

Published: 07/25/68

Identification

A user's manual for the (interim) tape system process  
D. L. Stone

Purpose

The tape system process provides the Multics Operator with the ability to read 7punch format tapes and to allow 7punch and IMCV format tapes to be written. The tape process is invoked as a standard command and monopolizes the teletype channel to which the calling process is attached.

Usage

Any process may become the tape system process by calling: "tape\_daemon (input, output)" where "input" and "output" are one-character varying string ascii encodings of the drive numbers to be used by the tape process.

From command level, the invocation is

"tape\_daemon input output".

When the tape\_daemon segment is entered, it initializes its communications with the two logical tape units specified and sets up a channel of communication for other processes to notify it of requests. A loop is then entered, from which all the tape\_daemon's powers may be invoked. The typewriter connected to the process which called tape\_daemon will type the message:

"daemon calling wait. any requests?"

The valid responses are:

1. "quit" - the tape\_daemon cleans up and returns to command level.
2. "ret" - the input drive is checked for readiness. If it is not ready, the message:

"input tape not ready"

is typed and the main loop is reentered. If it is ready, the segment "td\_input" is invoked to read in the tape and put the files on it in their specified directories. In so doing, td\_input will type both the input card image for each file and the segment and directory names used to file the segment

(but see "brief"). Successful processing is indicated by "input completed"

"last line is: \_\_\_\_\_" which should be "stop"

If any unrecoverable errors are encountered, the main loop is re-entered after unloading the tape. See the error list in the appendix for more details.

3. "file" The message "type file name" is typed, and a character string of size  $\leq 50$  is read. This string is used as the name of a segment to be treated as a tape\_daemon control file. Either path names or entry names may be specified; the tape\_daemon actions are controlled by the file, if successfully initiated. (See BE.17.02 for more information).
4. "brief" A binary flag is complemented. The initial value is "verbose".
5. Any response line not beginning with one of the above tokens will cause the tape\_daemon to call the wait coordinator. From there, the tape\_daemon can be awakened either by special interrupts sent from a tape handler or by inter-process communication initiated by some process desirous of having a control file executed. In the former case, the response is as if "ret" had been typed; in the latter, as if "file". In the "file" case, the file name used in the "unique\_chars" encoding of the first seventy bits of the event id; a link by this name should have been made in the tape\_daemon directory by the signalling process.

#### Tape Mounting Messages

- "mount tape reelno, read" - mount specified tape on input drive.
- "mount tape reelno, write" - ditto on output drive. If no reelno is specified, mount a scratch tape.
- "type when ready" a mount request has been followed by a check for readiness. When the tape is ready, type a carriage return. This message will be repeated until the tape is ready.
- "Please unmount tape read" -
- "Please unmount tape write" -

## Errors

1. Possible error messages from processing by the tape\_daemon are:

"attachment error" the tape\_daemon was unable to attach to the drives specified. Returns to command level.

"ipc error"  
"error code is XXXXXXXXXXXX" unable to establish interprocess communication channels. Enters main loop. Error is given as 12 character octal.

"Smm error initiating control file XXXXX"  
"file is : name :" unable to initiate a control file. Main loop is entered.

"retrying file" due to a tape error, the tape\_daemon resignals the ipc event which awakened it, then enters main loop.

2. Errors reported from other segments are:

- a. file processing

"fatal scan error" badly constructed control file. If a tape has been mounted, it is unloaded and return is made to tape\_daemon.

"funny control line" a control line in a file was not recognized, it is ignored.

"tape error XXXX" a fatal tape error has been encountered.  
Error codes are: 1 = eof  
2 = tape error  
3 = format error on tape

"premature eof on file" control file ended too soon. A "stop" card is assumed, and processing continues.

"no file with name name" an unsuccessful attempt was made to initiate file name. The tape most recently mounted by this file is unloaded and return is made to tape\_daemon.

"no status" tape controller or gioc error.  
run dies with divide check.

b. tape reading

"delentry error XXXX" error return from "delentry"  
in decimal. tape is unloaded and  
return is made.

"setbc error XXXX"

"directory : dir :

"processing will continue" error return from "setbc"  
in decimal.

"bad tape"

"error XXXX" error returned from tape reader.  
(see tape error from file  
processing)

"scanning error" badly constructed control file.  
unload and return.

"lost status - read" tape controller or gioc error -  
an attempt to recover is made.