more recent than the SINCE date/time, are considered to match the select entry.

Conversely, the BEFORE (or <) date/time parameter is used to indicate that those files whose names conform to the given mask, and whose date/time last written is older than the BEFORE date/time, are considered to match the select entry.

The SINCE or BEFORE parameter may specify a date, a time, or both, which will be used in the time limit comparison.

The date component of the time limit may be given in any of the following formats:

dd/mm/yy	- e.g.,	25/12/83
mm/dd/yy	- e.g.,	12/25/83
dd/month-mnemonic/yy	- e.g.,	25/DECEMB/83
month-mnemonic/dd/yy	- e.g.,	DECEMB/25/83

If the all numeric date format is used (i.e., dd/mm/yy or mm/dd/yy), Fastback will interpret the date according to the currently sysgened date option. For example, if Fastback encountered a date of 25/12/83, then it will interpret the date as referring to the 25th of December 1983, provided the operating system was sysgened with European date format. However, if the operating system was sysgened with U.S. date format, then Fastback will declare the given date to be invalid and will generate an explanatory error message since the 25th month does not exist. On the other hand, if Fastback encountered a date of 4/10/83, it would not generate an explanatory error message since this date would be an acceptable input for both the European date format and the U.S. date format.

Be aware of the date format used when the system was sysgened and be consistent when using the date component of the SINCE command.

If the month-mnemonic date format is used, Fastback will be able to interpret the desired date, regardless of the sysgened date format.

The minimum abbreviations for month-mnemonics are as follows:

JANUARY	FEBRUARY	MARCH	APRIL
MAY	JUNE	JULY	AUGUST
SEPTEMBER	OCTOBER	NOVEMBER	DECEMBER

If a SINCE or BEFORE parameter is given without an explicit date following, the date defaults to the current date (i.e., the date of the day on which Fastback begins execution).

The time component of the time limit parameter may be expressed as:

hh:mm:ss - e.g., 15:20:55

hh:mm - e.g., 15:20

Thus the seconds field is optional and is ignored if specified.

If the time field of a SINCE or BEFORE parameter is omitted, a default time of 00:00:00 (i.e., the start of the given day) is used.

Thus if a SINCE or BEFORE parameter is found without both a date and time, then a date/time of 00:00:00 on the current day is assumed.

Examples:

BEFORE september/9/42 , 11:23:57

SINCE=30/decem/80 (default time setting)

< , 9:11 (before 09:11:00 am "today")

>23/7/81 (default time setting)

>,4:4:4 (before 04:04:04 am "today")

4.2.6 Reprotect

Files which match a select entry's mask and (optional) time limit criteria may have their protection keys altered by using the KEYS parameter.

Format:

KEYS [=] [wwrr]

Parameters:

ww	are two hexadecimal digits representing the new write protection keys.
rr	are two hexadecimal digits representing the new read protection keys.

Functional Details:

If a value is omitted from the KEYS parameter, the file's keys will be set to zero (0000).

Note that if a file matches multiple select entries, reprotection is governed by the select entry against which the file last matches.

Although the KEYS parameter may be used within a select entry which contains an "exclude" mask, it makes no sense to do so since any files matching an exclusion mask are not processed and therefore cannot have their keys changed.

Examples:

```
KEYS=FF00

key AABB

key                          (defaults to 0000)
```

4.2.7 Rename

Files which match a select entry's mask and (optional) time limit criteria may have their names altered through the use of the RENAME parameter. The RENAME facility may be used to avoid file name clashes during a restore operation, or it may be used to alter the names and accounts of files for distribution purposes, etc.

Format:

RENAME [=] new-mask

Parameters:

new-mask	is generally allowed to have the same format as the select entry's file descriptor mask.

Functional Details:

Although the format of the new-mask may be the same as the select entry's mask, the meaning of certain special mask characters is redefined. Within the RENAME mask the special characters %, $, and # are all taken to mean the same as the * character. That is, each of these special characters may be used to represent any valid character (including a space) in the file descriptor in the position denoted by the special character.

Wherever a mask character appears in a RENAME file descriptor, it means "please transfer whatever character appears in this position within the current filename to the corresponding position in the new filename." In other words, keep one or more characters from the "old" filename, and use it/them in the "new" filename.

NOTE

If a file matches multiple select entries, its renaming is governed by the select entry against which the file last matches.

Although the RENAME command may be used within a select entry which contains an "exclude" mask, it makes no sense to do so since any files matching an exclusion mask are not processed and therefore cannot be renamed.

NOTE

Use of the RENAME facility may result in duplicate file descriptors being created for different files. When performing a backup operation, Fastback cannot detect files with identical names being written to tape; hence the user should exercise caution when using complex rename masks during a backup. It should also be noted that renaming during a backup may complicate later verifyonly operations since such operations always use the tape as the input medium. In such a case, the renames performed during the backup would have to be "reversed" by the select list in a verifyonly operation.

It is thus recommended that renaming is generally best performed during a restore operation.

Table 4-1 shows what happens when Fastback encounters an invalid or duplicate file descriptor.

TABLE 4-1 INVALID OR DUPLICATE FILE NAMES: ACTION TAKEN

OPERATION	INVALID FD.	DUPLICATE FD.
Backup	Original fd retained	Not detectable
Restore	Intermediate file left	Determined by KEEPFILES
Verifyonly	File not processed	File is verified

For a given select file descriptor mask, not all possible RENAME transformations are allowed since some may lead to the creation of invalid (or duplicate) file descriptors. When Fastback reads in its select list, it checks any rename parameters against their corresponding file selection mask in each select entry in order to discover possible illegal filename transformations. Those transformations which are deemed illegal are:

1. Any transformation which maps an implied space character in the file selection mask onto an alphanumeric character in the rename mask, where the previous character in the select mask was an implied space or a "don't care" (*) mask, and the previous character in the restore mask was a "don't care" (*, %, #, or $) mask.

Example:

```
RKI.AUS/-,RENAME = RKI**BSD.USA/-
```

The above rename is illegal because filenames which result from the renaming would have the form RKI^^BSD.USA (where ^ represents a space character). This filename contains imbedded spaces, and hence is invalid. The select entry in this example would be rejected because it uses the illegal transformation described above.

2. Any transformation which maps a "don't care" (*) character in the file selection mask onto an alphanumeric character in the rename mask, where the previous character in the select mask was an implied space or a "don't care" (*) mask, and the previous character in the rename mask was a "don't care" (*, %, #, or $) mask.

Example:

```
WJB-.AUS/53,RENAME WJB*SYN.-/#
```

The above example is illegal because it uses transformation (2) above. The select file mask could match files whose name field was WJB^^^^ (where ^ represents a space character). Such a match, under the above transformation, would result in the attempted creation of files with names WJB^SYN (where ^ represents a space character), which are invalid.

The renaming restrictions enforced by Fastback are aimed at preventing the creation of invalid/duplicate file descriptors. Removal of the restrictions would render it easy to create file descriptors with imbedded spaces, such as SCR^^ABC.USA/222 (where ^ represents a space character), and other illegal file descriptors.

Some examples of possible rename combinations follow.

Example:

```
SCR9876.AUS/100 , RENAME = XYZ1234.USA/222
```

This example illustrates the alphanumeric to alphanumeric character transformation. Each (alphanumeric) character in the name of the selected file is changed to a different character in the desired file descriptor.

Example:

```
SCR-.AUS/100,REN=-.USA/222
```

The above example shows the alphanumeric to special character relationship. The name field of the input file will be retained when the file is renamed. The extension and account fields, however, will be altered according to the corresponding fields in the rename mask.

Example:

```
SCR###.$$$/*****  , RENA -.$$$/222
```

This example illustrates both the special character to special character mapping and the special character to alphanumeric character mapping. Files selected by the initial mask will have their names and extensions retained by the rename procedure, since the special characters in the name and extension fields of the rename mask request the characters from the original file descriptor to be retained. The special character to alphanumeric character mapping is shown by the account number fields. The original account number may be any valid account, but during the rename it is forced to account number 222.

Example:

```
SCR-.-/-,RENAME=SCR**ABC.-/-
```

The above select entry would be rejected because it uses the illegal transformation given as illegal transformation (2), in the earlier discussion.

Example:

```
SCR.-/- , RENAM SCR%%ABC.-/-
```

This select entry would also be rejected. This arises from the fact that the implied space to alpanumeric character mapping is illegal. This is outlined as illegal transformation (1) in the preceding discussion.

Example:

```
SCR-.-/- ,RENAM= -.-/655**
```

This final example shows a select entry that would be accepted. However, caution should be exercised when using the above type of rename mask, since a file descriptor whose account number is, for instance, 177 will become invalid after renaming because its account number will be 65577 (i.e., exceeds 65535).

4.3 EXAMPLES OF VALID SELECT ENTRIES

The following examples show the use of all parameters available in a select entry.

```
BDV-.$$$/##4

!F$%*BACK.-/00#

-.-/-,BEFORE=16/JULY/83,13:04:59,KEYS=FF00,RENAME=/111

/194,>,8:00,ren/199                    (all files under account
                                       194, written at or after
                                       8am "today", renamed to
                                       account 199)

!-.dir/0

-/P,sin9/9/83,keys,renam/g             (all private files,
                                       written on and after
                                       9/9/83, keys changed to
                                       0000, renamed to group
                                       account)

DRING-.-/-;BOCTOR.WAG/153;ID-.dmu/-    (multiple select
                                       entries on a line)
```

4.4 PROCESSING OF SELECT ENTRIES

The following discussion assumes that the reader is familiar with the general format of file selection entries, as described in the preceding sections.

The order in which select entries are read by Fastback may play a significant role in determining which files are processed and which are not. When Fastback reads a select entry, it encodes the entry and builds it into an internal table of select entries. The entries in the table are held in the same order in which those entries were presented to (read by) Fastback. For example, the select list:

```
FRED.XMP/-
ma-.xmp/-
dummy.-/-    ; !-/194 ; ALL.-/-
-.XMP/-
%%%%BACK/-
/*                          (end of select list)
```

would be encoded by Fastback, and held internally in the sequence that follows.

```
FIRST   :  FRED.XMP/-
SECOND  :  ma-.xmp/-
THIRD   :  dummy.-/-
FOURTH  :  !-/194
FIFTH   :  ALL.-/-
SIXTH   :  -.XMP/-
SEVENTH:   %%%%BACK/-
```

Files from the input medium are processed against the select list as follows. Each file on the input medium is tested against each successive select entry in turn. The file will either match the selection criteria (i.e., both the mask and any date/time limit) or will not. If it matches, the file is said to be "included" unless the select entry contains the exclude indicator, in which case it is said to be "excluded". Thus, the file is checked against each successive select entry, and, when matching occurs, the file may switch from "included" to "excluded" or vice versa. The file will be processed if, after it has been compared with the last select entry, it is still "included".

To further illustrate this procedure, follow what happens when the file MATCH.XMP/194 is processed against the select list illustrated above. Note that, initially, the file is in an "excluded" state.

MASK	EXCLUDE INDICATOR	MATCH	REMARKS
FRED.XMP/-	No	No	File still excluded
ma-.xmp/***	No	Yes	File included
dummy.-/-	No	No	File still included
!-/194	Yes	Yes	File excluded
ALL.-/###	No	No	File still excluded
-.XMP/%%%	No	Yes	File included
%%%%BACK/-	No	No	File still included
/*	(end of list)		RESULT: FILE INCLUDED

Having been compared against every entry in the select list, the file MATCH.XMP/194 will be selected for processing. This is because the filename's last successful match in the select list was against an INCLUDE mask.

There is one other aspect of select lists of which the user should be aware. This involves the special case of a select list which contains only a single entry, and that entry has an exclude indicator. Once selective operation has been requested, Fastback will initially assume that no files on the input medium are to be processed. Moreover, when a lone select entry indicates that some subset of the input files are not to be processed, the overall effect is that no files on the input medium will be processed. Thus if the user desires that all files on the input medium should be copied/displayed, with the exception of some subset, then the following select criteria should be used:

```
-.-/-           = process all files
!SUBSET-.-/-    = except this subset
/*              (end of select list)
```

Even in the above simple example the user should be aware of the significance of the ordering of the two select entries. Reversal of the order would result in the processing of all files on the input medium.

The reader should also realise that, because select entries are scanned sequentially, any select entry that is entered incorrectly by the user may be "cancelled" by the use of an exclude indicator (!) combined with a select entry which is identical to the incorrect entry.

Example:

```
JOE.-/194       = select entry for JOE. under acct 194
FERD.-/194      = incorrect select entry (should have been
                  FRED.-/194)
!FERD.-/194     = don't select files called FERD.-/194
                  (i.e., omit the incorrect entry by using
                  the exclude indicator (!))
FRED.-/194      = the correct entry (what was originally meant
                  to be entered)
PETER.-/3       = the next select entry
/*              = end of select list
```

4.5 NON-MATCHED MASKS

Whenever Fastback is run with the SELECT option, it displays those select entries which never matched any file on the input medium. This is done when the program has finished scanning the input medium to determine which files are to be processed.

Because Fastback encodes select masks into an internal format as soon as they are encountered, whenever it needs to display a non-matched mask, the mask may not be displayed in the same format in which it was entered.

For instance, if the mask:

```
SCR-/-
```

was entered to Fastback, then, if the mask never matches any input file, it will be printed as:

```
****:SCR*****.***/*****
```

This occurs because Fastback decodes the encoded select entry one character at a time, and the - character in the original entry will be interpreted as multiple occurrences of the * mask character. Notice that a volume name mask is always output as part of the total filename mask. If no volume name was given in the original select entry, the "any volume" mask (****:) will be output for the non-matched select entry.

The user should be aware of the difference between non-matched select entries output by Fastback, and those output by OS/32 Backup. Fastback checks files on the input medium against all entries in the select list, and hence a non-matched mask is one which has not matched any file on the input medium.

OS/32 Backup, however, only checks filenames against the select list until it finds a select entry that matches the given filename. Thus it is possible that select entries that could match one or more filenames may be output as not matched. Files which could match these select entries were first matched by another entry earlier in the select list.

CHAPTER 5
FASTBACK OPERATION

5.1 DESCRIPTION

Fastback is a disc backup and restore utility aimed at maximising performance of backup operations. To achieve the desired backup throughput, Fastback combines the following techniques: overlapped and multi-buffered input/output (I/O) operations, data compaction, and physical data transfers optimised by the program's knowledge of disc rotational characteristics. Each time Fastback is executed, it determines the amount of main storage available and it selects suitable data structures to maximise the performance of the required operation.

As part of its backup and restore procedures, Fastback may use the defective sector information contained in the (optional) disc pack administration file (PACKINFO.DIR). When restoring data to a disc, the program also updates the restore history information within the pack administration file. If a disc pack does not contain a pack administration file, then Fastback will log a message informing the user that the file cannot be found, and will complete the desired operation. A pack administration file need not exist for Fastback to be executed.

5.2 ACTIONS PERFORMED FOR EACH FUNCTION

Fastback is capable of performing the following functions:

- backing up a disc pack to magnetic tape,
- restoring data from a magnetic tape to a disc,
- verifying data on a previously created Fastback tape, against data held on disc, and
- displaying information pertaining to files held on a previously created Fastback tape or files to be backed up from an existing disc.

The following sections describe the procedures which Fastback uses to accomplish the above operations.

5.2.1 The Backup Operation

When Fastback commences operation, it processes the commands passed to it by the user and sets up the required data structures and option information. If the input disc is not marked on protected, and the NOPROTECT command has not been specified, then Fastback will output the following message to the user:

```
DISC devn:  (VOLUME voln) NOT MARKED ON PROTECTED
```

Unprotected discs may be backed up by using the NOPROTECT command. However, this command should be used only when it is operationally impossible to mark on the input disc protected. To understand the reason for this recommendation, the user is advised to read this section (on how backup operations work) and then read the description of the NOPROTECT command.

Depending on the amount of memory available, Fastback allocates its main storage to its error pool, select list, defective sectors list, I/O buffers, and sector allocation map.

The sector allocation map (SAM) is a bit array, where each bit represents some number of sectors on the disc. The minimum (and default) sized SAM is organized such that each bit represents an integer number of sectors. The minimum sized SAMs for various types of disc are given in Table 5-1.

TABLE 5-1 FASTBACK MEMORY SIZE

DISC TYPE	# SECTORS	# SECTORS/SAM BIT	SAM SIZE
256Mb	1000768	8	15637 Bytes
67Mb	263360	8	4115 Bytes
13.5Mb	52672	1	6584 Bytes
5Mb	19584	1	2448 Bytes
Floppy	1001	1	126 Bytes

The number of sectors represented by each bit in the SAM is termed the "granularity" of the SAM. The granularities shown in the above table are the minimum which Fastback will allow. If the prevailing memory size does not allow such a granularity, then Fastback will attempt to reduce the requested buffersize in order to make more memory available for the SAM. If, having reduced the buffersize to the minimum acceptable value (8 kb), there is still insufficient memory, Fastback will terminate with the following message:

```
INSUFFICIENT MEMORY FOR MINIMUM BUFFERSIZE
SEGMENT SIZE INCREMENT OF nn KB REQUIRED.  (SUGGEST mm KB)
```

On the other hand, if more memory than the minimum is available, then Fastback will optimise the size and number of buffers to be used and may also increase the granularity of the SAM. (Hence, in the best case, the SAM will be of a size such that one bit in the SAM represents one sector on the disc pack.)

Once all the above has been accomplished, Fastback writes the required volume and header labels to the tape device, and then reads the directory of the input disc. Should it encounter an invalid directory pointer, Fastback will output the message:

```
WARNING:  DIRECTORY SECTOR AT xxxxxx HAS INVALID FORWARD
          POINTER yyyyyy
          TREATING AS END OF DIRECTORY
```

It will then terminate the directory scan and begin to backup the data contained in those files whose directory entries it has already processed.

As it encounters each file's directory entry, the program tests the entry against the select criteria (if any), to ascertain whether or not the file is to be backed up. If so, the file's directory entry is used to construct a file association table (FAT) entry for the file. The FAT entry is then buffered for writing to tape. If the current file is an indexed type file (i.e., "normal" indexed, non-buffered indexed, long record length indexed, or extendable contiguous), its index blocks are read, "compressed" where possible, and buffered for writing to tape. Errors within the file's index blocks are detected, and appropriate warning messages are output.

As it assimilates directory and index block information, Fastback sets bits in the SAM to indicate which sectors or blocks of sectors, depending on the SAM granularity, are used by the current file to hold data. Index blocks/sectors do not have their corresponding SAM bits set. The program then moves on to the next directory entry and repeats the above procedure.

When the program has processed the entire disc directory, the result is a file on tape containing a mixture of (compressed) index blocks and FAT (i.e., "directory") entries. The resultant tape file serves as a "directory" for any restore or verifyonly operation. Having created this tape file, Fastback writes the appropriate end-of-file labels and filemarks. If VERIFY was specified, then Fastback repositions the tape to the start of the file and reads the file to check its integrity. Note that Fastback only reads each record of the file and validates its checksum; it does not recompute the written data or perform a data comparison. Should Fastback find a checksum error, it outputs the following message and then terminates.

```
CHECKSUM MISMATCH IN BLOCK nnn OF DIRECTORY/INDEX FILE
COMPUTED CHECKSUM = xxxxxx, EXPECTED yyyyyy
```

Once the file is verified, the program writes the file header labels for the forthcoming data file. At this time, the completed sector allocation map, held in memory, represents those sectors or blocks of sectors on the input disc which need to be copied. Fastback now scans the SAM and reads the indicated data blocks from the input disc, compresses and buffers them, and writes them to tape. During this process Fastback is using the defective sector information obtained from the pack administration file (PACKINFO.DIR) to avoid reading defective sectors (since this would involve a potentially large number of retries through the disc driver). If the administration file does not exist, the program will have already displayed the message:

```
****WARNING: PACK ADMINISTRATION FILE (PACKINFO.DIR) NOT FOUND
```

If this file does not exist, Fastback will run the backup operation without the defective sector information. If the file does exist but is found to contain invalid pointer data, then this message is generated:

```
****WARNING: PACK ADMINISTRATION FILE (PACKINFO.DIR) CORRUPTED
```

The backup then continues without the aid of this file.

When the SAM has been exhausted, all required data blocks have been copied. Verification of the data blocks, if requested, is now done by repositioning the tape to the start of the data file and comparing the copied data to that found on the input disc pack. If the copied data does not verify against the original disc data, Fastback will display the offending sector data from both the tape and disc devices upon the list device. The nonverified data is output in both hexadecimal and ASCII format, with pointers being used to indicate the location of the mismatch.

NOTES

1. The backup operation involves a copy of physical data blocks from a disc to tape (i.e., the operating system's logical access paths are not used for the data copy). If Fastback should encounter a defective sector whilst attempting to read a data block from the disc, it will add the address of the faulting sector to its error pool. Further actions taken by the error pool handler are given in Section 5.3.

2. Because Fastback accesses file entries sequentially down the disc directory, marking the input disc on with the control directory (CDIR) option does not increase the speed of the backup operation.

3. The current revision of Fastback does not update the backup history records within the (optional) pack administration file (PACKINFO.DIR).

To reduce the possibility of errors in a data tranfer to magnetic tape, the recommended blocking factors are:

TAPE RECORDING DENSITY	BLOCKING FACTOR
800 BPI	12.5K
1600 BPI	25.0K
6250 BPI	100.0K

Using larger blocking factors than those recommended gains little additional storage space and results in an insignificant reduction in processing time. However, it does increase the probability of data transfer errors resulting in verify errors.

5.2.2 The Restore Operation

When Fastback commences operation, it processes the commands passed to it by the user and sets up the required data structures and option information. If, at this time, the output disc is not marked on unprotected, then the program will display the message:

```
DISC devn:  (VOLUME voln) NOT MARKED ON UNPROTECTED
```

It will then pause to allow the user to mark the disc on.

In accordance with the amount of memory available, the program allocates storage to its error pool, select list, I/O buffers, file association table (FAT), and link table (LT).

The FAT is built on disc and in memory, using an array paging method (APM). The LT is a table used to hold mapping information so that the data on the tape - in physical sector sequence - can be reconstructed into logical records and files on the output disc. The LT is built totally on disc scratch space. The disc space required by the FATs and LTs will be obtained on the output disc unless the user requests another disc to be used, via the WORKSPACE command.

Once the above steps have been done, the program will read the file containing the FAT entries and (compressed) index blocks from the tape. If Fastback cannot read this file, or if checksum errors are detected, the program will output an error message and then will terminate.

FAT entries are tested against any active select criteria to determine whether they represent files to be restored. Entries which meet the select criteria are copied into the (APM'ed) FAT. As each file's entry is placed in the FAT, the file's desired name is noted. The desired name is either the original name or that determined by an active RENAME command. If the selected KEEPFILES option was either DISC or NEWER, Fastback now checks for a file on the output disc whose name matches the desired name of the file being restored. If such a file exists (and in the case of the NEWER option, it is newer than the file being restored), then the following message is given:

```
**** FILE fd ALREADY EXISTS
CURRENT FILE ON DISC WILL BE RETAINED
```

The existing disc file is kept, and the tape file is not restored. If the KEEPFILES = TAPE option was specified (or if NEWER was specified and the tape file is newer than the duplicate disc file), Fastback will restore the tape file to disc as an intermediate file, then delete the file with the duplicate name (if any), and rename the intermediate file.

Having taken care of any duplicate files, an "intermediate" file of the appropriate type is allocated. The intermediate files are allocated under an encoded name and with protection keys of FBAC. This is done to help prevent unwarranted access to the files until they are fully restored. The encoded filenames will take the form of FBxxxxxx.INT/act (where xxxxxx represents sequential hexadecimal numbers starting from zero and act represents the current account number under which Fastback is running) unless an alternative format is specified via the TEMPLATE command.

Contiguous files are allocated with their correct size, whilst indexed-type (i.e., "normal" indexed, non-buffered indexed, long record length indexed, and extendable contiguous) files are allocated and then prefilled with their correct number of logical records, in order to create each file's index and data blocks on the output disc.

This prefilling is done by writing the appropriate number of dummy records into each file, except in the case of extendable contiguous files. These files are prefilled by doing a single random write of the final dummy record of the file. (OS/32 allocates uninitialised data blocks for all records up to the final record.) Note that this prefilling of each indexed-type intermediate file may require a significant amount of time, and hence restore operations that involve mostly contiguous files (which do not require prefilling) and extendable contiguous files (which are prefilled with a single random write) will usually execute much faster than restore operations which involve "normal" indexed, nonbuffered indexed, or long record-length indexed files.

The input tape is now repositioned for a second pass of the FAT/index file on the tape, in order to create LT entries for the files. These are computed for contiguous files, whereas for indexed-type files, they are created by reading the index blocks of the intermediate disc file and matching the data pointers against the "original" data pointers obtained from the input tape. The resultant pairs of pointers ("old" data pointers and "new" data pointers) are entered into the LT. These pointers now map the location of the data blocks on the tape to their new location on the output disc (e.g., the data held in sector 1234 on the original backed up disc must now be written to sector 5678 on the output disc).

The LT is now sorted into ascending order of "old" data pointers. Having performed this, the LT represents data blocks in the same order as they will appear on the input tape.

Fastback now walks down the LT entries and the data blocks contained on the input tape. Each data block that is needed is extracted from the tape and written to the correct location on the output disc, guided by the pointers in the link table.

Verification, if requested, is then done by walking down the tape data file again and comparing the data against that written to the output disc. Data sectors which do not pass the verification check are dumped in hexadecimal and ASCII to the list device, along with indicators to show the occurrence of the mismatch. The sector addresses are also entered into Fastback's error pool. This information is used, once the restore is completed, to determine which files contain the nonverified data. These files are identified on Fastback's list output.

Finally, when all data blocks have been copied and verified, if so commanded, Fastback walks down its FAT and renames and reprotects each intermediate file to become the desired final output file. Files which did not verify, and those whose contents could not be extracted from the input tape, are identified on the restore listing.

NOTES

1. Because Fastback does a large amount of allocating and renaming of files, the speed of the restore operation will be increased if the output disc is marked on with the CDIR option.

2. The pack administration file (PACKINFO.DIR) is preferred, but not mandatory, for restore operations.

3. If a pack administration file exists on the output disc, then, when Fastback has completed the restore operation, it will update the restore history record within the administration file.

5.2.3 The Verifyonly Operation

This operation is handled in the same manner as the restore process. That is, the tape device is always considered to be the input device.

In order for the verifyonly to be performed, Fastback must build the FATs and LTs as outlined in the previous section. Intermediate files, however, are not constructed since Fastback verifies against the existing files of the appropriate names on the output disc.

Having built the FATs and LTs, the program runs the verify pass in an identical manner to that used by a "restore with verify" operation.

5.2.4 The Display Operation

This function may be used to list those files on a given device (disc or tape) which would be processed if a backup, restore, or verifyonly operation was effected using a specified select list.

For example, the entire contents of a previously created Fastback tape may be listed by specifying DISPLAY without a select list, or with a select list equivalent to -.-/-. Of course, a select display is also possible simply by specifying the desired file descriptor mask(s) in a Fastback select list.

In this mode, Fastback simply scans the input directory (on disc or tape) and lists those files which match any given select criteria. FATs, LTs, etc., are not required for display operations.

5.3 THE ERROR POOL

Because Fastback generally copies data to or from physical sectors on disc, if an error occurs, then the program does not usually know to which logical file (if any) the error pertains. To offset this condition Fastback maintains an error pool. During a backup, restore, or verifyonly operation, Fastback will place in its error pool the addresses of any unreadable sectors, sectors whose data does not verify, etc.

At the end of the operation, Fastback will scan the disc device to ascertain to which file(s) the problem sector(s) belong. Any files located in such a manner will be reported to the user in Fastback's listing output with the message:

```
**** SECTOR xxxxxx - reason it was added to error pool
     SECTOR BELONGS TO FILE fd
     POSSIBLE CORRUPT DATA BEGINNING AT RECORD nnnnnn
   [ NOTE: ERROR (POSSIBLY) OCCURRED BEYOND LOGICAL
                                          END-OF-FILE
         FILE PROBABLY/POSSIBLY UNAFFECTED          ]
```

At the commencement of its execution Fastback will have allocated a certain amount of memory to the error pool. This value may be altered through the use of the ERRORPOOL command. Should more errors be discovered than will fit in the error pool, the program will display:

```
******** ERROR POOL IS NOW FULL ********
         UNABLE TO RELATE FURTHER ERRORS TO SPECIFIC FILES
```

Whenever this happens, Fastback cannot keep track of further sector-related errors.

5.4 TIMING INFORMATION

The amount of time taken to perform a backup, restore, or verifyonly operation is heavily dependent on the amount of memory available to the program. In the backup case, it is fair to say that Fastback will generally keep the tape drive moving as fast as it can go. GCR tape drives (i.e., 6250 BPI), operating at speeds of 75 ips or above, may be the exceptions to this rule. In the general restore or verifyonly case, Fastback may move the tape unit in a slower fashion. This is due to the fact that more processing, record deblocking, and random disc I/O is required to effect these operations.

Tables 5-2 and 5-3 give the estimated times (in minutes) that Fastback should take to perform most operations involving reasonably full disc packs on a lightly loaded machine.

Note that tape rewind times, necessary before each tape swap in a multi-reel operation, are included in the times, but the actual tape swap times are not.

TABLE 5-2 FASTBACK OPERATING TIMES FOR 1600 BPI TAPE*

DISC TYPE	BACKUP (no ver)	BACKUP (+ ver)	OPERATION RESTORE (no ver)	RESTORE (+ ver)	VERIFYONLY
256Mb	70-80	140-150	350-380	500-530	250-280
67Mb	17-25	35-45	60-75	80-95	35-45
5Mb	2-4	3-7	15-20	25-30	5-10
Floppy	1	1-2	3-7	4-8	2-3

* For 75 ips, 1600 bpi Perkin-Elmer tape.

TABLE 5-3 FASTBACK OPERATING TIMES FOR 6250 BPI TAPE*

DISC TYPE	BACKUP (no ver)	BACKUP (+ ver)	OPERATION RESTORE (no ver)	RESTORE (+ ver)	VERIFYONLY
256Mb	9-15	20-35	130-140	150-170	30-40
67Mb	3-5	7-11	45-60	50-65	10-20
5 Mb	1-3	2-6	10-15	15-20	2-4
Floppy	1	1-2	2-6	3-7	2-3

* For 125 ips, 1600 bpi GCR tape.

The above times may vary with those obtained in practice. Restore times, especially, may vary. This is because of the large proportion of restore time which is spent prefilling the indexed type intermediate files. Hence a restore operation that involves mostly contiguous files will execute significantly faster than one which involves mostly indexed type files.

Other factors which affect the speed of each operation are: the segment size increment with which Fastback was loaded (since this determines the number of buffers that can be made available), the tape block size (i.e., buffer size), tape drive speed and density, how well the data on the disc is "organized", and whether or not the tape and disc are on separate selector channels. For information on how to "tune" Fastback, please refer to Section 5.5.

In general, Fastback operations typically take less time than the corresponding OS/32 Backup operations. The exception is probably restore (with no verify) operations, which may well take longer due to the prefilling of the indexed type intermediate files.

5.5 TUNING INFORMATION

There are three main things which affect Fastback's performance:

1. the configuration of the computer on which Fastback is running,

2. the segment size increment with which Fastback is loaded, and

3. the degree of organization of the disc pack involved.

The configuration of the machine includes such things as the types of disc and tape drives being used and the number and arrangement of disc/tape controllers and selector channels. Fastback will obviously perform better with peripherals that can transfer data at a high rate, as opposed to those that only support a slow data transfer rate. Thus a 1600 bpi tape drive operating at 75 ips will take less time to backup a given disc than an 800 bpi tape running at 45 ips.

Apart from the peripherals themselves, the configuration of controllers and selector channels on the host computer is important. A backup involving a disc and tape on separate selector channels will proceed faster than one where the two devices share a common selector channel. This is due to the fact that having individual selector channels allows the overlapping of data transfers to and from the machine's memory.

Hence, in general, Fastback will perform at its best given fast disc and tape drives, each located on its own selector channel and controller. In fact, Fastback is optimised for systems in which the tape and disc devices are on separate selector channels.

The second factor determining the speed of Fastback's operation is the amount of memory made available to it, in the form of a segment size increment. If Fastback has only enough memory to hold its SAM (see Section 5.2.1) and one tape buffer for a backup operation, it will operate by filling the buffer from disc and then writing it to tape. However, if sufficient memory for two buffers is available, the program will backup by reading and writing data in parallel (i.e., overlapping the I/Os).

As well as using memory to hold data buffers, Fastback may expand its SAM to a size where one SAM bit represents one sector on the input disc (i.e., a full bit map). Given this fine a granularity, Fastback can determine that an individual sector needs to be written to tape, rather than simply copying a fixed size block of sectors to tape, where the block contains only one required sector. Thus the segment size increment affects how much redundant data is written to tape.

The segment size increment plays an important part in the speed of backup operations. It also plays an important part in restore and verifyonly operations. These two operations execute fastest when at least three tape buffers are available. Also, the more memory available for the LT and the APM'ed FAT, the less disc accesses will be required for the use of these tables.

The released version of Fastback comes with a default segment size increment (as built by OS/32 LINK) which is sufficient to hold 2 x 16kb buffers and a SAM of reasonable granularity for most backup operations. For restore and verifyonly operations, the same segment size increment is sufficient to hold 3 x 16KB buffers and approximately 4kb each for the LT and FAT. These values were used because they represent a reasonably optimal memory configuration for typical Fastback operations. However, the user should be aware that loading Fastback with a larger segment size increment (and increasing the buffersize) will usually result in better performance.

The released version of Fastback is linked with a default segment size increment of 40kb.

The third factor influencing the speed of Fastback's operation is how well the data on the disc pack is organized. A well organized pack (i.e., one where the files occupy an almost contiguous area of the pack) will lend itself to faster backup, restore, and verifyonly operations. This is because Fastback is operating over a subset of the pack, and there are few unused sectors to skip over. Also, a well organized pack will usually require fewer LT entries to be generated for restore and verifyonly operations than will a disorganized pack.

The degree of pack organization also includes the organization of the disc directory. If all the directory entries fit on the preallocated area of the directory (as may be generated by OS/32 FASTCHEK), then Fastback's directory lookups will execute much faster than if the directory consists of a "chain" of sectors spread randomly throughout the pack.

Given all the above considerations, Table 5-4 outlines the approximate maximum, optimum, and minimum segment size increments (in K bytes) for backup (with verify) operations involving most disc devices. Note that the optimum increment represents the best tradeoff between performance and memory requirements. Loading Fastback with more than the maximum segment size increment will usually not gain the user any more throughput.

TABLE 5-4 SEGMENT SIZE INCREMENTS

DISC TYPE	PERFORM-ANCE	BUFFER SIZES 8	12	16	20	24	28	32
256Mb	MAXIMUM	134	146	158	170	182	194	206
256Mb	OPTIMUM	27	39	51	63	75	87	99
256Mb	MINIMUM	19	27	35	43	51	59	67
67Mb	MAXIMUM	44	56	68	80	92	104	116
67Mb	OPTIMUM	16	28	40	52	64	76	88
67Mb	MINIMUM	8	16	24	32	40	48	56
13.5Mb	MAXIMUM	18	30	42	54	66	78	90
13.5Mb	OPTIMUM	18	30	42	54	66	78	90
13.5Mb	MINIMUM	10	18	26	34	42	50	58
5Mb	MAXIMUM	14	26	38	50	62	74	86
5Mb	OPTIMUM	14	26	38	50	62	74	86
5Mb	MINIMUM	6	14	22	30	38	46	54
FLOPPY	MAXIMUM	12	24	36	48	60	72	84
FLOPPY	OPTIMUM	12	24	36	48	60	72	84
FLOPPY	MINIMUM	4	12	20	28	36	44	52

The above segment size increments have been calculated by using the following formulae:

MAXIMUM = INT(3 * B + (S / 8192) + 4 - 16 + 0.5)

OPTIMUM = INT(3 * B + (S / G / 8192) + 4 - 16 + 0.5)

MINIMUM = INT(2 * B + (S / G / 8192) + 4 - 16 + 0.5)

In these formulae:

B is the buffersize to be used for the backup operation.

S is the total number of sectors on the disc pack which is to be backed up.

G is the desired granularity of the SAM. (Note that the optimum figures in the above performance table have been calculated using SAM granularities of eight sectors per bit for the 256Mb and 67Mb discs, and one sector per bit for the remaining disc packs.)

The user may use the above formula to calculate segment size increment values for other discs being supported (e.g., 2.5Mb or 160Mb discs, etc.).

5.6 PACK ADMINISTRATION FILE

The pack administration file (PACKINFO.DIR/0) contains both a list of the defective sectors on the pack and a record of the administrative history of the pack. Its primary function is to provide the defective sector information so that when a pack is initialized or checked and the bit map has to be rebuilt, this can be done without performing a surface check to find the defective sectors.

The file is a contiguous file of some nine sectors for hard discs and only one sector on floppy discs since no administration history is maintained on these discs. The file is protected against deletion and update by normal application tasks by maintaining the directory entry with a write count of -1. The file is organised as a set of 64-byte records packed four to a sector. The first record is a control record containing global information and pointers to the data records that are either history records or defective sector records.

The history records record the following events:

- full pack backup performed,
- incremental backup performed (i.e., backup using SINCE or BEFORE, but no other file selection),
- select pack backup performed, and
- pack restore (full or select) performed.

Additional types of history records are supported by other utilities.

NOTE

Revision R00-01 of Fastback only supports pack restore (full or select) history records.

The pack administration file is created when OS/32 Fastchek is used to initialize a pack in the FILL or READCHECK mode. It will also be created if the NOREADCHECK mode is specified. Since the file is created at initialization time, it is always the first file in the directory and always occupies the first error free area of the required size (which will in general directly follow the volume descriptor).

If the user desires to examine the pack administration file, it can be dumped using the DISPLAY command of COPY/32. For an explanation of the pack administration file format, please refer to Appendix D of the OS/32 Fastchek Reference Manual.

CHAPTER 6
ERROR HANDLING, LIST OUTPUT AND MESSAGES

6.1 COMMAND ERROR HANDLING

The action taken when a command error is detected depends on the mode of command entry being used.

If either the immediate or batch command entry mode is being used, then the erroneous command line is displayed on the system console or multi-terminal monitor (MTM) terminal together with an error message and an indication of where in the line the error occurred. For example, if the start command

```
ST ,BACKUP,DISC=DSC1:,TAPE=MAG1:,VORIFY
```

is used, then this will result in the following messages:

```
UNRECOGNIZABLE KEYWORD
BACKUP,DISC=DSC1:,TAPE=MAG1:,VORIFY
                             ^
```

and the program will then terminate with end of task code 2.

Similarly, if the program is started specifying a file as the command device and that file contains the following commands:

```
BACKUP
DISC=DSC2:
TAPE=MAG1: ; VERIFY; NOERROR=SKIP
```

then the following messages will be displayed:

```
UNRECOGNIZABLE KEYWORD
TAPE=MAG1: ; VERIFY; NOERROR=SKIP
                     ^
```

and the program will terminate with end of task code 3.

If the interactive command entry mode is used, then an error in response to a prompt will cause an error message to be displayed on the interactive terminal, followed by the erroneous response and an indication of the position of the error. For example, if, in response to the prompt,

```
TAPE POSITIONING (#REWind, NORewind or nn files to skip) ?
```

the user replies with

```
REEWIND
```

then the messages

```
UNRECOGNISABLE KEYWORD
REEWIND
^

TAPE POSITIONING (#REWind, NORewind or nn files to skip) ?
```

will be displayed and the user should enter the correct response.

If batch or immediate command entry is used, then once all the individual commands have been checked, a final consistency check of the commands is made.

Two other types of errors can occur during the command processing phase. First, the program may be unable to assign the specified command device, and second, an input/output (I/O) error may occur on the command device.

If the command device specified in the start command cannot be assigned, then the message

```
ASSIGN ERROR xxlu FOR voln:filename.ext/act
  <error description>
```

is displayed on the system console or MTM terminal and the program then terminates with end of task code 2.

If an I/O error occurs on the command device, then the following message is displayed on the system console or MTM terminal.

```
I/O ERROR ON LU nn voln:filename.ext/act FUNC=xx
    STATUS=xxxx   <error description>
```

If the interactive command entry mode is being used, the program will pause, and, on being continued, will retry the errored I/O. If batch command entry is being used, the program will terminate with end of task code 3.

6.2 LIST OUTPUT

As part of each backup, restore, verifyonly, or display operation, Fastback produces a listing showing which files were processed, which select entries were never matched (if any), and what errors occurred during the operation.

6.2.1 List Output and Error Handling

No output to the list device occurs until all commands have been processed and validated. In addition, if a file is specified as the list device, all output will be appended after any existing data in the file.

If the specified list device is a printer (i.e., has device code X'7X'), then the program will output the heading:

```
PERKIN-ELMER OS/32 FASTBACK Rnn-nn LISTING  mm/dd/yy hh:mm:ss
PAGE nnn
```

in one line at the top of each page. Note that a maximum of 55 lines of data is output to each page.

If the list device is not a printer and is not the same device as the command device, then no pagination is performed and the heading

```
PERKIN-ELMER OS/32 FASTBACK Rnn-nn LISTING mm/dd/yy hh:mm:ss
```

is output only once.

If the list device is assigned to the same interactive device as the command device, then no pagination is performed, and no heading is output.

After the heading is printed, a message is output giving the operation to be performed and any options.

If an I/O error occurs on the list device, the message

```
I/O ERROR ON LU nn voln:filename.ext/act FUNC=xx
    STATUS=xxxx   <error description>
```

is displayed on the system console or MTM terminal, or on the interactive command device (if this is not the same as the list device). The program can then pause, and, on being continued, retries the errored I/O.

Error and informational messages are output to the list device as soon as the corresponding error, or other condition, is detected.

6.2.2 List Contents

The list output is made up of three intermixed components. These may be called the header, file details, and message components.

The header information is given on top of the list output. It includes such information as the requested function, the date and time when the function was performed, the devices involved, and the options selected.

The file details component gives information pertaining to each file processed. This is presented in a similar fashion to the OS/32 or MTM DISPLAY FILES command output. Additional information such as blocking factors and previous name (if the file was renamed) are also given.

The third component of the list output is the message component. The messages given include both error and informational messages. For example, I/O errors and data nonverifies will cause appropriate messages to be generated. Nonverify errors cause the offending data to be displayed in hexadecimal and ASCII format with indicators highlighting the actual data mismatch.

6.2.3 Sorting the List Output

The list output requires a file or device whose record length should be 132 characters. However, because this is not always available, the format of the list output is designed to keep as much of the important information as possible within the first 60 to 80 characters of each line. This is because the minimum usable width of an 80 character system console is approximately 60 characters, with the remaining characters being taken up by date/time and taskid stamping. Thus the preferred width of the output device or file is 132 characters, but lesser record lengths may also be used. Fastback does not dynamically adjust its output width to match the list device.

In columns 135 to 140 of the list output, Fastback places character sequences to aid sorting the output. For the header information, the sequence Annnnn is given, where nnnnn represents consecutive hexadecimal numbers starting from zero. All file descriptions have the characters FILE output in columns 135 to 140, while messages have the characters Mmmmmm output, where mmmmm represents consecutive hexadecimal numbers starting from zero. In the normal case of a 132 character printer, or a 132 character list file, the sequences in columns 135 to 140 will be lost. However, if the user wishes, he may use a list file with a record length of 140 bytes. Having done this, the output may be sorted (probably into ascending sequence). Such sorting is usually done on the file descriptor (fd) field, or some subset of it. Sorting the output by the date created, blocking factors, etc., is generally neither useful nor practical.

The list output sorting should always be done with columns 135 to 140 specified as the primary sort key so that the output header information will appear first, followed by the file descriptions and, finally, any informational or error messages, etc.

Table 6-1 shows where the various fields of the fds appear in Fastback's list output, and the corresponding secondary key parameters (assuming the sort is being done in ascending sequence) which may be specified to the KEY command of Perkin-Elmer's SORT/MERGE LEVEL II utility.

TABLE 6-1 FILE DESCRIPTOR FIELDS

FILE DESCRIPTOR FIELD	APPEARS IN COLUMNS	LENGTH OF FIELD	SORT/MERGE II KEY PARAMETER
Whole-fd	3-20	18	3/18/A
Filename	3-20	8	3/8/A
Extension	12-14	3	12/3/A
Account	16-20	5	16/5/A

The following examples show how to use Perkin-Elmer's SORT/MERGE LEVEL II utility to sort the list output.

Example 1:

To sort the list output into alphabetical order (i.e., ascending sequence) by whole fd, use the following SORT/MERGE LEVEL II commands:

```
>RECL 140
>KEY  135/6/A, 3/18/A
>SCRATCH
>SORT input-file > output-file
>END
```

Example 2:

To sort the list output into account number order, and into extension and filename order within each account, use the following SORT/MERGE LEVEL II commands:

```
>RECL 140
>KEY  135/6/A, 16/5/A, 12/3/A, 3/8/A
>SCRATCH
>SORT input-file > output-file
>END
```

When Fastback's list output is sorted using the above procedure, any additional lines appended after a file's description to inform the user of errors, etc., will be repositioned by the sort to the end of the listing. This may make it difficult to relate error information to specific files once the list has been sorted. For this reason, the original list output should always be kept.

The sorting ability is provided purely to allow the user to rearrange the list output so that specific files may be located more easily.

6.2.4 Using Output as a Select List

OS/32 EDIT may be used to create a select list from the list output. A select list created in this manner takes the form of complete fds (i.e., no masks) and no RENAME or KEYS parameters, etc. The following commands to OS/32 or MTM and OS/32 EDIT will create a select list from a Fastback list output. Note that * represents an OS/32 or MTM command prompt, and > is used to represent an OS/32 EDIT prompt.

```
*XALLOCATE SELECT.LST,IN,80/5          file to hold select list
*LOAD .BG,EDIT32/S                    load editor
*TASK .BG
*START                                start it
----- OS/32 EDIT ANNOUNCEMENT ----
>OPTION LENGTH=20,LIST=SELECT.LST     set length and list o/p
>GET FASTBACK.OUT                     get Fastback's list output

FILE RECORD LENGTH = 140, CURRENT LINE LENGTH = 20 editor warning

>FIND /./11,1-                        extract file descriptions
>OPTION LIST=CON:                     set list back to CON:
>DELETE 1-                            disconnect from list file
>OPTION LENGTH=80                     set length back to 80
>GET SELECT.LST                       get file of file descriptors
>COLUMN 1,/          /,1-             10 spaces kills line numbers
>CHANGE / /,//,1-                     remove spaces (redundant)
>DONE                                 save and then exit
```

Of course, the above procedure is only one of many possible methods which may be used to alter or adjust the list output to the user's requirements.

6.3 MESSAGE SUMMARY

This section documents all error and informational messages generated by the program. The messages are in alphabetical order. Thus messages that contain leading spaces (i.e., the messages are indented) are listed first, then messages that contain leading **** characters, and then normal messages where the first character is part of the message itself.

For the sake of clarity, the messages are shown in upper case with any parameters shown in lower case, although the actual messages are displayed in upper and lower case. For example, the message displayed as:

```
    Device devn:  is not a Disc
```

is documented as:

```
    DEVICE devn:  IS NOT A DISC
```

In addition, in order to avoid repetitious definitions, the following conventions are used in specifying the parameters.

- parameters shown as xxxx, yyyy, or zzzz represent hexadecimal values,
- parameters shown as llll, mmmm, or nnnn represent decimal values, and
- the parameters voln:filename.ext/act, filename.ext/act or fd, represent the name (i.e., file descriptor) of a file.

Messages:

WHILE ACCESSING DIRECTORY

Meaning:

This message may be generated immediately following a standard I/O error message, to inform the user that the indicated I/O error occurred while Fastback was accessing the disc directory. This message is usually generated by the directory lookup routines involved with finding/checking the pack administration file.

Program Action:

Fastback continues, and will inform the user that the pack administration file is unusable.

Required Operator Action:

None

PLEASE CLOSE THE FILE AND THEN CONTINUE FASTBACK

Meaning:

This message may be generated when Fastback is performing a restore or verifyonly operation and it finds a file which it needs, currently assigned for write access, to some other task. This message may be generated after the "File....is assigned for write" message. For further information please refer to the description of that message.

Program Action:

Fastback pauses to allow the user to close the assigned file.

Required Operator Action:

Identify which task currently has the file assigned for write, and close the appropriate logical unit (lu) or assign the file for read-only access.

WHILE ACCESSING FILE ASSOCIATION TABLE

Meaning:

This message accompanies I/O error messages for I/O errors which occur while Fastback is accessing its file association table (FAT) during a restore or verifyonly operation.

Program Action:

If the I/O error is not an unrecoverable error (i.e., SVC 1 status X'84'), then Fastback will pause. If the I/O error is an unrecoverable error, then Fastback will terminate if the ONERROR action is set to ABORT; otherwise, it will pause. Should Fastback terminate, it will generate a message saying "Fatal I/O error on file association table" and then terminate. It cannot attempt any cleanup procedures such as deleting outstanding intermediate files, because the FAT contains the information needed to perform such cleanup procedures.

Required Operator Action:

If Fastback pauses, the operator may be able to take corrective action, depending on the cause of the I/O error.

!GO NOT ALLOWED YET

Meaning:

Fastback is operating in interactive mode and the user has specified the !GO response in reply to one of the nondefaultable opening questions in the interactive command dialogue. The !GO response is not allowed because replies have not yet been obtained for all the nondefaultable questions, (i.e., there is still some information which Fastback must have before it can commence operation).

Program Action:

The prompt is reissued.

Required Operator Action:

None

**** 80 BYTE EOF1 OR EOV1 LABEL NOT FOUND AFTER FILEMARK

Meaning:

Fastback has encountered a filemark while reading the tape. It then expects to find either an EOF1 or an EOV1 tape label. Instead it has found either another filemark or a record that is not 80 bytes long.

Program Action:

Fastback will abort with end of task code 22. Any intermediate files, etc., will be deleted.

Required Operator Action:

None

**** BACKUP WITHOUT VERIFY COMPLETE - SUGGEST A VERIFYONLY BE RUN

Meaning:

This message is output to inform the user that a backup operation is complete, and no verify pass was made. The lack of a verify pass may have been caused by either the user not specifying verify, or by Fastback attempting to backup (and verify) an unprotected disc, with insufficient memory available to build a full sector allocation map (SAM) of granularity 1. In this latter case, a verify will not have been possible, and Fastback will have output a message to this effect, at the start of the operation. For further information on this condition, please refer to the description of the NOPROTECT command.

Program Action:

Fastback is about to terminate.

Required Operator Action:

None - except that a verifyonly operation should be run, if possible.

**** CANNOT ASSIGN FILE fd TO CHECK EXISTENCE AND AGE

Where:

fd	is the file descriptor of the file which Fastback cannot assign, in order to check if it exists, and if so, its date last written (i.e., its age).

Meaning:

When Fastback is performing a restore operation, before it allocates an intermediate file for any given file being restored, it may check whether or not a file already exists on the output disc with a duplicate name to that of the file currently being restored. (It will perform this check if KEEPFILES=DISC or KEEPFILES=NEWER.) In order to perform the check, Fastback attempts to assign a file of the given name (i.e., duplicate) on the output disc. If the assignment fails with supervisor call (SVC 7) status other than 4 (= file does not exist), 6 (= protection keys do not match), 7 (= privilege error), or 8 (= insufficient system space), then if ONERROR=SKIP, the above mentioned error message will be generated.

Thus, if Fastback's attempted assignment fails due to a file with a duplicate name not existing, or due to some error which reveals that such a file exists but cannot be assigned, then Fastback will not generate the above message.

Hence the message is intended to inform the user that the KEEPFILES check could not be made, and, since ONERROR=SKIP, Fastback is continuing with its current operation.

Program Action:

Fastback will continue the restore, but will not restore the file identified in the message since it cannot be sure if a file of the same name exists.

Required Operator Action:

None

**** END OF MEDIUM DETECTED WHILE WRITING LINK TABLE RECORD
WORKSPACE DISC IS FULL

Meaning:

Fastback is performing a restore or verifyonly operation and was building its link table (LT) on the WORKSPACE disc when it encountered end of medium (EOM) status while writing a record to the LT. That is, there was not a contiguous area on the WORKSPACE disc whose size was equal to, or greater than, the block size of the LT file.

Program Action:

Fastback will terminate, since it cannot build any more mapping information into the LT.

Required Operator Action:

Rerun Fastback with a WORKSPACE disc which has more free space.

**** EOF1 LABEL NOT FOLLOWED BY A FILEMARK

Meaning:

While reading a tape, Fastback has encountered a filemark, then an EOF1 label, and then expects to find another filemark. Its expectations were not fulfilled (it did not find a filemark).

Program Action:

Fastback will abort with end of task code 22. Any intermediate files, etc., will be deleted.

Required Operator Action:

None

**** FATAL I/O ERROR ON FILE ASSOCIATION TABLE

Meaning:

This message is generated when Fastback has encountered an unrecoverable I/O error (i.e., SVC 1 status X'84') while accessing its FAT, and the user's selected ONERROR action is set to ABORT.

Program Action:

Fastback will terminate with end of task.

Required Operator Action:

None

**** FILE ASSOCIATION TABLE APM ARRAY IS NOT CURRENTLY OPEN
CANNOT DELETE INTERMEDIATE FILES (TEMPLATE=fd)

Where:

fd	is the intermediate filename template used in the restore operation. The default template is FBnnnnnn.INT/act, where nnnnnn represents hexadecimal numbers in the range 000000 to FFFFFF, and act is the account number under which FASTBACK is currently operating.

Meaning:

Fastback is currently performing a restore operation, and either the user has invoked the SEND STOP command or else an error has occurred with the ONERROR action set to ABORT. Usually, under the given circumstances, Fastback will delete all outstanding intermediate files which it has allocated on the output disc, but, in this case, the error or STOP message occurred in the middle of the LT sort phase, when Fastback had temporarily closed its FAT which contains the names of the intermediate files. Hence, Fastback cannot delete any outstanding intermediate files. The message identifies the intermediate file template that is currently in use by Fastback, so that the user may manually delete those files that match the given template.

Program Action:

Fastback will continue with the activated termination routine, and delete any work files such as the FAT file and the LT file. It will then terminate.

Required Operator Action:

The operator may decide to manually delete all the remaining intermediate files.

```
**** FILE OUT OF SEQUENCE
**** FILE SEQUENCE NUMBER = nnnn, EXPECTED mmmm
PLEASE REPOSITION TAPE, OR MOUNT CORRECT TAPE, AND CONTINUE
```

Where:

nnnn	is the file sequence number read from tape.
mmmm	is the expected file sequence number.

Meaning:

Fastback is processing the header (HDR1) label on a new input tape. The file section number is correct, but the file sequence number is not.

As part of its backup procedure, Fastback writes two files to tape. The first is the FAT/Index file, which contains directory-type information, and the second file is the data file. The above message is generated when Fastback is processing a new header (HDR1) label, either on a new input tape or preceding one of the two mentioned files, and the file sequence number is incorrect. This number should always have a value of either 1 or 2, since there are only two files written to tape no matter how many reels of tape are used.

The message means that file number n follows the header label, instead of file number m.

Program Action:

Fastback will pause.

Required Operator Action:

The operator should check that the mounted tape is the correct one, and that it is positioned correctly. If all this appears correct, then Fastback may have written an incorrect header label during the backup operation, or else its expected value is incorrect. The problem should be reported as a potential Fastback bug.

```
**** FILE fd ALREADY EXISTS
     CURRENT FILE ON DISC WILL BE RETAINED
```

Where:

fd	is the file descriptor of the file on the output disc, whose name is identical to that of the file being read from the input tape.

Meaning:

During a restore operation, Fastback has encountered a file on the output disc whose fd is identical to that of the file currently being read from the input tape. Either the KEEPFILES=DISC option has been selected, in which case Fastback is retaining the file on the disc as requested, or else KEEPFILES=NEWER is in effect and the file on the output disc is newer than the file coming from the input tape. In both cases, the message informs the user that a file with a duplicate name exists on the output disc, and is being retained in preference to the file on the input tape.

Program Action:

The disc file is retained, and the tape file is not restored.

Required Operator Action:

None

**** FILE fd ALREADY EXISTS. CANNOT RENAME INTERMEDIATE FILE

Where:

fd	is the target file descriptor to which Fastback tried to rename the intermediate file.

Meaning:

Fastback is performing a restore operation, and is renaming the intermediate files to their target names. It has tried to rename an intermediate file, only to find that a file with the desired target name already exists.

If the ONERROR action is not set to PAUSE, then this message will be followed by further messages to identify the intermediate file, etc.

Program Action:

If ONERROR=PAUSE, then Fastback will pause, else Fastback will continue to rename the remaining intermediate files.

Required Operator Action:

If Fastback pauses, use a DISPLAY LU command to see which intermediate file is assigned to Fastback's lu4. This is the file which cannot be renamed. The operator should either delete the file whose name matches Fastback's desired name (i.e., target) for the intermediate file, or else the file should be renamed. Fastback should then be continued.

If Fastback does not pause, no immediate operator action is required. However, when Fastback terminates, any outstanding intermediate files should either be renamed or deleted.

**** FILE fd IS ASSIGNED FOR WRITE
[PLEASE CLOSE THE FILE AND THEN CONTINUE FASTBACK]

Where:

fd	is the file descriptor of the disc file that is currently assigned for write.

Meaning:

When Fastback is executing a restore or verifyonly operation, it needs to read the directory entries (and index blocks, for indexed, nonbuffered, and extendable contiguous files) for those files which it is restoring/verifying. If, having read a file's directory entry, Fastback determines that the file is currently assigned for write, Fastback will generate the above message.

Program Action:

If the specified ONERROR action is set to ABORT, then Fastback will terminate. If ONERROR=SKIP, then the file will be skipped, and will not be restored or verified. Finally, if PAUSE was selected as the ONERROR action, Fastback will generate the "Please close the file...." message and then pause.

Required Operator Action:

If Fastback pauses, the operator should attempt to find out which tasks have the file assigned and then close the appropriate logical units of those tasks. Fastback should then be continued.

**** FILE-ID FIELD OF TAPE HEADER LABEL (HDR1) IS INCORRECT
**** TAPE WAS NOT CREATED BY EITHER FASTBACK OR OS/32 BACKUP
**** SUGGEST USE OS/32 COPY TO EXAMINE CONTENTS OF TAPE
PLEASE REPOSITION TAPE, OR MOUNT CORRECT TAPE, AND CONTINUE

Meaning:

Fastback has read an 80 byte header (HDR1) label from a new input tape, but the file-id field within the label does not contain FASTBACK. Tapes created by Fastback have a file-id field set to FASTBACK Rnn-nn, where Rnn-nn is Fastback's revision and update levels. These revision and update levels are not checked when reading the tape.

Program Action:

Fastback will pause.

Required Operator Action:

Mount the correct tape. The current tape appears to be a labelled tape but has not, apparently, been created by Fastback.

**** FILEMARK FOUND AFTER HDR1 LABEL
**** UHL1 LABEL EXPECTED AFTER HDR1 LABEL
**** TAPE WAS NOT CREATED BY EITHER FASTBACK OR OS/32 BACKUP
**** SUGGEST USE OS/32 COPY TO EXAMINE CONTENTS OF TAPE
PLEASE REPOSITION TAPE, OR MOUNT CORRECT TAPE, AND CONTINUE

Meaning:

Fastback is processing an input tape, and has successfully read a volume label - VOL1 and a header label - HDR1. It has then tried to read the first of its required user header labels (UHLs), and has found a filemark instead of the expected UHL1 label. The tape was, apparently, not created by Fastback or OS/32 BACKUP, and if the user wishes, he should examine the contents of the tape with the DISPLAY feature of OS/32 COPY.

Program Action:

Fastback will pause, to give the user a chance to check the tape, etc.

Required Operator Action:

The correct tape should be mounted, or the current tape positioned to the start of Fastback's data, and then Fastback should be continued.

**** FILEMARK FOUND AFTER UHLn LABEL. EXPECTED NEXT UHL LABEL
PLEASE REPOSITION TAPE, OR MOUNT CORRECT TAPE, AND CONTINUE

Where:

n — is the number of the last UHL that Fastback successfully processed from the input tape.

Meaning:

Fastback is starting to process an input tape, and has successfully processed at least one UHL. While reading what it expected to be another UHL, Fastback encountered a filemark.

Program Action:

Fastback will pause to allow the user to intervene.

Required Operator Action:

The tape should be checked to ensure that it is a Fastback tape, and since Fastback will have displayed the contents of those UHLs that it did manage to process, this output should be checked to see if it appears to be valid Fastback UHL-type information (compare it with the output of other Fastback restores/verifyonlys/tape displays).

If the tape is incorrectly positioned, this should be corrected. If the wrong tape is mounted, the correct tape should be mounted.

**** FILEMARK FOUND. EXPECTED A TAPE LABEL
**** TAPE WAS NOT CREATED BY EITHER FASTBACK OR OS/32 BACKUP
**** SUGGEST USE OS/32 COPY TO EXAMINE CONTENTS OF TAPE
PLEASE REPOSITION TAPE, OR MOUNT CORRECT TAPE, AND CONTINUE

Meaning:

Fastback has begun processing a new input tape for a display, restore, or verifyonly operation. The first thing read on the tape was a filemark instead of the expected header label.

Program Action:

Fastback will pause to allow operator intervention.

Required Operator Action:

Please mount the correct tape, or position the current tape correctly. Then continue Fastback.

**** INSUFFICIENT DISC SPACE FOR A SINGLE LINK TABLE RECORD

Meaning:

For restore and verifyonly operations, Fastback allocates indexed files for its FAT and LT. When the LT file is allocated, Fastback writes a single dummy record into it, to ensure that there is sufficient disc space to at least get off the ground with the LT. (If this was not done, Fastback would perform a large amount of processing only to find that there was insufficient disc space to begin the LT and, hence, it would terminate after having proceeded unnecessarily into the restore/verifyonly operation.)

Hence, the above message means that there is insufficient contiguous disc space on the WORKSPACE disc, for even a single record (i.e., data block) of the LT file.

Program Action:

Fastback will delete the FAT and LT files, and will terminate with end of task code 5.

Required Operator Action:

Delete some files from the selected WORKSPACE disc, or specify another WORKSPACE disc.

**** INSUFFICIENT DISC SPACE TO ALLOCATE INTERMEDIATE FILE
(INTERMEDIATE FILE=fd1, ORIGINAL NAME OF FILE TO BE RESTORED=fd2)
[FILE fd2 WILL NOT BE RESTORED]

Where:

fd_1 is the file descriptor of the intermediate file which cannot be allocated/assigned due to insufficient disc space.

fd_2 is the target file descriptor of the file being restored. The target file descriptor will either be the name of the file as read from the input tape or, if the RENAME parameter of the select processor was in use, then the target file descriptor will be the RENAMEd filename.

Meaning:

When Fastback attempted to allocate/assign an intermediate file for the file to be restored, it received an SVC 7 error status indicating insufficient disc space. This condition will usually occur with contiguous files being restored, but can also occur with heavily blocked indexed files.

Note that the third line of the message, "File .... will not be Restored", is given only if Fastback continues the restore because the user has selected SKIP as the ONERROR action.

Program Action:

Fastback's action depends on the ONERROR option. If ONERROR=ABORT, then the program will terminate after deleting any intermediate and work files. If PAUSE was selected as the ONERROR option, then Fastback will pause and retry the allocate/assign if continued. Otherwise (ONERROR=SKIP), Fastback will simply not restore the file.

Required Operator Action:

If Fastback pauses, then the operator may take some corrective action, such as deleting some redundant files from the disc, and then continue Fastback.

**** INTERMEDIATE FILE int-fd NOT RENAMED nor-rep
(ORIGINAL NAME WAS orig-fd)

Where:

int-fd	is the file descriptor of the intermediate file which was not renamed, etc.
nor-rep	will either be nonexistent, or will be NOR REPROTECTED.
orig-fd	is the ORIGINAL file descriptor of the file read from tape. This is the name of the "real" file which is represented by the intermediate file. It is the name of the file as read from tape, not a name set by a RENAME command in a select entry.

Meaning:

During a restore operation, Fastback is renaming the intermediate files to their target filenames. An error occurred when Fastback attempted to rename an intermediate file to its target name, or when it tried to reprotect the file (i.e., to get rid of the FBAC protection keys).

Program Action:

Fastback will continue to rename the other intermediate files, but will leave the given intermediate file in its current state.

Required Operator Action:

Check the given intermediate file and manually rename and reprotect it.

**** NON 80 BYTE RECORD FOUND WHILE SKIPPING UHL LABELS

Meaning:

Fastback is performing a restore operation where the directory/index file on the tape crosses onto another tape. Fastback has begun to make its second pass of the directory/index file on the first tape and, while skipping over the UHLs, it has encountered a record that is not 80 bytes long (i.e., the record cannot be a UHL.)

Something is obviously wrong since Fastback should have checked the UHLs before the first pass of the directory/index file on the first tape. The most obvious explanation is that the currently mounted tape is not really the first tape.

Program Action:

Fastback gives the above message and then starts aborting the restore. Intermediate files, etc., will be deleted. Fastback will terminate with end of task code 22 (tape data format error).

Required Operator Action:

None

**** RECORD FOLLOWING FILEMARK IS NEITHER AN EOF1 NOR EOV1 LABEL

Meaning:

Fastback encountered a filemark while reading the tape. It then expected to find either an EOF1 or an EOV1 tape label. Instead, it found an 80 byte record that is neither an EOF1 nor an EOV1 label.

Program Action:

Fastback will abort with end of task code 22. Any intermediate files, etc., will be deleted.

Required Operator Action:

None

**** RECORD FOUND AFTER uhln LABEL. EXPECTED A FILEMARK
PLEASE REPOSITION TAPE, OR MOUNT CORRECT TAPE, AND CONTINUE

Where:

uhln	is the identifier of the last UHL (e.g., UHL5) that was successfully processed from the input tape.

Meaning:

Fastback was processing the UHLs on an input tape, and had determined that it had just processed the last (expected) UHL. It then performed a read on the tape, expecting to find a filemark, but instead read a record.

Program Action:

Fastback will pause to allow the user to examine the situation.

Required Operator Action:

The tape labels, etc., should be examined to ensure that the tape data is valid Fastback data and labels. If Fastback has erroneously attempted to process more UHLs than necessary, this should be reported as a potential bug. Otherwise, the correct tape should be mounted, or the tape repositioned correctly, etc., and then Fastback continued.

**** RECORD READ IS NOT A FASTBACK UHL LABEL.
EXPECTED NEXT UHL LABEL
PLEASE REPOSITION TAPE, OR MOUNT CORRECT TAPE, AND CONTINUE

Meaning:

Fastback had begun processing an input tape, and has read an 80 byte record (that it expected to be a UHL, but which is not a UHL).

Program Action:

The program will pause to allow the operator/user to check the tape positioning and/or contents.

Required Operator Action:

The tape positioning and/or contents should be checked. The correct tape should be mounted and positioned appropriately.

```
**** SECTOR xxxxxx - status
     SECTOR BELONGS TO FILE fd
     POSSIBLE CORRUPT DATA BEGINNING AT APPROXIMATELY
     RECORD nnnnnn

   [ NOTE: ERROR OCCURRED BEYOND LOGICAL END-OF-FILE
     ===== FILE PROBABLY UNAFFECTED ]

   [ NOTE: ERROR POSSIBLY OCCURRED BEYOND LOGICAL END-OF-FILE
     ===== FILE POSSIBLY UNAFFECTED ]
```

Where:

xxxxxx	is the hexadecimal logical block address of a sector, extracted from the error pool.
status	describes the reason that the file was in the error pool. This will be one of: CONTENTS UNREADABLE DURING ORIGINAL BACKUP DID NOT VERIFY COULD NOT BE READ FROM DISC COULD NOT BE READ FROM TAPE COULD NOT BE WRITTEN TO DISC
fd	is the file descriptor of the file on the disc to which the identified sector belongs; i.e., it may be a data sector in an indexed file, or a sector within a contiguous file, etc.
nnnnnn	is the approximate record number within the file which corresponds to the given sector. In the case of a contiguous file, this is the exact (relative) sector number within the file. In the case of an indexed file, this is the approximate record number of the first record in the file which is liable to have been corrupted by the nonverified, lost data, etc.

Meaning:

Fastback has completed the requested operation and has found that the error pool contains one or more entries. It is now scanning the disc to determine which files are affected by the sectors in the error pool.

If Fastback determines that a particular error (e.g., nonverify) may be beyond the logical end of file for an indexed-type file, then it will also generate the ERROR (POSSIBLY) OCCURRED BEYOND..... message. If this message is generated, then when the Fastback operation is completed, the user should check the contents of the specified files and ensure that they are unaffected. Note that Fastback does not generate POSSIBLY in the message if it is sure that the error is beyond the logical end of file.

Program Action:

Fastback will continue to examine the disc and error pool, and will eventually terminate.

Required Operator Action:

When the Fastback operation terminates, the user should check the contents of those files that Fastback believes are unaffected by the specified errors.

**** TAPE APPEARS TO HAVE BEEN CREATED BY OS/32 BACKUP Rnn-nn
DISC VOLUME: voln, SEQN#: mm, DATE/TIME RUN: date/time
PLEASE REPOSITION TAPE, OR MOUNT CORRECT TAPE, AND CONTINUE

Where:

nn-nn	is the revision and update levels of the version of OS/32 Backup which created the tape.
voln	is the volume name of the disc which is backed up on the tape.
mm	is the sequence number of the tape. For example, it might be the second reel of a backup which occupied three tapes.
date/time	is the date and time when OS/32 Backup was run (to create the tape). The date will either be given as dd/month-mnemonic/yy or month-mnemonic/dd/yy, depending on whether the operating system Fastback is running under was sysgened with European or U.S. date format, respectively. The time will always be given as hh:mm.

Meaning:

The tape that Fastback has read, seems to have been created by OS/32 Backup. The first record read was an 80-byte record, which is not a Fastback-type header label, and appears to be an OS/32 Backup label.

Program Action:

Fastback will pause to allow the operator to change or reposition the tape.

Required Operator Action:

The tape should be changed for the correct tape, or it should be positioned to the correct position on the tape. Fastback should then be continued.

**** TAPE OUT OF SEQUENCE
**** FILE SECTION NUMBER (I.E. REEL SEQN#) = nnnn, EXPECTED mmmm
PLEASE REPOSITION TAPE, OR MOUNT CORRECT TAPE, AND CONTINUE

Where:

nnn	is the file section number (i.e., reel sequence number) read from tape.
mmmm	is the expected tape sequence number.

Meaning:

During a multi-reel display, restore, or verifyonly, Fastback has begun processing a new input tape, only to find that the file section number (i.e., tape sequence number) does not match the expected number. For example, the message would be generated if the operator mounted the third tape of a multi-reel backup instead of the second tape.

Program Action:

Fastback will pause.

Required Operator Action:

Please mount the correct tape.

**** TAPE VOLUME LABEL (VOL1) NOT FOLLOWED BY HEADER LABEL (HDR1)
**** TAPE WAS NOT CREATED BY EITHER FASTBACK OR OS/32 BACKUP
**** SUGGEST USE OS/32 COPY TO EXAMINE CONTENTS OF TAPE
PLEASE REPOSITION TAPE, OR MOUNT CORRECT TAPE, AND CONTINUE

Meaning:

Fastback read a volume label (VOL1) from a new input tape. It then expected to find a header label (HDR1). Instead, it read an unrecognizable 80-byte record.

Program Action:

Fastback will pause.

Required Operator Action:

Mount the correct tape since the current one does not have an HDR1 label.

**** THE FOLLOWING ENTRIES WERE MADE TO THE ERROR POOL:
(NOTE: FILE RECORD NUMBERS START FROM ZERO)

Meaning:

This message is given as a heading when Fastback is scanning the disc to locate files which contain sectors that had been added to the error pool. The message is immediately given before Fastback outputs the name of the first file which owns a sector in the error pool.

The note accompanying the message informs the user that when Fastback outputs a message saying possible corrupt data at approximately record n, the record number n starts from zero. Hence the first record in an indexed file, or sector in a contiguous file, would be given as record zero.

Program Action:

Fastback outputs any sectors from the error pool, whose owner file has been determined.

Required Operator Action:

None

**** VOLUME DESCRIPTOR HAS INVALID DIRECTORY POINTER xxxxxxxx

Where:

xxxxxxxx — is the invalid pointer to the first sector of the primary directory, as obtained from the disc's volume descriptor.

Meaning:

The disc specified to the DISC command has an invalid pointer to the first sector of the primary directory. The pointer contains a sector address which is too large for the type of disc involved.

Program Action:

Fastback will terminate with end of task code 8.

Required Operator Action:

Check the disc with OS/32 Fastchek.

**** WARNING: INSUFFICIENT MEMORY TO VERIFY UNPROTECTED DISC
SUGGEST A VERIFYONLY BE RUN WHEN BACKUP IS COMPLETED

Meaning:

This message will be output at the start of a backup operation if the input disc is not marked on protected, the NOPROTECT command was specified, and Fastback has insufficient memory available to build a SAM of granularity 1. Refer to Chapter 5 for a description of SAM.

This message is to inform the user that a verify pass, as requested, cannot be run and that a verifyonly pass should be run when the backup operation is completed.

A verify pass cannot be made because Fastback's SAM granularity is not equal to one bit that corresponds to one sector on disc. For example, the granularity might be such that one SAM bit maps eight sectors on the input disc. This means that Fastback walks down the disc setting SAM bits for every data sector that needs to be copied to tape, and then scans the SAM and copies all the indicated sectors to tape.

If the granularity is eight, then it is possible that eight sectors could be copied to tape even though only one of them holds required data. If a verify pass were then run, one of the redundant seven sectors (from the above mentioned eight sector block) may have been allocated to a file and written to. Hence, Fastback would get nonverify errors on the tape data block, even though the needed data sector verified successfully. A large number of nonverifies could be generated in this manner.

Program Action:

Fastback will continue the backup operation, but will not attempt to verify the data written to tape.

Required Operator Action:

None, except that a verifyonly pass should be run if possible, when the backup operation is completed.

**** WARNING: PACK ADMINISTRATION FILE (PACKINFO.DIR) CORRUPTED

Meaning:

Fastback has read the pack administration file on the specified disc device and found it to contain corrupted data.

Program Action:

Fastback will continue without the (defective sector list) information contained in the administration file.

Required Operator Action:

None, as far as Fastback is concerned, but OS/32 Fastchek should be used to rebuild the pack administration file.

**** WARNING: PACK ADMINISTRATION FILE (PACKINFO.DIR) NOT FOUND

Meaning:

Fastback could not find the pack administration file on the specified disc device.

Program Action:

Fastback will continue operating without the information (defective sector list) contained in the administration file.

Required Operator Action:

None

**** WARNING: PACK ADMINISTRATION FILE (PACKINFO.DIR) OVERFLOWED

Meaning:

Fastback has read the pack administration file and found that the defective sector list has overflowed. That is, there are more defective sectors on the disc than can be held in the defective sector list.

Program Action:

Fastback will continue operation, using the available defective sector list.

Required Operator Action:

OS/32 Fastchek should be used to check and, if necessary, reinitialize the disc. A large number of defective sectors on the disc may indicate a hardware problem, and the pack involved may require reformatting and the disc drive should be checked.

**** WARNING: PACK ADMINISTRATION FILE (PACKINFO.DIR) UNUSABLE

Meaning:

Fastback encountered an I/O error while accessing the disc directory, during its search for directory entry for the pack administration file (PACKINFO.DIR). Since Fastback cannot determine at the time the error occurred whether or not PACKINFO.DIR exists, it declares the file to be unusable and continues operation without it.

Program Action:

Fastback will continue operation without the pack administration file.

Required Operator Action:

Fastback will continue and will experience disc directory problems. OS/32 Fastchek should be run with REPORTONLY to check the state of the disc pack. Depending on Fastchek's report, the user may have to try to stitch the disc directory chain back together again and then run Fastchek again.

Note that repairs to the directory chain are not hard if the disc was initialized by Fastchek with a sufficiently large preallocated directory, since each successive directory sector is located at regular intervals on the disc.

```
**** nn BYTES READ FROM TAPE. 80 BYTE LABEL EXPECTED
**** RECORD READ IS NOT A FASTBACK UHL LABEL.
     EXPECTED NEXT UHL LABEL
PLEASE REPOSITION TAPE, OR MOUNT CORRECT TAPE, AND CONTINUE
```

Where:

nn — is the decimal length of the record which was read from the input tape.

Meaning:

Fastback was expecting to read an 80 byte UHL label from the input tape, but instead it read a record whose length was not 80 bytes.

Program Action:

The program will pause to allow operator action.

Required Operator Action:

The tape should be checked, and if it is the wrong tape, the correct tape should be mounted. If the tape was incorrectly positioned, it should be positioned to the start of the Fastback backup on the tape. When the user is satisfied that the situation has been corrected, Fastback should be continued.

```
**** nn BYTES READ FROM TAPE. 80 BYTE LABEL EXPECTED
**** TAPE WAS NOT CREATED BY EITHER FASTBACK OR OS/32 BACKUP
**** SUGGEST USE OS/32 COPY TO EXAMINE CONTENTS OF TAPE
PLEASE REPOSITION TAPE, OR MOUNT CORRECT TAPE, AND CONTINUE
```

Where:

nn — is the decimal number of bytes read from the tape.

Meaning:

Fastback is processing a new tape for a display, restore, or verifyonly operation. It may have read one or more header labels, and is expecting another one, but instead it has read a record whose length is not 80 bytes. (Header labels are all 80 bytes long.)

Program Action:

Fastback will pause to allow operator action.

Required Operator Action:

Mount the correct tape, or reposition the current tape to the correct position and then continue Fastback.

```
**** nn BYTES READ FROM TAPE. 80 BYTE LABEL EXPECTED
**** TAPE VOLUME LABEL (VOL1) NOT FOLLOWED BY HEADER LABEL (HDR1)
**** TAPE WAS NOT CREATED BY EITHER FASTBACK OR OS/32 BACKUP
**** SUGGEST USE OS/32 COPY TO EXAMINE CONTENTS OF TAPE
PLEASE REPOSITION TAPE, OR MOUNT CORRECT TAPE, AND CONTINUE
```

Where:

nn is the decimal number of bytes read from tape.

Meaning:

After reading a VOL1 label from a new input tape, Fastback has read an unrecognizable record whose length was not 80 bytes.

Program Action:

Fastback will pause to allow operator action.

Required Operator Action:

Mount the correct tape, or position the current tape to the start of the Fastback data.

******** ERROR POOL IS NOW FULL ********
UNABLE TO RELATE FURTHER ERRORS TO SPECIFIC FILES

Meaning:

Fastback's internal error pool is now full. The error pool is used to collect the addresses of disc sectors whose data is corrupted, unreadable, nonverified, etc. At the end of its execution, Fastback attempts to identify to which file each sector in the error pool belongs. Because the error pool is now full, any further sector related errors that occur cannot be stored, and the files to which they belong cannot be identified later on.

Program Action:

Fastback will continue execution, but cannot store any further sector related errors.

Required Operator Action:

None

ALLOCATE/ASSIGN ERROR xxll FOR voln:filename.ext/act
<error description>

Where:

xx	is the SVC 7 error code.
ll	is the logical unit number which was involved in the SVC 7 error code.
voln:filename.ext/act	is the file descriptor of the file being allocated.

<error description> will be one of the following

ERROR CODE	DESCRIPTION
1	ILLEGAL FUNCTION
2	ILLEGAL LU
3	VOLUME NOT MOUNTED
4	FILE ALREADY EXISTS
5	INSUFFICIENT DISC SPACE
6	PROTECTED BY KEYS
7	PRIVILEGE ERROR
8	INSUFFICIENT SYSTEM SPACE
9	ALREADY ASSIGNED OR OFFLINE
A	VOLUME NOT DIRECT ACCESS
B	INVALID FILE NAME
C	TRAP GENERATING DEVICE
D	ACCOUNT VIOLATION
E-7F	UNKNOWN ERROR
80-FF	I/O ERROR, STATUS FROM SVC1

Meaning:

The specified file cannot be allocated/assigned for the indicated reason.

Program Action:

The action taken depends on what Fastback was attempting to do with the SVC 7 error code. Generally, the action taken is determined by the state of the ONERROR or the KEEPFILES commands. See Chapter 3.

Required Operator Action:

None

ASSIGN ERROR xxll FOR voln:filename.ext/act
<error description>

Where:

xx	is the SVC 7 error code.
ll	is the logical unit number.
voln:filename.ext/act	is the file descriptor of the device being assigned.

<error description> will be one of the following:

ERROR CODE	DESCRIPTION
1	ILLEGAL FUNCTION
2	ILLEGAL LU
3	VOLUME NOT MOUNTED
4	DOES NOT EXIST
6	PROTECTED BY KEYS
7	PRIVILEGE ERROR
8	INSUFFICIENT SYSTEM SPACE
9	ALREADY ASSIGNED OR OFFLINE
B	INVALID FILE NAME
C	TRAP GENERATING DEVICE
D	ACCOUNT VIOLATION
E-7F	UNKNOWN ERROR
80-FF	I/O ERROR, STATUS FROM SVC1

Meaning:

The specified file/device cannot be assigned for the indicated reason.

Program Action:

If the command device is being assigned, the program will terminate with end of task code 2. If the device is either the list or disc device, then if the interactive command entry mode is being used, the appropriate prompt will be reissued; otherwise, the program will terminate.

If a workfile, etc., is being assigned then the action taken will usually depend on the state of the ONERROR option. Otherwise, Fastback will pause or continue, if possible, after generating a warning message.

Required Operator Action:

None

CHECKSUM ERROR IN BLOCK nnn OF DIRECTORY/INDEX FILE
CHECKSUM READ WAS xxxxxxxx, EXPECTED yyyyyyyy

Where:

nnn — is the decimal tape block (i.e., record) number of the block whose imbedded checksum does not validate.

xxxxxxxx — is a hexadecimal dump of the checksum which was calculated over the tape block when the block was read.

yyyyyyyy — is a hexadecimal dump of the checksum which was imbedded in the read tape block.

Meaning:

Fastback is currently processing an input tape for a display, restore, or verifyonly operation. It is reading the first data file (which contains the FAT (i.e., directory-type information) and compressed index block pointers) and has just read a block, calculated the checksum, and discovered that the calculated checksum does not match the checksum which is imbedded in the block itself.

Program Action:

Fastback will attempt to delete any intermediate files and workfiles that have already been allocated and will terminate.

Required Operator Action:

None

CHECKSUM MISMATCH IN BLOCK nnn OF DIRECTORY/INDEX FILE
COMPUTED CHECKSUM=xxxxxxxx, EXPECTED yyyyyyyy

Where:

nnn — is the tape block number (in decimal) of the block with the checksum mismatch.

xxxxxxxx — is a dump of the checksum (in hexadecimal) which was calculated over the tape block.

yyyyyyyy — is a dump of the expected checksum (in hexadecimal) as obtained from within the tape block itself.

Meaning:

During a backup with verify operation, Fastback is verifying the directory/index block tape file, having just completed writing it to tape. While reading the file, it has found a block whose checksum does not match the expected value (i.e., the imbedded value).

Program Action:

Fastback will continue to verify the rest of the FAT/index file, but will terminate with end of task code 9 when the verify of the FAT/index file is complete.

Required Operator Action:

Retry the requested backup operation.

COMMENCING VERIFY PASS OF DATA FILE ON TAPE nn

Where:

nn	is the number of the current tape in the sequence; e.g., the first tape of any operation is tape number 1.

Meaning:

This message is simply to inform the user that Fastback is about to start verifying the data on the current tape.

Fastback does not inform the user when, during Backup operations, it verifies the directory file that it writes to tape. This is because this file is small and takes an insignificant amount of time to verify.

Program Action:

Fastback verifies the current data file on the current tape.

Required Operator Action:

None

DELETE ERROR xxll FOR voln:filename.ext/act
<error description>

Where:

xx	is the SVC 7 error code.
ll	is the logical unit number.
voln:filename.ext/act	is the file descriptor of the file being deleted.

<error description> will be one of the following:

ERROR CODE	DESCRIPTION
1	ILLEGAL FUNCTION
3	VOLUME NOT MOUNTED
4	DOES NOT EXIST
6	PROTECTED BY KEYS
7	PRIVILEGE ERROR
8	INSUFFICIENT SYSTEM SPACE
A	VOLUME NOT DIRECT ACCESS
B	INVALID FILE NAME
D	ACCOUNT VIOLATION
E-7F	UNKNOWN ERROR
80-FF	I/O ERROR, STATUS FROM SVC1

Meaning:

The specified file cannot be deleted for the indicated reason.

Program Action:

The program action is determined by the state of the KEEPFILES and/or the ONERROR command. See Chapter 3.

Required Operator Action:

None

DEVICE devn: IS A DISC - INVALID AS COMMAND, LIST OR SELECT DEVICE

Where:

devn: is the device mnemonic of the device concerned.

Meaning:

A bare disc device has been specified to the COMMAND, LIST, or SELECT commands. Although Fastback can assign a bare disc, such a disc must not be used as a command or select input device nor as a list output device.

Program Action:

If operating in interactive mode, Fastback will reprompt the user. If operating in batch or immediate mode, Fastback will terminate.

Required Operator Action:

None

DEVICE devn: IS NEITHER A DISC NOR A TAPE

Where:

devn is the device mnemonic specified by the user.

Meaning:

The input device specified for a display operation is neither a disc nor a tape. The specified device must be either a disc or tape.

Program Action:

Fastback will reprompt the user since it must be operating in interactive mode.

Required Operator Action:

Correct the device specification.

DEVICE devn: IS NOT A DISC

Where:

devn: is the device mnemonic of the specified device.

Meaning:

The device specified to the DISC or WORKSPACE commands is not a disc.

Program Action:

Fastback will abort if operating in batch or immediate modes; otherwise, it will reissue the appropriate prompt.

Required Operator Action:

None

DEVICE devn: IS NOT A TAPE

Where:

devn: is the device mnemonic which the user specified to the TAPE command.

Meaning:

The device which the user specified to the TAPE command is not a tape.

Program Action:

If operating in interactive mode, Fastback will reprompt the user. If operating in batch or immediate modes, Fastback will terminate.

Required Operator Action:

None

DISC AND/OR TAPE DEVICE NOT SPECIFIED

Meaning:

The user has started Fastback either by passing it a group of start arguments or by specifying a noninteractive command device. Fastback has processed all arguments/commands and has not encountered commands to define both the disc and tape devices to be used. In the case of the display operation, no device (disc or tape) has been specified.

Program Action:

Fastback will terminate with end of task code 4.

Required Operator Action:

None

DISC devn: (VOLUME voln) NOT MARKED ON priv

Where:

devn: is the device mnemonic of the disc drive containing the pack that is not marked on with the necessary access privilege.

voln is the volume name of the pack that is not marked on with the necessary access privilege.

priv is the disc access privilege that is required for Fastback to perform the user's requested operation. For backup operations, Fastback requires the input disc to be marked on PROTECTED, unless the NOPROTECT command was specified. For restore operations, the output disc must always be marked on UNPROTECTED. For verifyonly operations, the disc must be marked on PROTECTED or UNPROTECTED.

Meaning:

A disc pack involved in the user's requested operation is not marked on with the necessary access privilege. This message applies to the disc specified by the DISC command.

Program Action:

If Fastback was started in interactive mode, it will generate the error message and then reissue the appropriate prompt. In other modes (immediate and/or batch), Fastback will terminate with end of task code 4.

Required Operator Action:

Mark the indicated disc on with the required access privilege, and then run Fastback again.

ERROR(S) ENCOUNTERED IN SELECT CRITERIA

Meaning:

Fastback has been started with select entries to be read from a noninteractive device, and Fastback has discovered errors in the select entries while processing them.

Program Action:

Fastback will terminate with end of task code 7.

Required Operator Action:

None

FASTBACK TERMINATING WITH EOT CODE nnn

Where:

nnn is the decimal end of task code.

Meaning:

Fastback has completed the user's requested operation to the best of its abilities. This message presents the end of task code to the user so that he may be aware of how successful the operation was. A list of Fastback end of task codes may be found in Appendix B.

Note that if Fastback had to abort the operation because of a serious error, then the REQUESTED OPERATION ABORTED..... message will be given instead of this message.

Program Action:

Fastback terminates with the indicated end of task code.

Required Operator Action:

None

FINISHED WITH MAGNETIC TAPE. TAPE MAY BE REMOVED

Meaning:

Fastback has finished using the magnetic tape. The user may dismount the tape.

This message is most applicable to restore operations, where Fastback may continue operating for a significant period after it has finished using the magnetic tape. This message informs the user that he may dismount the tape while Fastback continues to operate. In restore operations, Fastback continues executing, renaming intermediate files and analyzing errors.

Program Action:

Fastback will continue to operate for some period of time. For backup, display, and verifyonly operations, this time period is usually very short, unless Fastback has to identify errored sectors.

Required Operator Action:

None

FUNCTION (BACKUP/RESTORE/VERIFYONLY/DISPLAY) NOT SPECIFIED

Meaning:

The user has started Fastback either by passing it a number of start arguments or by specifying a noninteractive command device. Fastback has processed all arguments/commands and has not encountered a command that specifies the operation to be performed (i.e., backup, restore, verifyonly, or display).

Program Action:

Fastback will terminate with end of task code 4.

Required Operator Action:

None

I/O ERROR ON LU nn voln:filename.ext/act FUNC=xx RAND=yyyyyyyy STATUS=zzzz <error description>

Where:

nn	is the logical unit number.
voln:filename.ext/act	is the file descriptor of the device assigned to the logical unit.
xx	is the SVC 1 function code.
yyyyyyyy	is the SVC 1 random address (in hexadecimal).
zzzz	is the SVC 1 status code.

<error description> will be one of the following:

STATUS	DESCRIPTION
C0XX	ILLEGAL FUNCTION
A0XX	DEVICE UNAVAILABLE
B0XX	DEVICE UNAVAILABLE - AFTER EOT
90XX	END OF MEDIUM
88XX	END OF FILE
98XX	END OF FILE - AFTER EOT
84XX	UNRECOVERABLE ERROR
94XX	UNRECOVERABLE ERROR - AFTER EOT
8281	I/O HALTED
9281	I/O HALTED - AFTER EOT
8282	TIMED OUT
9282	TIMED OUT - AFTER EOT
8283	DEVICE WRITE PROTECTED
9283	DEVICE WRITE PROTECTED - AFTER EOT
8291	REQUEST PURGED
9291	REQUEST PURGED - AFTER EOT
82XX	PARITY OR RECOVERABLE ERROR
92XX	PARITY OR RECOVERABLE ERROR - AFTER EOT
8100	ILLEGAL OR UNASSIGNED LU
80XX	UNKNOWN ERROR
0000	ZERO STATUS AFTER EOT - EXPECT X'9000'

Meaning:

An I/O error occurred on the specified lu performing the specified function.

Note that the RAND=yyyyyyyy is suppressed if random access was not being used in the function error.

Also note that all error messages containing AFTER EOT, must involve the lu to which a magnetic tape is assigned.

Program Action:

If the error occurred on the command or list devices, then the actions discussed in Sections 6.1 and 6.2, respectively, take place.

In other cases, the action taken generally depends on the state of the ONERROR option. In some isolated cases, Fastback will automatically continue after the I/O error and will inform the user of any actions taken or not taken as a result of the I/O error. See Section 3.2.13.

Required Operator Action:

If Fastback pauses, attempt to remedy any obvious faults (such as device going offline, etc.), and then continue Fastback. If recovery is not possible, then Fastback may be reexecuted to retry the whole operation.

IMMEDIATE SELECT ENTRY EXCEEDS 132 CHARACTERS

Meaning:

The user has used Fastback's select immediate utility, (i.e., SELECT=(select entry)) and has specified an immediate select entry which is longer than the maximum allowed length of 132 characters.

Program Action:

Fastback will terminate if operating in batch or immediate modes; otherwise, it will reissue the appropriate command prompt.

Required Operator Action:

None

IMMEDIATE SELECT ENTRY NOT TERMINATED BY ")"

Meaning:

The user has invoked Fastback's select immediate utility, (i.e., SELECT=(select entry)) and has left off the closing parenthesis from the immediate select entry.

Program Action:

Fastback will reprompt the user when operating in interactive mode. However, in batch and immediate modes, Fastback will terminate.

Required Operator Action:

None

INSUFFICIENT MEMORY FOR MINIMUM BUFFER SIZE
SEGMENT SIZE INCREMENT OF nn KB REQUIRED. (SUGGEST mm KB)

Where:

nn — is the minimum segment size increment (in decimal) required for Fastback to support the requested - not minimum - buffersize.

mm — is the recommended segment size increment (in decimal) for Fastback to support the requested - not minimum - buffersize. This segment size increment represents a good tradeoff between memory requirements and performance for backup operations. mm segment size is recommended because it gives a significant improvement performance over backup operations performed with the minimum segment size increment.

Meaning:

Fastback has insufficient memory to perform a backup operation even using its smallest allowable buffer size (8kb). When Fastback commences a backup operation, it uses the selected buffer size to calculate the amount of memory it requires. If Fastback finds that it has insufficient memory, it decrements the buffer size from the selected value down to the next lower acceptable value and calculates the amount of memory it requires. This process continues until either Fastback finds that it has sufficient memory to support a particular buffer size, or it comes down to its lowest allowable buffer size (8kb) and finds that it has too little memory to even support that buffer size. Having reached this stage, it outputs the messages given above.

Program Action:

Fastback will terminate with end of task code 5.

Required Operator Action:

Reload Fastback with a larger segment size increment, preferably the suggested segment size increment.

INSUFFICIENT MEMORY FOR REQUIRED ll KB BUFFERS
SEGMENT SIZE INCREMENT OF mm KB REQUIRED. (SUGGEST nn KB)

Where:

ll — is the buffer size Fastback needs to use, expressed in a decimal number of kbytes.

mm — is the minimum segment size increment that Fastback must be loaded with in order to support the required buffer size.

nn — is a suggested segment size increment. Loading Fastback with this segment size increment will result in better performance than loading it with the above mentioned minimum segment size increment.

Meaning:

Fastback has insufficient main memory available to it. It is unable to perform the required operation.

Program Action:

After giving the above message(s), Fastback will terminate with end of task code 5.

Required Operator Action:

Fastback should be reloaded with a larger segment size increment. The segment size increment should be at least equal to the minimum value given and preferably equal to or greater than the suggested value given.

INSUFFICIENT MEMORY FOR SELECT LIST

Meaning:

Fastback has been reading select entries and encoding them internally. The above message is generated if Fastback runs out of memory to store the encoded select entries.

Program Action:

Fastback will terminate with end of task code 5.

Required Operator Action:

Rerun Fastback with a larger segment size increment.

INVALID DATE/TIME

Meaning:

The user has specified an invalid date or time to either the SINCE or BEFORE commands. This can also occur in select entries that use the SINCE (>) or BEFORE (<) parameters.

Program Action:

If the invalid date/time were specified to the SINCE or BEFORE commands, then Fastback will terminate when operating in batch or immediate modes, or will reissue the appropriate prompt when operating in interactive mode.

If the invalid date/time were found in a select entry, Fastback will flag the select entry as containing an error. If the select entries were being entered interactively, then the invalid entry can be reentered correctly. Otherwise, Fastback will terminate with end of task code 7 (i.e., if the select entries were not being entered interactively) when all select entries have been processed.

Required Operator Action:

Either interactively or otherwise, correct the erroneous select entry.

INVALID DEVICE/VOLUME NAME

Meaning:

A device or volume name which the user specified to Fastback is invalid. For example, DISC=BADNAME is invalid since BADNAME will be interpreted as a volume name and is longer than the maximum allowed 4 characters.

Program Action:

Fastback will abort if operating in batch or immediate modes, but will reissue the appropriate command prompt if running in interactive mode.

Required Operator Action:

None

INVALID FILE DESCRIPTOR

Meaning:

An fd specified to a command has an invalid format; e.g., in the command LIST=55555:1234.XXX, the volume name (55555) begins with a numeric character and is too long (5 characters), etc. Thus, the fd is invalid.

Program Action:

If Fastback was started in batch or immediate modes, it will abort; else, it will reissue the appropriate prompt message.

Required Operator Action:

None

INVALID FILE DESCRIPTOR TRANSFORMATION IN RENAME
SELECT MASK=sel-mask, RENAME MASK=ren-mask
^ ^

Where:

sel-mask	is the file selection mask, entered at the start of the select entry. It is output in the format in which it is held internally. Thus if no volume field was specified by the user, the volume field will be output as ****. The same applies for other fields of the file descriptor mask. For a more detailed description please refer to Chapter 4.
ren-mask	is the file descriptor mask, which was specified to a RENAME parameter in a select entry. It is output in the format in which it is held internally (see above note on select-mask). For a more detailed description please refer to Chapter 4.
^	is an up-arrow. An up-arrow is generated under each file descriptor mask, to point to the corresponding characters in each mask where the inconsistency occurred.

Meaning:

During the processing of select entries, Fastback has encountered one where the given RENAME file descriptor is not allowed (i.e., invalid). The RENAME file descriptor is syntactically correct but is not allowed as a RENAME mask for the select file descriptor because obeying the RENAME mask could result in files with invalid file descriptors being created. For a more detailed description (including what is and what isn't allowed), please refer to Chapter 4.

Program Action:

If the select entries were being entered interactively, then the user may reenter the invalid select entry. Otherwise, Fastback will process all remaining select entries and will then terminate with end of task code 7.

Required Operator Action:

Correct the invalid select entry.

INVALID PARAMETER

Meaning:

This message is output when an invalid/unrecognisable parameter has been specified with a Fastback command; e.g., in the command POSITIONING=NONE, the parameter NONE is invalid since it is not one of the allowable parameters of REWIND, NOREWIND, or nn filemarks.

Program Action:

If Fastback was started in interactive mode, it will re-issue the appropriate prompt; otherwise, it will terminate.

Required Operator Action:

None

INVALID PROTECTION KEYS

Meaning:

While processing select entries, Fastback has encountered one which uses the KEYS parameter and whose specified keys are not valid. The protection keys must be hexadecimal digits in the range 0000 to FFFF.

For example, the parameter ....KEYS=XYZ.... is not valid.

Program Action:

If Fastback is reading the select entries interactively, then the user may reenter the erroneous select entry. Otherwise, Fastback will process all remaining select entries and then terminate with end of task code 7.

Required Operator Action:

Correct the erroneous select entry.

INVALID SEND MESSAGE IGNORED

Meaning:

The user has sent a message to Fastback via the SEND..... command. The message received is neither STOP or PAUSE and, hence, is invalid.

Program Action:

Fastback simply ignores the message.

Required Operator Action:

None

KEYWORD CONFLICT WITH PREVIOUS ENTRY

Meaning:

This message is generated when Fastback is operating in batch or immediate mode, and the user has specified two (or more) conflicting commands. For example, the conflicting START parameters ....,BACKUP,....,RESTORE,.... will cause the abovementioned message to be generated.

Program Action:

Fastback will terminate with end of task code 2.

Required Operator Action:

None

LIST DEVICE ASSIGNED TO fd

Where:

fd	is the file descriptor of the system console (or the user's MTM terminal).

Meaning:

No list device was specified, so Fastback attempted to assign device PR: as the list device. However, the attempted assignment failed, so Fastback has assigned the list device to the console.

Program Action:

Fastback continues the specified operation with the console as the list device.

Required Operator Action:

None

MISSING PARAMETER

Meaning:

The user has started Fastback in either batch or immediate modes and has not specified a parameter with a command which requires a parameter. For example, if Fastback is started with the command DISC=,....., then Fastback will complain that the disc pack/volume name is missing.

Program Action:

Fastback will terminate with end of task code 2.

Required Operator Action:

None

NAME FIELD OF FILE DESCRIPTOR TEMPLATE IS TOO LONG

Meaning:

This message is given if the user specifies a valid file descriptor whose name field is longer than 2 characters to the TEMPLATE command. Hence TEMPLATE=RI is acceptable, but TEMPLATE=RKI is not. For further information, please refer to the description of the TEMPLATE command given in Chapter 3.

Program Action:

Fastback will reissue the ....TEMPLATE.... prompt if it is executing in interactive mode; otherwise, it will terminate.

Required Operator Action:

None

NO DEFAULT ALLOWED

Meaning:

This message is output when Fastback is operating in interactive mode, and the user has attempted to default (i.e., just hit carriage return in response to a prompt) the reply to a nondefaultable prompt. (Nondefaultable prompts are those which contain no # character, which is used to identify default prompt responses.) For example, the first question (FUNCTION.....?) in the interactive dialogue does not accept default responses.

Program Action:

After giving the above mentioned message, Fastback again issues the prompt concerned.

Required Operator Action:

Enter a valid response to the prompt.

NO FILE DESCRIPTOR SPECIFIED

Meaning:

A file descriptor was not specified where one was expected. In particular, a select entry was read with RENAME=.... specified; and no file descriptor was found after the RENAME parameter.

Program Action:

If Fastback is reading select entries from an interactive device, the user may reenter the correct select entry. Otherwise, Fastback will process all remaining select entries and then terminate with end of task code 7.

Required Operator Action:

Correct the erroneous select entry.

NO FILES MATCHED THE FOLLOWING SELECT ENTRIES:

Meaning:

Fastback is about to output all select entries which did not match any file on the input medium (i.e., disc or tape). The select entries are decoded from their internally-held format and are output. Select entries may not be output in the same format as they were entered. For more information on this, please refer to Chapter 4.

Program Action:

Fastback outputs the select entries which did not match any files on the input medium. This is not an error condition. Fastback is simply reporting those select entries which never matched any files.

Required Operator Action:

None

NULL VOLUME NAME NOT ALLOWED

Meaning:

The user has specified a file descriptor mask in a select entry with a null volume name (e.g., :ASDG-.-/-).

Program Action:

If Fastback is reading select entries from an interactive device, then the user may reenter the correct select entry. Otherwise Fastback will continue to process all remaining select entries, and will then terminate with end of task code 7.

Required Operator Action:

Correct the erroneous select entry.

OPERATING SYSTEM IS Rnn-nn - Rmm-mm REQUIRED

Where:

nn-nn	is the revision and update levels of the release of OS/32 under which Fastback was loaded and started.

mm-mm — is the revision and update levels of the release of OS/32 that Fastback requires in order to execute.

Meaning:

Fastback usually requires a higher release level of OS/32 for it to execute.

Program Action:

Fastback terminates with end of task code 2.

Required Operator Action:

Boot a higher release of OS/32 (if this is possible).

OS/32 FASTBACK Rnn-nn function OF VOLUME voln STARTING

Where:

nn-nn — is the revision and update levels of Fastback.

function — is the user's requested function, except display (i.e., backup, restore, or verifyonly.)

voln — is the volume name of the disc pack involved in the requested operation.

Meaning:

The above message is logged to the system console when Fastback begins the requested operation, provided that the operation is not a display. The message is also not logged if Fastback is executing from the console itself.

Program Action:

Fastback executes the requested operation.

Required Operator Action:

None

PACK ADMINISTRATION FILE LAST UPDATED ON dd/mm/yy hh:mm:ss
ADJUST SYSTEM DATE/TIME IF REQUIRED, THEN CONTINUE

Where:

dd/mm/yy	is the date when the pack administration file (PACKINFO.DIR) was last updated (i.e., the date was fetched from OS/32 when the file was last updated). The date is given in U.S. format if OS/32 is sysgened with the U.S. date option; otherwise, the date is given in European format.
hh:mm:ss	is the time of day when the pack administration file was last updated.

Meaning:

Fastback has fetched the date and time from OS/32 and determined that the current date/time is earlier than the date/time at which the pack administration file was last updated. Thus, either the date/time is currently set incorrectly, or it was at the time when PACKINFO.DIR was last updated.

Program Action:

The program pauses to allow the operator to correct the current date/time setting in case the current setting is incorrect. When Fastback is continued, it refetches the date and time and uses the values obtained as the current date and time.

Required Operator Action:

Check that OS/32 has the correct date and time set. If not, correct the date/time setting. Then continue Fastback.

PARAMETER TOO BIG

Meaning:

This message indicates that a numeric parameter is too large. For example, the command BUFFERSIZE=37 would invoke the above response since the maximum allowed buffer size is 32KB.

Program Action:

If Fastback was started in interactive mode, it will reissue the appropriate prompt otherwise, it will terminate.

Required Operator Action:

None

PLEASE ENTER FILE SELECTION CRITERIA

Meaning:

Fastback has been started with the SELECT=interactive device, therefore Fastback has output to the device an invitation to enter the select entries.

Program Action:

Fastback waits for the user to enter the select criteria. See also Chapter 4.

Required Operator Action:

Enter the file select criteria.

PLEASE MOUNT TAPE NUMBER nn. CONTINUE FASTBACK WHEN READY

Where:

nn	is the (decimal) number of the next reel of tape required in a multi-reel operation.

Meaning:

Fastback has finished with the current tape. It now needs the next tape in the multi-tape sequence.

Program Action:

Fastback pauses.

Required Operator Action:

Mount the next tape in the sequence.

PLEASE POSITION TAPE TO FASTBACK'S START POINT, AND CONTINUE

Meaning:

Fastback has previously requested the operator to (re)mount the first tape in a multi-tape sequence. Having continued after the tape was mounted, Fastback finds that the POSITIONING option is set to NOREWIND. Since it therefore cannot automatically position the tape to the correct start point (because it doesn't know where the start point is), Fastback generates the above message and will then wait for the operator to manually position the tape.

Program Action:

Fastback pauses.

Required Operator Action:

Position the tape to the correct start point of the Fastback backup on the tape, then continue Fastback.

REQUESTED FASTBACK OPERATION ABORTED - EOT CODE = nnn

Where:

nnn	is the (decimal) end of task code. Fastback is about to terminate with this end of task code.

Meaning:

Fastback is aborting the requested operation, and is about to terminate with the given end of task code. The reason for the termination will usually have been indicated by a previous message.

Program Action:

Fastback will terminate with the given end of task code.

Required Operator Action:

None

REQUESTED FASTBACK OPERATION COMPLETED OK

Meaning:

Fastback has completed the requested operation, and is about to terminate. This message will not be output if Fastback aborts the requested operation (e.g., due to errors, etc.).

Program Action:

Fastback will terminate.

Required Operator Action:

None

SIZE MISMATCH IN BLOCK nnn OF DIRECTORY/INDEX FILE
ACTUAL SIZE=mmmmm BYTES,
SIZE EMBEDDED IN BLOCK=11111 BYTES

Where:

nnn	is the (decimal) tape block number of the block which contains the size error.
mmmmm	is the (decimal) number of bytes actually read for the block.
11111	is the (decimal) number of bytes embedded in the block header. This should always be equal to the size of the block (i.e., equal to the number of bytes actually read for the block).

Meaning:

While reading the directory/index file from tape during a restore, verifyonly, or (tape) display operation, Fastback has read a block whose embedded size field does not agree with the number of bytes actually read for the block.

Program Action:

Fastback will abort with end of task code 22 since something is obviously wrong with the tape block. Any intermediate files, etc., will be deleted.

Required Operator Action:

None

SPECIFIED VOLUME NAME DOES NOT MATCH INPUT DISC VOLUME

Meaning:

As part of a select entry, the user has specified a volume name which does not match that of the current input disc. The current operation must be either a backup or a (disc) display.

For example, if a backup of disc DEMO were requested then: (1) a select entry with volume name D*M-: is valid, but (2) a select entry with volume name D*X-: is not because the X does not match the M in the input disc volume name. See Chapter 4.

Program Action:

If Fastback is reading select entries from an interactive device, then the user may reenter the select entry correctly. Otherwise, Fastback will process all remaining select entries and will then terminate with end of task code 7.

Required Operator Action:

Correct the erroneous select entry.

UNEXPECTED TASK QUEUE ENTRY xxxxxxxx IGNORED

Where:

xxxxxxxx	is a (hexadecimal) dump of the unexpected task queue entry.

Meaning:

Fastback has found an entry on its task queue whose reason code is unrecognisable.

Program Action:

Fastback ignores the task queue entry.

Required Operator Action:

This message should never be generated, so it may be worthwhile repeating the current Fastback operation. If the message is generated again then the operator should submit an SCR against Fastback.

UNEXPECTED TRAILING CHARACTER

Meaning:

A Fastback command has been parsed without error, but there is some character other than a separator following the command; e.g., ONERROR=ABORT AND SOME TRAILING JUNK.

Program Action:

If Fastback was started in interactive mode, it will reissue the appropriate prompt; otherwise, it will terminate.

Required Operator Action:

None

UNEXPECTED TRAILING CHARACTER(S) IGNORED

Meaning:

A command has been specified to Fastback that contains trailing characters. These trailing characters are such that Fastback is still able to interpret and obey the given command, so the above message is given to indicate that the trailing characters have been seen and ignored.

Program Action:

Fastback continues as if no trailing characters had been encountered in the command.

Required Operator Action:

None

UNRECOGNISABLE 80 BYTE RECORD READ
**** TAPE WAS NOT CREATED BY EITHER FASTBACK OR OS/32 BACKUP
**** SUGGEST USE OS/32 COPY TO EXAMINE CONTENTS OF TAPE
PLEASE REPOSITION TAPE, OR MOUNT CORRECT TAPE, AND CONTINUE

Meaning:

While processing the header labels for a new tape during a display, restore, or verify only operation, Fastback has read an 80-byte record which it cannot recognise. That is, it is not a Fastback header label and, in the case of it being the first record on a tape, it is not an OS/32 BACKUP-type header label.

Program Action:

Fastback will pause to allow the operator to intervene.

Required Operator Action:

Reposition the tape to the correct point, mount the correct tape, etc.

UNRECOGNISABLE KEYWORD

Meaning:

Fastback cannot recognise a particular command that the user has specified. For example, if Fastback were to be driven by START arguments, then the above message would be invoked by the start sequence START ,ASDG,BACKUP.... This is because the keyword ASDG is not recognisable to Fastback.

Program Action:

If Fastback is operating in interactive mode, it will reissue the appropriate command prompt; otherwise, it will terminate.

Required Operator Action:

None

VOLUME NAME INVALID

Meaning:

This message is given if the user specifies an invalid disc volume name to Fastback. For example, the command DISC=2345 will cause the message to be generated since volume names must not begin with a numeric character.

Program Action:

If Fastback was started in interactive mode, it will reprompt the user. Otherwise, Fastback will terminate.

Required Operator Action:

None

VOLUME NAME MUST NOT BE SPECIFIED

Meaning:

This message is given if the user specifies a volume name in the file descriptor and specifies it to the TEMPLATE command. The TEMPLATE command will not accept file descriptors containing a volume name (since the volume name is effectively that of the output disc).

Program Action:

Fastback will terminate if executing in batch or immediate modes. If operating in interactive mode, Fastback will again issue the ....TEMPLATE.... prompt.

Required Operator Action:

None

VOLUME NOT FOUND IN SYSTEM

Meaning:

The user has apparently specified a volume name to the DISC or WORKSPACE command, and Fastback cannot find any (marked on) disc in the system whose volume name matches that specified. For example, the message could be generated if the command DISC=ASDG were specified, and there was no marked on disc whose volume name was ASDG.

Program Action:

Fastback will terminate if executing in batch or immediate modes; otherwise, it will reprompt the user.

Required Operator Action:

None

WARNING: BACKED-UP FILE fd IS ASSIGNED FOR WRITE

Where:

fd	is the file descriptor of the file which is currently being backed up.

Meaning:

Fastback is performing a backup operation, and has started to copy the file's directory entry (and index blocks for indexed files) to tape. Fastback has discovered that the file is currently assigned for writing, and thus Fastback will only backup the current image of the file.

For contiguous files, the current image is that which is present on the disc when Fastback later copies the actual data sectors to tape.

For indexed files, the current image consists of the number of logical records in the file as indicated by the directory entry when Fastback processed that directory entry. This is the number of logical records at which the file was last checkpointed. Thus, for example, if Fastback is to back up a newly created indexed file which is currently being written to, then zero records of the file will be copied since the directory entry of the file will indicate that the file is assigned for write but currently has no records.

Program Action:

Fastback will continue the backup operation.

Required Operator Action:

None

WARNING: CONTINUING AFTER DIRECTORY I/O ERROR. TREATING AS END OF DIRECTORY

Meaning:

During a backup operation, Fastback has encountered an I/O error while reading the (input) disc's directory. (A standard I/O error message will have been generated.) This message is given only if the ONERROR action is set to SKIP.

Program Action:

Fastback will continue the backup operation as if it had encountered the end of the disc directory. Thus, only those files seen so far will be backed up.

Required Operator Action:

None

WARNING: DATA POINTERS FOR FILE fd ARE MISSING OR CORRUPT
FILE TRUNCATED TO nn RECORDS

Where:

fd	is the file descriptor of the file whose data pointers are missing or corrupt.
nn	is the number of (logical) records to which the (indexed) file has been truncated.

Meaning:

Fastback has been processing the directory (i.e., FAT/INDEX) file on an input tape. It has encountered a directory entry for an indexed file (identified by the above message) which states that the file has some number of logical records. When Fastback then processes the data pointers for the indexed file (which follow the directory entry on the tape), it finds that some of the pointers are missing or corrupted. For instance, Fastback might have calculated that an indexed file required 100 data pointers to map it. However, the tape only contained 95 data pointers, indicating that something went wrong during the original backup of the file (such as the file having corrupt data pointers on the original backed up disc).

Program Action:

Fastback calculates how many logical records of the file it can restore/verify, given the number of valid data pointers that it has. The file is then treated as if it had only this number of records.

Required Operator Action:

None

WARNING: DIRECTORY DATA MISMATCH FOR FILE fd
DISC DIRECTORY DATA DOES NOT MATCH DATA READ FROM TAPE
FILE fd WILL NOT BE function

Where:

fd	is the file descriptor of the file whose directory information has mismatched the data read from tape.
function	will either be RESTORED or VERIFIED, depending on whether the current function being performed is a restore or a verifyonly.

Meaning:

The meaning of the message really depends on whether the operation is a restore or a verifyonly.

During restore operations, Fastback allocates and prefills intermediate files for all files that are to be restored. It remembers the file sizes, etc., of these intermediate files. Having allocated and prefilled these files, Fastback searches for their directory entries on the output disc. When Fastback finds a directory entry for a file whose name matches one of the files it is looking for, Fastback checks that the file type and size, etc., match the remembered values from the time the file was allocated and prefilled. The above message is generated if a mismatch occurs.

For verifyonly operations, for every file to be verified on tape, Fastback must find a file on the disc whose name, etc., matches the file held on tape. The above message is given if Fastback finds a file whose name matches that of a file to be verified, but the file type or blocking factors, etc., are different.

Program Action:

The indicated file is not restored or verified.

Required Operator Action:

None

WARNING: DIRECTORY SECTOR AT yyyyyy HAS INVALID FORWARD POINTER xxxxxxxx
TREATING AS END OF DIRECTORY
[RESUMING FILE SEARCH AT START OF DIRECTORY]

Where:

yyyyyy	is the (hexadecimal) logical block address of the directory sector whose forward pointer is invalid.
xxxxxxxx	is a (hexadecimal) dump of the invalid forward chain pointer.

Meaning:

During a backup, display, restore, or verifyonly operation, Fastback has been processing directory entries and has come to the end of some directory sector. When it fetches the forward pointer in preparation for reading the next directory sector, it finds that the value of the pointer exceeds the number of sectors on the disc pack.

The qualifying message Resuming file search..... is only given if Fastback is performing a restore or verifyonly operation, since both of these operations involve searching the disc directory for specific files to restore (i.e., intermediate files) or verify.

NOTE

The generation of the above messages does not definitely mean that the disc directory chain is broken. If Fastback is reading along the directory chain while OS/32 is extending the directory chain (e.g., some task has allocated a file that causes OS/32 to discover that there are no free slots in the directory; hence OS/32 obtains a free sector and links it into the directory chain), Fastback may follow a forward pointer to a directory sector which has not yet been flushed out from OS/32's buffer. Fastback will then examine this sector as if it were a directory sector and will follow this sector's forward pointer. If this pointer contains rubbish (an odds-on chance), then Fastback may well experience the invalid forward pointer.... phenomenon previously outlined.

Program Action:

Fastback will continue with the operation as if it had found the end of the directory.

Required Operator Action:

If the warning was generated due to a real break in the directory chain (to check for a broken directory chain, do a DISPLAY FILES and see if it lists all of the files that ought to be on the disc; e.g., if there ought to be over 500 files on the disc and the DISPLAY FILES only lists 100, then the directory chain is broken), then OS/32 FASTCHEK should be run on the disc, initially with the REPORTONLY option set. If the disc was initialised with a sufficiently large preallocated directory, then the next directory sector following the break in the chain can be easily found since the directory sectors will have been allocated at regular intervals around the disc.

WARNING: DISC FILE fd HAS A DATE/TIME LAST WRITTEN
DIFFERENT TO THAT OF CORRESPONDING TAPE FILE

Where:

fd	is the file descriptor of the disc file whose date/time last written does not match that obtained from tape.

Meaning:

Fastback is running a verifyonly operation. For the given file, Fastback knows that it must verify it and so has found the directory entry for the file of the same name on the output disc. Fastback then checks that the file types, etc., match before it does the verify. However, the date/time last written for the disc file does not match that of the tape file, so Fastback generates the above message.

Program Action:

Fastback will go ahead and attempt to verify the given file (even though the last written date/times do not match) because the file types, number of records, etc., do match.

Required Operator Action:

None

WARNING: EXPECTED FILE fd NOT FOUND
FILE fd WILL NOT BE VERIFIED

Where:

fd — is the file descriptor of the missing disc file.

Meaning:

Fastback is executing a verifyonly operation and cannot locate a file on the disc whose name matches that of a file read from tape. For instance, file FRED was read from tape but there is no file called FRED on the disc to verify the tape file against.

Program Action:

Fastback will continue with the verifyonly pass, but will not attempt to verify the identified file.

Required Operator Action:

None

WARNING: EXPECTED INTERMEDIATE FILE int-fd NOT FOUND
CORRESPONDING FILE cor-fd WILL NOT BE RESTORED

Where:

int-fd — is the file descriptor of the missing intermediate file. If the default intermediate file TEMPLATE is in use, then the file descriptor will be of the form FBnnnnnn.INT/act. See also the description of the TEMPLATE command in Chapter 3.

cor-fd — is the file descriptor of the real file which the intermediate file represents on the output disc.

Meaning:

During a restore operation, Fastback allocates an intermediate file on the output disc for every file which is to be restored. It then scans the disc directory for the entries for each intermediate file so that it can determine where each intermediate file is located on the disc.

The above message is given because, when Fastback scans the disc directory for the entry for a given intermediate file, it cannot find a directory entry for the intermediate file (e.g., this condition would occur if the user deleted an intermediate file shortly after Fastback had allocated it, and before it walked down the disc directory and found the file's entry).

Program Action:

Fastback will continue the restore operation, but will not restore the contents of the identified real file.

Required Operator Action:

None

WARNING: EXPECTED INTERMEDIATE FILE int-fd NOT FOUND
**** INTERMEDIATE FILE int-fd NOT RENAMED NOR REPROTECTED
(ORIGINAL NAME WAS orig-fd)

Where:

int-fd	is the file descriptor of the missing intermediate file.
orig-fd	is the original name of the file. That is, it is the name of the file as read from tape, not the name which may have been specified by a RENAME command in a select entry.

Meaning:

Fastback is performing a restore operation and is currently renaming intermediate files to their correct (i.e., target) names. The intermediate file representing the original file cannot be found on the output disc, so Fastback cannot rename it to the desired target filename.

Program Action:

Fastback will continue the restore, renaming all found intermediate files to their target names. The identified original file is not restored since something has happened to its corresponding intermediate file.

Required Operator Action:

Check if someone deleted the intermediate file.

WARNING: FILE TYPE MISMATCH FOR FILE fd
DISC DIRECTORY DATA DOES NOT MATCH DATA READ FROM TAPE
FILE fd WILL NOT BE function

Where:

fd	is the file descriptor of the file which has the type mismatch.
function	is either RESTORED or VERIFIED, depending on whether the current operation is a restore or a verifyonly.

Meaning:

Fastback is performing a restore or a verifyonly operation. According to the information read from tape, the given file should be an indexed or contiguous file, but the disc directory entry for the file indicates that it is not the expected type (i.e., indexed or contiguous).

For verifyonly operations the mismatch is between the file on tape and the corresponding file on disc which is to be verified.

For restore operations the mismatch is found during Fastback's second pass of the FAT/index file on tape. During the first pass of this tape file, the intermediate files are allocated and prefilled on the output disc. During the second pass, the directory entries for the intermediate files are examined. It is during this process that Fastback has located an intermediate file which it thought should be indexed/contiguous, but the directory entry indicated that it was contiguous/indexed.

For example, this situation could occur if, during a restore operation, an intermediate file was deleted and then reallocated with the same name but as a different type of file. This deletion/allocation would have to have occurred between Fastback's passes 1 and 2 of the FAT/index file on the input tape.

Program Action:

Fastback will skip restoring/verifying the given file.

Required Operator Action:

None

WARNING: FILE fd HAS FEWER DATA POINTERS THAN EXPECTED
FILE TRUNCATED TO nnnnnn RECORDS

Where:

fd	is the file descriptor of the file which is missing some data pointers.
nnnnnn	is the number of records in the file which will be restored/verified.

Meaning:

This message is given when Fastback is reading the directory/index file from tape for a restore or verifyonly. From the directory information Fastback expects the file to contain some number of logical records (and hence it expects a certain number of data pointers), but the number of data pointers found on the tape (for this particular file) is insufficient to map the expected number of records.

For example, the directory entry for an indexed file might indicate that the file contains 100 records (mapped by, for instance, 15 data pointers). If Fastback finds only eight data pointers on the input tape, then only those (55 odd) records mapped by the eight data pointers can be restored/verified.

Program Action:

Fastback will continue the restore/verifyonly but will only restore/verify the indicated number of records for the given file.

Required Operator Action:

None

WARNING: FILE fd TRUNCATED TO nn RECORDS

Where:

fd	is the file descriptor of the file whose backed up copy is to be truncated.
nn	is the (decimal) number of logical records to which the backed up copy of the indexed file will be truncated.

Meaning:

When Fastback processes an indexed file during a backup operation, it reads the file's directory entry and then the file's index blocks. If it encounters an error (such as an I/O error reading an index block, or an invalid index or data pointer), then Fastback outputs a message describing the error and terminates processing the indexed file. The backup copy of the file will hence be limited to the number of logical records that can be mapped by the index blocks that Fastback had already processed for the file. Thus the backed up copy of the file is truncated to some number of records, as given by the above message. Fastback does not attempt to repair the indexed file on the input disc.

Program Action:

Fastback continues with the backup operation, but cannot back up all records from the indexed file.

Required Operator Action:

None, except that OS/32 Fastchek should be run on the disc when the backup is completed. Fastchek should first be run with the REPORTONLY option, to check the state of the files on the disc.

```
WARNING: ILLEGAL FILE DESCRIPTOR CREATED BY RENAMING
         ORIGINAL  FD=fd1, RENAME MASK=fd-mask
         RESULTANT FD=fd2
         **** ORIGINAL FD RETAINED ****
                        or
         **** FD OF  INTERMEDIATE FILE RETAINED ****
                        or
         FILE fd1 WILL NOT BE VERIFIED
```

Where:

fd_1	is the original file descriptor of the file.
fd-mask	is a (select entry) RENAME mask used to control the renaming of a file.
fd_2	is the (illegal) file descriptor generated when the rename mask is applied to the original file descriptor.

Meaning:

During a backup, display, restore, or verifyonly operation, a file (given by the original fd) has matched a select entry which contains a rename mask. When the rename mask has been applied to the original fd, the resultant file descriptor is illegal. For further information, please refer to Chapter 4 of this manual.

Program Action:

If the operation is a restore, Fastback will continue but the appropriate intermediate file will not be renamed to either the original fd nor the resultant fd at the end of the restore. The intermediate file will be left, but will be identified to the user by a later message.

For backup operations, the original fd will be retained.

Finally, for verifyonly operations, the file will not be verified (since Fastback cannot be sure of the name of the file to be verified against on the output disc).

Required Operator Action:

None

WARNING: INSUFFICIENT DISC SPACE TO PRE-FILL INTERMEDIATE
FILE int-fd
CORRESPONDING FILE real-fd TRUNCATED TO nnnnnn RECORDS

Where:

int-fd	is the file descriptor of the intermediate file which was being prefilled.
real-fd	is the file descriptor of the file which is represented on the disc by the intermediate file. That is, the name to which the intermediate file will ultimately be renamed (i.e., the target filename).
nnnnnn	is the number of logical records to which the file (both intermediate and target) has been truncated.

Meaning:

Fastback is running a restore operation and is prefilling an indexed intermediate file. While writing the dummy prefill records to the file, Fastback received an SVC 1 end of medium (EOM) status in response tc a record write. This means that there was insufficient contiguous disc space available for OS/32 to allocate another index or data block for the file. Hence, Fastback will leave the intermediate file containing only as many records as had already been written into it. For example, the file is truncated because not all required records could be written to it.

Program Action:

Fastback will continue the restore operation (it will continue to try to allocate and prefill any more needed intermediate files).

NOTE

It does give up the restore (i.e., the creation of intermediate files) because the prefill which resulted in EOM status may have been for a heavily blocked indexed file. Thus, there may still be sufficient contiguous disc space to restore other files.

Required Operator Action:

None

WARNING: INSUFFICIENT MEMORY FOR ll KB BUFFERS
USING mm KB BUFFERS

Where:

ll	is the buffer size that the user had requested Fastback to use (default is 16kb).
mm	is the buffer size which Fastback has chosen to use because of memory size limitations.

Meaning:

Fastback has been loaded with insufficient memory (i.e., segment size increment) to support the requested/default buffer size. It has decremented the buffer size through its allowable values until it has found a buffer size which it has sufficient memory to support. Fastback has then begun the requested (backup) operation with its chosen buffer size.

Program Action:

Fastback has begun the backup operation, using its chosen buffer size.

Required Operator Action:

None

WARNING: INVALID FILE TYPE - FILE fd
IN DIRECTORY BLOCK AT yyyyyy OFFSET zz
FILE WILL NOT BE COPIED

Where:

fd	is the file descriptor of the file whose type is invalid.
yyyyyy	is the (hexadecimal) sector address of the directory sector that contains the file's directory entry.
zz	is the (hexadecimal) offset into the directory sector of the file's directory entry.

Meaning:

During a backup (or disc display) operation, Fastback has encountered a directory entry for a file whose file type is invalid. That is, the directory entry's attributes byte indicates that the file is neither contiguous, indexed, non-buffered indexed, long record length indexed, nor extendable contiguous.

Program Action:

Fastback will continue with the requested operation, but will ignore the indicated file.

Required Operator Action:

OS/32 FASTCHEK should be run on the disc to check the integrity of the disc directory and files. FASTCHEK should preferably be run in REPORTONLY mode first.

WARNING: ORIGINAL INPUT DISC SECTOR xxxxxx NOT COPIED TO TAPE
(MAPS TO SECTOR yyyyyy ON CURRENT DISC)
CONTENTS OF SECTOR CANNOT BE RESTORED/VERIFIED

Where:

xxxxxx — is the (hexadecimal) logical block address of the sector on the original backed up disc, which could not be copied to tape.

yyyyyy — is the (hexadecimal) logical block address of the sector on the current disc to which the original sector contents were to have been restored, or verified against.

Meaning:

During a restore or verifyonly operation, Fastback has encountered a data sector on the input tape, whose contents were not copied to the tape during the original backup operation. Hence the contents of that sector (whichever file it belongs to) cannot be restored or verified.

Program Action:

Fastback will continue the restore or verifyonly operation, skipping the contents of the given sector.

Required Operator Action:

When Fastback finishes, it will search for the file which owns the given sector. If such a file is found, the operator should check the contents of the identified file.

WARNING: TAPE DATA BLOCK HAS INCORRECT HEADER WORD CHECKSUM
COMPUTED CHECKSUM=xxxxxxxx, EXPECTED yyyyyyyy
CONTENTS OF TAPE DATA BLOCK WILL NOT BE
RESTORED/VERIFIED

Where:

xxxxxxxx — is the (hexadecimal) checksum that was computed from the tape block contents.

yyyyyyyy — is the (hexadecimal) checksum that was imbedded in the tape block, and that Fastback expected the computed checksum to match.

Meaning:

Fastback is reading the actual data file from tape (as opposed to the FAT/Index file), and has read a record whose header word checksum does not match that calculated from the block.

Program Action:

Fastback will continue the current operation but will not attempt to use any data from the tape block in question. Hence for verifyonly operations and verify passes of restore operations, the data in the tape block will not be verified. If the incorrect header word checksum was found during the "write data from tape to disc" phase of a restore operation, then Fastback will not restore any data from the tape block. If this occurs, and Fastback later makes a verify pass of the tape during which it manages to read the given tape block correctly, then Fastback will generate nonverify error messages for all the sectors that should have been restored from the tape block but were not.

If the operation is a backup verify, then the contents of the tape block will not be verified.

Required Operator Action:

If the operation is a backup verify, then the backup operation should be repeated if possible.

WARNING: fd - INVALID DATA BLOCK ADDRESS xxxxxxxx
IN INDEX BLOCK AT yyyyyy OFFSET zz

Where:

fd — is the file descriptor of the indexed file on the input disc that contains the invalid data pointer.

xxxxxxxx — is a hexadecimal dump of the invalid data pointer, as obtained from the (indexed) file's index block.

yyyyyy — is the (hexadecimal) logical block address of the index sector that contains the invalid data block pointer.

zz — is the (hexadecimal) offset into the index sector, where the invalid data pointer is located.

Meaning:

During a backup operation, Fastback was reading an indexed file's index block chain, and, while processing the contents of an index block, it found an invalid data pointer (i.e., a pointer too large for the type of disc or a pointer of zero).

Program Action:

Fastback will not process the index chain any further and will truncate the backed up copy of the involved file. Hence this message is followed by the File truncated.... message. Please refer to the description of that message.

Required Operator Action:

OS/32 Fastchek should be used to check/repair the integrity of the disc pack. Note that Fastchek should first be run with the REPORTONLY option so that it does not delete any files, etc.

```
WARNING: fd - INVALID FIRST INDEX BLOCK ADDRESS xxxxxxxx
         IN DIRECTORY BLOCK AT yyyyyy OFFSET zz
         FILE TRUNCATED TO 0 RECORDS
```

Where:

fd — is the file descriptor of the indexed file whose first index block pointer is corrupt.

xxxxxxxx — is the file's first index block pointer, as obtained from the directory entry.

yyyyyy — is the (hexadecimal) logical block address of the directory sector containing the file's directory entry.

zz — is the (hexadecimal) offset into the directory sector of the file's directory entry.

Meaning:

Fastback has accessed the directory entry for an indexed file during a backup operation. It has fetched the pointer to the file's chain of index blocks and found the pointer to be either zero or a value which exceeds the number of sectors on the disc pack.

Program Action:

Since the pointer is invalid, Fastback cannot read the file's index blocks, so it cannot backup any records from the file. Fastback thus backs up the file as if it contained no (i.e., zero) records. Apart from this, the backup operation continues normally.

Required Operator Action:

The disc should be checked with OS/32 Fastchek, initially utilising the REPORTONLY option so that no file deleting, etc., takes place.

WARNING: fd - INVALID FIRST SECTOR ADDRESS xxxxxxxx
IN DIRECTORY BLOCK AT yyyyyy OFFSET zz
FILE WILL NOT BE COPIED

Where:

fd	is the file descriptor of the contiguous file whose starting sector address is invalid.
xxxxxxxx	is the invalid starting sector address for the contiguous file, as obtained from the disc directory.
yyyyyy	is the logical block address of the directory sector that contains the entry for the contiguous file.
zz	is the (hexadecimal) offset into the directory sector of the file's directory entry.

Meaning:

During a backup operation, Fastback has begun to process the directory entry for some contiguous file. It has fetched the starting address of the file's contiguous extent on the disc, and the starting address is either zero or some value exceeding the number of sectors on the pack.

Program Action:

Fastback will not backup the identified file but will otherwise continue processing as normal.

Required Operator Action:

OS/32 Fastchek should be run on the disc to check the integrity of directory and files. Fastchek should initially be run with the REPORTONLY option set.

```
WARNING: fd - INVALID LAST SECTOR ADDRESS xxxxxxxx
         IN DIRECTORY BLOCK AT yyyyyy OFFSET zz
         FILE WILL NOT BE COPIED
```

Where:

fd	is the file descriptor of the contiguous file that has an invalid last sector address.
xxxxxxxx	is a (hexadecimal) dump of the invalid last sector address, as obtained from the file's directory entry.
yyyyyy	is the (hexadecimal) sector address of the directory sector which contains the file's directory entry.
zz	is the (hexadecimal) offset into the directory sector of the file's directory entry.

Meaning:

During a backup operation, Fastback has encountered a directory entry for a contiguous file where the pointer to the last sector in the file is either zero or a value exceeding the number of sectors on the pack.

Program Action:

Fastback will continue the backup operation but will not backup the indicated file.

Required Operator Action:

OS/32 Fastchek should be run on the disc to check the integrity of the directory and files.

WARNING: fd - INVALID NEXT INDEX BLOCK ADDRESS xxxxxxxx
IN INDEX BLOCK AT yyyyyy OFFSET zz

Where:

fd	is the file descriptor of the file containing the invalid forward index block pointer.
xxxxxxxx	is the invalid forward index block pointer given in hexadecimal.
yyyyyy	is the (hexadecimal) logical block address of the index sector containing the invalid forward pointer.
zz	is the (hexadecimal) offset in the index block where the invalid pointer may be found. This should always be set to 4.

Meaning:

While processing the index blocks for an indexed file during a backup operation, Fastback has read a new index block and found it to contain a forward chain pointer whose value is too large for the type of disc or is zero (= end of chain) when Fastback believes more index blocks should follow.

Program Action:

Fastback will stop processing the index blocks, and hence the copy of the file which will be written to tape will only contain those records which were identified by those index blocks which Fastback had already processed. Hence the backup copy of the file will be truncated to some (incomplete) number of logical records. This message is followed by the File truncated.... message. Please refer to the description of that message.

Required Operator Action:

OS/32 Fastchek should be run on the disc, preferably with the REPORTONLY option set.

WARNING: fd - INVALID PREVIOUS INDEX BLOCK ADDRESS xxxxxxxx
IN INDEX BLOCK AT yyyyyy OFFSET zz

Where:

fd	is the file descriptor of the file whose index chain contains an invalid previous index block address.

xxxxxxxx — is the invalid backward index chain pointer.

yyyyyy — is the (hexadecimal) logical block address of the index block containing the invalid reverse pointer.

zz — is the (hexadecimal) offset in the index block, where the invalid pointer may be found. This should always be set to 0.

Meaning:

While processing an indexed file during a backup operation, Fastback has read the file's index chain and found a backward pointer that does not point to the previous index block (or is not zero in the case of the first index block). Fastback will have digested all index blocks up to the one containing the invalid backward pointer and will truncate the backup copy of the file to the number of records which can be mapped by the already-digested index blocks. Hence this message is followed by the File truncated.... message. Please refer to the description of this message.

Program Action:

Fastback will continue with the backup operation but will truncate the backup copy of the file to some incomplete number of logical records.

Required Operator Action:

OS/32 Fastchek should be run on the disc, preferably with the REPORTONLY option set initially.

```
WARNING: fd - LAST SECTOR ADDRESS xxxxxxxx IS LESS THAN
         FIRST wwwwwwww
         IN DIRECTORY BLOCK AT yyyyyy OFFSET zz
         FILE WILL NOT BE COPIED
```

Where:

fd — is the file descriptor of the contiguous file involved.

xxxxxxxx — is the pointer to the last sector of the contiguous file, as obtained from the directory entry.

wwwwwwww — is the pointer to the first sector of the contiguous file, as obtained from the directory entry.

yyyyyy — is the logical block address of the directory sector that contains the file's directory entry.

zz — is the (hexadecimal) offset into the directory sector of the file's directory entry.

Meaning:

The pointer to the last sector of the contiguous file is less than the pointer to the first sector of the file (i.e., the file's extent ends before it starts).

Program Action:

Fastback will not backup the given file but will otherwise continue the backup operation normally.

Required Operator Action:

OS/32 Fastchek should be run on the disc to check the integrity of the directory and file structure. It should initially be run with the REPORTONLY option set (so that it doesn't delete any files, etc.).

WARNING: nnn UNREADABLE DATA BLOCK(S) SKIPPED
UNREADABLE/LOST DATA WILL NOT BE COPIED/VERIFIED

Where:

nnn — is the number of unreadable data blocks that had to be skipped.

Meaning:

Fastback is reading real data blocks from a Fastback tape and has encountered one or more unreadable blocks. (When Fastback finds an unreadable block, it forward records over it and then tries to read the next block. This continues until it reads a good data block, or it finds a filemark. It then reports, via the above message, the number of blocks that it could not read.)

Program Action:

Fastback will continue with the current operation but will not be able to copy or verify the contents of the unreadable data block(s).

If the operation is a restore or a verifyonly, Fastback will be able to determine (from its link table) which sectors on the unreadable block(s) it needed. These sectors will be added to the error pool, and, before Fastback terminates, it will attempt to identify the owner(s) of the missing data sector(s).

Note that if one or more tape blocks were unreadable during the write data from tape to disc phase of a restore operation, then their contents will not be restored. If, during a verify pass of the restore operation, Fastback is able to read one or more of the (previously unreadable) blocks, it will report nonverify errors for all the sectors which do not match their counterparts on disc (which were never restored).

Required Operator Action:

None

WORKSPACE DISC devn: (VOLUME voln) NOT MARKED ON UNPROTECTED

Where:

devn:	is the device mnemonic of the disc drive containing the pack that is not marked on unprotected.
voln	is the volume name of the pack that is not marked on unprotected.

Meaning:

The disc specified by the WORKSPACE command is not marked on unprotected.

Program Action:

If Fastback was started in interactive mode, it will generate the error message and then reissue the appropriate prompt. In other modes (immediate and/or batch), Fastback will terminate with end of task code 4.

Required Operator Action:

Mark on the indicated disc unprotected, and then run Fastback again.

CHAPTER 7
INTERNAL FAILURE

7.1 FASTBACK ERROR RECOVERY

In certain conditions, Fastback may detect some error or internal inconsistency from which it cannot recover. When this happens, Fastback will pause after printing the following message on the system console (or MTM terminal):

```
INTERNAL FAILURE rrnn AT xxxxxxx  STACK PTR ssssssss
  yyyyyyyy  zzzzzzzz
  yyyyyyyy  zzzzzzzz
  ........  ........
```

Where:

rr	is the major internal failure code and identifies the area of failure.
nn	is the minor internal failure code and identifies the specific failure.
xxxxxxxx	is the address at which the failure occurred.
ssssssss	is the current value of the workspace stack pointer.
yyyyyyyy	and zzzzzzzz give the stack walkback information.

If Fastback is then continued, it will terminate with end of task code 251. The task pauses rather than terminating directly so as to enable the user to dump any task related information, for example, by using the DISPLAY REGISTERS or EXAMINE commands.

If an internal failure should occur, the user is requested to contact the nearest Perkin-Elmer support office. The following information will assist the diagnosis of the fault:

- a copy of the system console log covering the period in which the failure occurred,
- a copy of any listing output produced,

- a memory map of the system,
- a dump of the task's registers,
- a dump of the impure segment of the task (this is most simply produced by using the EXAMINE command to dump the required area; alternatively a panic dump of all memory can be made), and
- a copy of the select list, if any.

If Fastback failed during a restore, verifyonly, or tape display operation, the user is encouraged to include the tape from which the data was being restored, verified or displayed, if possible.

Internal failure conditions are not due to any operator error, and it is highly likely that rerunning the task with the same parameters will result in the same failure.

APPENDIX A
FASTBACK COMMAND SUMMARY

BACKUP

BEFORE[=][date] [,time]

BUFFERSIZE[=][{nn / 16}]

DISC[=]{devn: / voln}

DISPLAY

END

ERRORPOOL[=][{entries / 64}]

KEEPFILES[=]{TAPE / NEWER / DISC}

LIST[=][{fd / PR:}]

NOPROTECT

ONERROR[=][{SKIP | ABORT | PAUSE}]

POSITIONING[=][{nn | NOREWIND | REWIND}]

RESTORE

SELECT[=]{fd | @ | immed-entry}

SINCE[=][date][,time]

TAPE[=]{devn:}

TEMPLATE[=][short-fd]

VERIFY

VERIFYONLY

or

VO

WORKSPACE[=]{devn: | voln}

APPENDIX B
END OF TASK CODES

The end of task codes used by Fastback are given below.

END OF TASK CODE	MEANING
0	Normal completion (requested operation accomplished without error)
1	Backup operation completed; verify not possible because input disc was not marked on protected, and Fastback had insufficient memory available to build a sector allocation map (SAM) with granularity 1
2	Error in start arguments or incompatible operating system revision level
3	Error in command read from batch command file
4	Inconsistency in specified commands
5	Insufficient memory or workspace
6	Checksum error encountered on tape
7	Error(s) detected in select list
8	Invalid volume descriptor (may occur for any operation involving a disc pack)
9	Nonverify during backup, restore, or verifyonly
10	Fatal I/O error
11	Workfile related error
12	Error on list device
13	Could not assign file in order to check if it exists
14	File to be restored/verifyonlyed is currently assigned for write

END OF TASK CODE	MEANING
15	I/O error (other than end of medium (EOM)) while prefilling an (indexed) intermediate file
16	Unable to assign an intermediate file prior to renaming
17	Unable to delete file with duplicate name
18	Unable to rename an intermediate file
19	Date/time fetched from OS/32 is earlier than date/time when PACKINFO.DIR was last updated
20	Fatal I/O error on command device/file
21	Fatal I/O error on select input device/file
22	Tape data format error; the format of the data on the tape is not what Fastback expected (for example, Fastback might have found an EOF1 (end of file) label not followed by a filemark, or a filemark not followed by either an EOF1 label or an EOV1 (end of volume) label, etc.)
23	No files transferred (i.e., backed up, restored or verifyonlyed) or displayed
24	Recoverable error during operation (for example, during a restore operation Fastback might have found an unreadable tape data block which it then skipped and later reported all sectors/files whose data was not restored)
250	STOP message received or !STOP response to interactive command mode prompt
251	Internal failure
255	Cancelled by (console) operator

APPENDIX C
LINK PROCEDURE

Fastback is supplied both in task image and object form. If the user wishes to establish Fastback, then OS/32 Link must be used and the required Link commands are given below.

```
ESTABLISH TASK
DCMD
MAP
INCLUDE FASTBACK.OBJ
BUILD FASTBACK.TSK
END
```

Note that the required options are set using Link commands imbedded in the Fastback object file and are set as follows.

```
ABS=0
SYSSPACE=FFFFF
WORKSPACE=(A000,80000)
SEGMENTED
DISC
ACP
LU=9
IOBLOCKS=1
UT
NKEY
NROLL
NFLOAT
NDFLOAT
```

Link will give a warning message because the absolute space is less than 100, and on completion, will terminate with end of task code 1.

APPENDIX D
LOGICAL UNIT USAGE

The logical unit assignments used by Fastback are given in the following table.

LOGICAL UNIT	ACCESS PRIVILEGE	USAGE
0	*NONE*	Reserved (for journal output)
1	SRW	Used in restore and verifyonly operations to access the file association table (FAT)
2	SRW	Used in restore and verifyonly operations to access the link table (LT)
3	SWO	List device or file
4	SRO/ERW	Select list device or file (SRO); also used to assign files (ERW) for prefilling and renaming during restore operations.
5	SRW	Command device or file
6	SRO	Input device (disc or tape)
7	SRW	Output device (disc or tape)
8	*NONE*	Reserved (for stack-journal output)

APPENDIX E
COMPATABILITY WITH OTHER PRODUCTS

E.1 16-BIT SYSTEMS

Fastback can be used to backup, restore, verifyonly, and display disc packs created and used under OS/16. However, any pack processed by Fastback must be marked on with the NEW option when returned to the OS/16 environment.

In addition (in common with OS/32 Backup), Fastback does not support files with account numbers greater than 255. Thus, if Fastback is used to process a pack containing files allocated under OS/16 with account numbers over 255, then these files will not be processed because they will appear to have illegal file names.

E.2 OS/32 BACKUP

Fastback is totally incompatible with OS/32 Backup. The two programs operate in dramatically different manners, and each program's tape format is completely incompatible with the other.

E.3 OS/32 COPY

Fastback is totally incompatible with OS/32 Copy. OS/32 Copy cannot be used to restore tapes created by Fastback.

E.4 NON-STANDARD DISC DEVICES

Fastback can be used to process discs supported by user-written or nonstandard drivers, provided that the drivers obey the following conventions:

- The device control block (DCB) is set up using the standard DCB macros.
- The device attributes and flags are the same as those used for standard disc devices.

- The device code used is either a standard OS/32 device code or is in the range 240 (X'F0') to 254 (X'FE'). The user should exercise caution when reusing standard device codes, since these imply certain sizes and properties on the involved disc(s). This particularly applies to virtual discs that use standard device codes.

- The driver supports the standard SVC 1 function codes in the same manner as the standard disc drivers.

- The driver uses the standard device independent status codes. Note also that a status of X'8283' returned for a write function is taken to indicate that drive is hardware write protected.

APPENDIX F
COMPARISON WITH OS/32 BACKUP

F.1 INTRODUCTION

OS/32 Fastback is not intended to replace OS/32 Backup. Rather, it is an enhanced backup utility for those environments where a fast backup capability is required.

F.2 BACKUP FUNCTION

The following table gives the equivalent Fastback command for each of the Backup commands, for performing a backup operation.

BACKUP PARAMETER	FASTBACK COMMAND
IN = devn:	DISC [=] devn: DISC [=] voln:
OUT = devn:	TAPE [=] devn:
LIST = fd	LIST [=] fd
COMMAND = fd	COMMAND [=] fd
END	END
VERIFY	VERIFY
ABORT	ONERROR [=] ABORT
SKIP	ONERROR [=] SKIP
SINCE = date,time	SINCE [=] [date][,time]
SELECT = fd	SELECT [=] fd SELECT [=] @ SELECT [=] (entry)

Thus the following START commands are equivalent:

```
BACKUP:      ST ,IN=DSC1:,OUT=MAG1:,LI=PR:
FASTBACK:    ST ,BACKUP,DISC=DSC1:,TAPE=MAG1:,LIST=PR:


BACKUP:      ST ,IN=DSC1:,OUT=MAG1:,LIST=PR:,VERIFY,
                        SELECT=SYS:ASDG.BKP/0
FASTBACK:    ST ,BACKUP,DISC=DSC1:,TAPE=MAG1:,LIST=PR:,
                        VERIFY,SELECT=SYS:ASDG.BKP/0


BACKUP:      ST ,IN=DSC2:,OUT=MAG1:,VERIFY,LIST=PR:,
                        SINCE=7/AUG/81,10:17:05
FASTBACK:    ST ,BACKUP,DISC=DSC2:,TAPE=MAG1:,VERIFY,
                        LIST=PR:,SINCE=7/AUG/81,10:17:05
```

F.3 RESTORE FUNCTION

The following table gives the equivalent Fastback command for each of the Backup commands for performing a restore operation.

BACKUP PARAMETER	FASTBACK COMMAND
IN = devn:	TAPE [=] devn:
OUT = devn:	DISC [=] devn: DISC [=] voln
LIST = fd	LIST [=] fd
COMMAND = fd	COMMAND [=] fd
END	END
VERIFY	VERIFY
ABORT	ONERROR [=] ABORT
SKIP	ONERROR [=] SKIP
SINCE = date,time	SINCE [=] [date][,time]
SELECT = fd	SELECT [=] fd SELECT [=] @ SELECT [=] (entry)
DELETE	KEEPFILES [=] NEWER
DELETE/NODATECHECK	KEEPFILES [=] TAPE

Thus the following START commands are equivalent:

```
BACKUP:      ST ,IN=MAG1:,OUT=DSC1:,LI=PR:
FASTBACK:    ST ,RESTORE,TAPE=MAG1:,DISC=DSC1:,LIST=PR:


BACKUP:      ST ,IN=MAG1:,OUT=DSC1:,LIST=PR:,VERIFY,
                        SELECT=SYS:ASDG.BKP/0
FASTBACK:    ST ,RESTORE,DISC=DSC1:,TAPE=MAG1:,LIST=PR:,
                        VERIFY,SELECT=SYS:ASDG.BKP/0


BACKUP:      ST ,IN=DSC2:,OUT=MAG1:,VERIFY,LIST=PR:,
                      SINCE=7/AUG/81,10:17:05,
                        DELETE/NODATECHECK
FASTBACK:    ST ,BACKUP,DISC=DSC2:,TAPE=MAG1:,VERIFY,
                        LIST=PR:,SINCE=7/AUG/81,10:17:05,
                        KEEPFILES=TAPE
```

F.4 VERIFYONLY FUNCTION

The following table gives the equivalent Fastback command for each of the Backup commands for performing a verifyonly operation.

Note that for verifyonly operations, Fastback will always treat the tape as the input device, although it does not specifically correspond to the Backup input device as is implied by the following table:

BACKUP PARAMETER	FASTBACK COMMAND
IN = devn:	TAPE [=] devn:
OUT = devn:	DISC [=] devn: DISC [=] voln
LIST = fd	LIST [=] fd
COMMAND = fd	COMMAND [=] fd
END	END
VERIFYONLY	VERIFYONLY
ABORT	ONERROR [=] ABORT
SKIP	ONERROR [=] SKIP
SINCE = date,time	SINCE [=] [date][,time]
SELECT = fd	SELECT [=] fd SELECT [=] @ SELECT [=] (entry)

Thus the following START commands are equivalent:

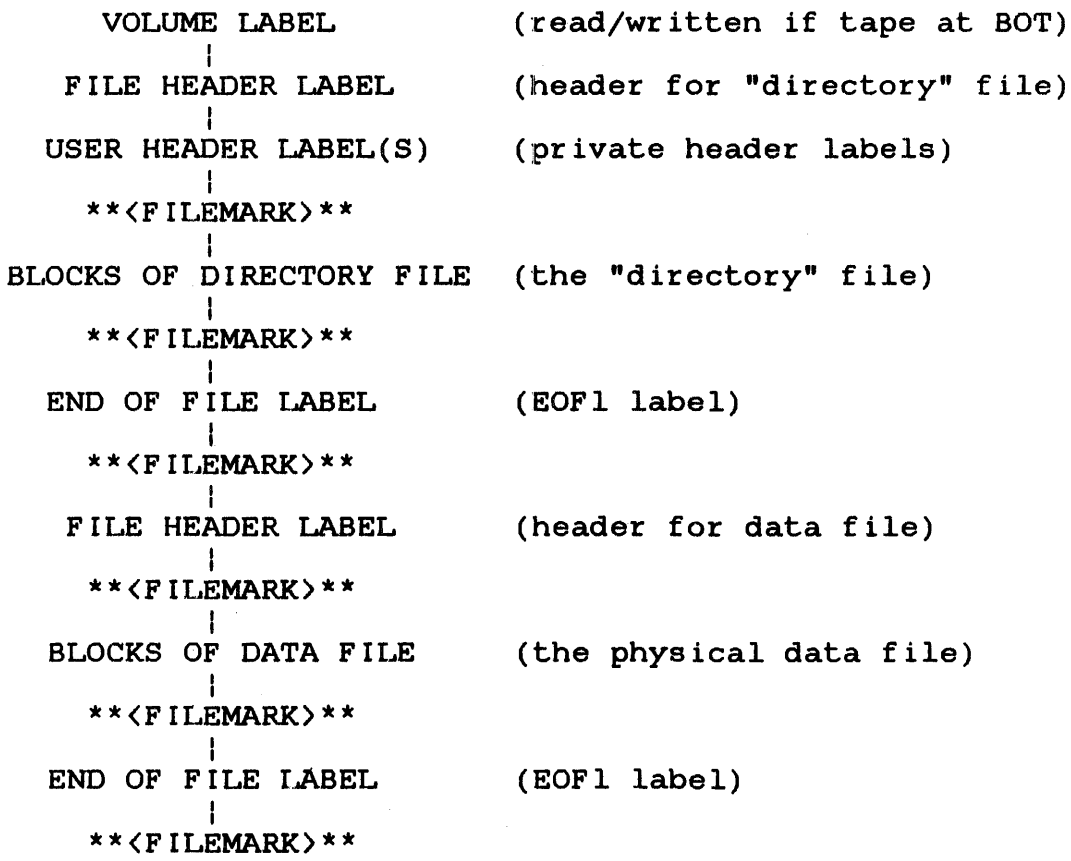
```
BACKUP:      ST ,VO,IN=MAG1:,OUT=DSC1:,LI=PR:
FASTBACK:    ST ,VERIFYONLY,TAPE=MAG1:,DISC=DSC1:,LIST=PR:

BACKUP:      ST ,IN=MAG1:,OUT=DSC1:,LIST=PR:,VO,
                        SELECT=SYS:ASDG.BKP/0
FASTBACK:    ST ,VO,DISC=DSC1:,TAPE=MAG1:,LIST=PR:,
                        SELECT=SYS:ASDG.BKP/0

BACKUP:      ST ,IN=DSC2:,OUT=MAG1:,VO,LIST=PR:,
                        SINCE=7/AUG/81,10:17:05,
FASTBACK:    ST ,VERIFYONLY,DISC=DSC2:,TAPE=MAG1:,
                        LIST=PR:,SINCE=7/AUG/81,10:17:05,
```

F.5 DISPLAY FUNCTION

OS/32 Backup does not have a function corresponding to Fastback's display function.

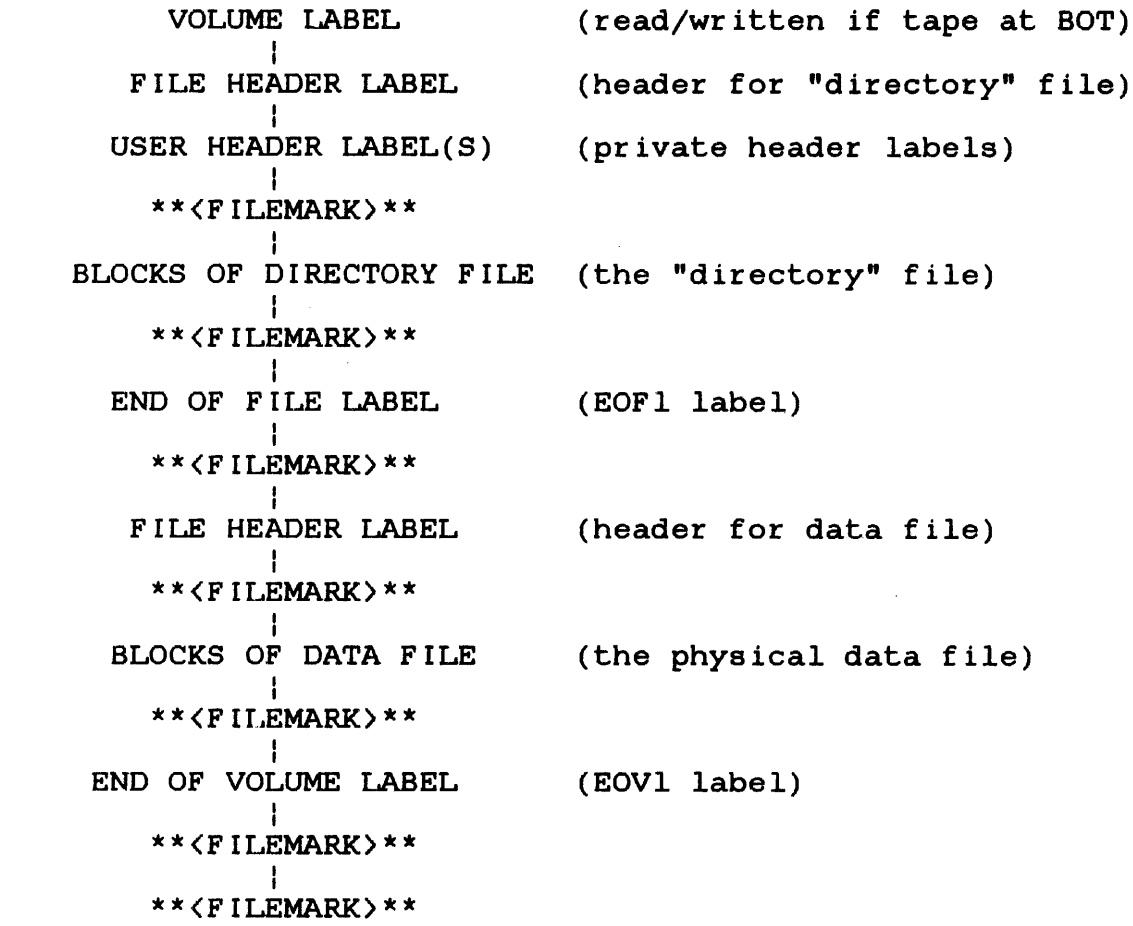
APPENDIX G
FASTBACK TAPE FORMAT

OS/32 Fastback builds tapes using ANSI tape labelling standards. A single reel backup (i.e., multi-file volume) has the format:

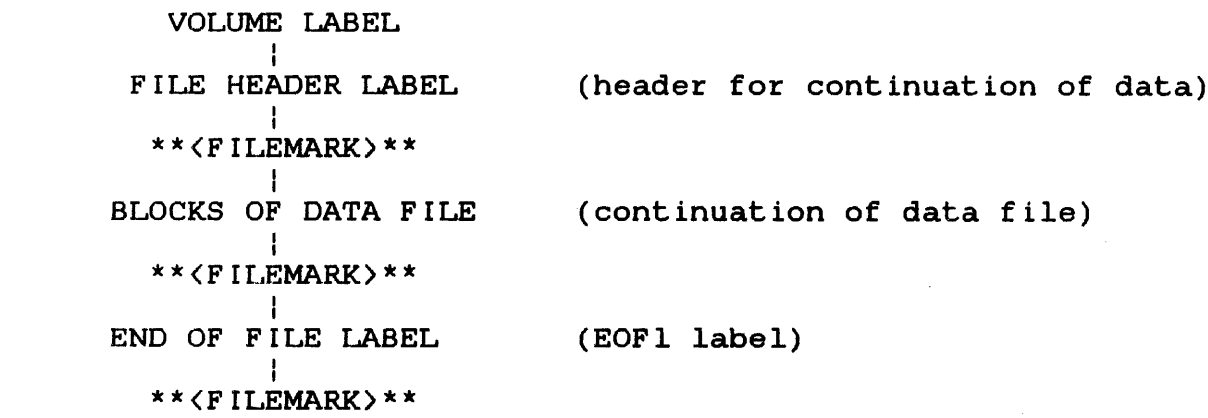


A backup requiring more than one reel of tape (i.e., a multi-file, multi-volume backup) has the following format.

TAPE 1:



TAPE n:



Note that the data file may be continued across as many reels of tape as is necessary.

Also, if the end of a tape should occur in the "directory" file, then the tape will be continued as illustrated above except that the directory file, and not the data file, will be continued.

Users who wish to examine the contents of Fastback's volume, header, end of file, and end of volume labels should dump the appropriate labels from a Fastback tape with the use of OS/32 Copy.

Finally, note that for the current revision of Fastback, most fields within the volume and header labels (etc.) are set to predefined values. For example, the tape serial number is always zero, and the tape expiration date is always 31st December 1999.

It is envisaged that future revisions of Fastback may allow users to specify the contents of some of these fields.

INDEX

PUBLICATION COMMENT FORM

We try to make our publications easy to understand and free of errors. Our users are an integral source of information for improving future revisions. Please use this postage paid form to send us comments, corrections, suggestions, ect.

1. Publication number_____

2. Title of publication_____

3. Describe, providing page numbers, any technical errors you found. Attach additional sheet if neccessary.

_____

_____

_____

4. Was the publication easy to understand? If not, why?

_____

5. Were illustrations adequate? _____

_____

_____

6. What additions or deletions would you suggest? _____

_____

_____

7. Other comments: _____

_____

_____

From _____ Date _____

Position/Title _____

Company _____

Address _____

_____

_____

STAPLE STAPLE

FOLD FOLD

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 22 OCEANPORT, N.J.

POSTAGE WILL BE PAID BY ADDRESSEE

PERKIN-ELMER

Data Systems Group
106 Apple Street
Tinton Falls, NJ 07724

ATTN:
TECHNICAL SYSTEMS PUBLICATIONS DEPT.

FOLD FOLD

STAPLE STAPLE