

## NMR NOTES #1

### NMR Data Storage on 1/4 inch and 8mm Tape (UNIX Systems)

Data can be stored on either 1/4 inch or 8mm tape cartridges, using the tar command. This command is the standard UNIX utility for archiving data. The bar command that existed under SunOs 4.1.x is no longer a part of the operating system and is no longer available. Any tapes that you have that were written with the bar format may be read and recovered, but the operation is not trivial.

The 1/4 inch tape cartridge used by the tape drive on the Sparc20 and Sparc10 workstations is a DC 6150. This is a 620 foot tape with a nominal capacity of 150 Mbytes. Both systems also have an 8mm tape drive in addition to the 1/4 inch drive. 8 mm tapes can store up to 5 GBytes per tape with the 115 meter tape cartridge. Both of these tapes are available from most manufacturers of magnetic storage media. Included on the tape cartridge is a write-protect device. Make sure that the arrow points away from the word 'safe' in order to write on the 1/4 inch cartridge, or that the slide on the back of the 8 mm cartridge points to record. It is probably a good idea to move this back to the 'safe' or 'save' position when you are done writing, just to avoid accidentally overwriting important data.

The tape drive used is specified with a device file name. On all systems, `/dev/rmt/1m` refers to the 1/4 inch tape drive. 'rmt' mean 'raw magnetic tape' and '1m' is unit #1 with medium density. The 8 mm tape is accessed as either `/dev/rmt/0l` (low density, 2.3 GB operation) or `/dev/rmt/0m` (medium density, 5.0 GB operation). Normally, the tape drive rewinds the tape after a complete operation. Simply appending an 'n' on the end of the device name specifies the device as a no-rewind drive: `/dev/rmt/0mn` specifies 'no rewind on raw streaming tape # 0, medium record density'. This permits multiple files to be placed on one tape cartridge.

When tar writes an archive to the tape, it stores all the files, along with subdirectory information, in one large logical structure (archive file) composed of fixed size tape blocks. Although the tape is a block oriented device, you don't need to worry about blocks at all - just consider the archive file as one file on the tape, composed of any number of individual files and subdirectories from the disk. One of the limitations of these tape drives is that they cannot backup or search backwards. One consequence of this is that you cannot add data to an existing tape archive file. Once it is written, it can only be read back or overwritten, but not appended, updated or modified in any way.

It is possible however to put several archive files on one tape. All that is required is that you keep track of how many different archive files you have in order to avoid overwriting data that you want. When you use tar to write to the tape, the system puts an **EOF** (End Of File) mark at the end of the file, and this normally marks the end of the valid data. It is possible to use the **mt** command to skip one or more of these **EOF** marks and to then write additional data to the tape. The **mt** command in fact has several useful options for manipulating a tape cartridge. For these operations it is easiest if you define a TAPE environment variable, using the appropriate number for the drive/density you wish to use:

**setenv TAPE /dev/rmt/0mn** device 0mn is the raw streaming tape unit #0 with no automatic rewind after a tape operation. This is what you will normally want to use.

**setenv TAPE /dev/rmt/0m** device 0m is the same thing, except that it will automatically rewind the tape cartridge after every operation.

**mt retension** This command retension the tape to avoid lost data from tape being stretched. This should be done with new tapes and any tape that has not been used for 'awhile'.

**mt rewind** Rewind the tape to the beginning.

**mt offline** Rewind the tape if necessary and take unit off line. This should be done before you remove your tape from the drive.

**mt fsf #** File Skip Forward # block(s). This skips to just past an EOF mark, so that you are ready to read or write another data block after the previous one.

**mt status** Reports the status of the tape, including the file and block numbers. File will indicate the number of EOF marks skipped, block is an offset into the current file.

**mt asf #** Skip to an absolute position on the tape. Effectively combines the functions of the rewind and fsf commands.

**mt eom** Skip to 'End Of Media' position on the tape. This moves past the last EOF mark on the tape so that you can append a new tar file on the tape. You do not have to know how many tar files currently exist on the tape. After the 'eom' command, 'status' will show the number of files present.

Normal operation would be to use tar to create an archive on the tape after retensioning and rewinding it. This will also write an EOF mark and you can use the -t option to get a directory of the tar file.

Later, when you want to add data to the tape, use either '**mt fsf 1**' or '**mt eom**' to skip over the existing data and position the tape just past the EOF mark. '**mt status**' should show file=1 and block=0. You can then use tar to create an new archive file here with the files to be added to the tape. '**mt offline**' will rewind the tape and take the unit offline.

Still later, **mt fsf 2** (or '**mt eom**') will skip over both of these file so that you can write a third data set, the same way you did the second.

When you want to access nmr data from the second data set, for instance, you would use **mt fsf 1** to skip the first file, and **tar -x <filename>** to extract the desired data. **mt status** would then show file=1 and block=xxxx, where xxxx is some offset into the file. **mt fsf 1** would then skip forward over 1 EOF mark, leaving the tape positioned at the third data set ready to read data. **mt status** would show file=2 block=0. Note that the **mt fsf #** command skips forward over the specified number of EOF marks, not to an absolute tape archive file position. Jumping to an absolute tape file number would require doing a tape rewind first. You can however use **mt asf #** to do an Absolute Skip Forward. The argument supplied is the file number of interest, with the first being 0. This is logically equivalent to doing **mt rewind** followed by **mt fsf #** except that the rewind is omitted if the desired location is forward of the current location. Effectively, the system uses mt status to find the current location and will skip forward (#-current) EOF marks.

Please note that you cannot skip to the beginning of the second block, overwrite the data there with new data, and keep the third data block. The tape is inherently serial access rather than random access, and you cannot modify information in the middle of the tape. You could however overwrite the last data block on the tape.

In order to make it a little easier to keep track of the data stored on a tar tape cartridge, there is a tape directory script defined in /usr/local/bin as **tape\_dir**. This command will read as many tar archives as it can find on your tape, list the contents of those files, and report a summary of the approximate amount of data stored on the tape. This output can be redirected to a disk file for latter reference or printing. Simply load your tape in the drive and use **tape\_dir > mytape.dir** to generate a disk file name mytape.dir with all the directory information present. If you want to print a hard copy, simply do **lp -dLaserJet4a prn -o nobanner mytape.dir**. Please note that this printer definition is the LaserJet on NMRSUN1. The printer on NMRSUN2 is LaserJet4\_prn. The -o nobanner option suppresses printing of the extra header page.