Backups

It is a good idea to backup your disks.

The larger the system the more formal the backup process should be.

The more dynamic the disk the more frequent it should be backed up.

`tar` – build a tape archive.
The archive can actually be built on any media.
The archive can be built in a file.

(Software releases are often sent as compressed `tar` files.)

`tar` *options list*

The files/directories in the list are placed into the archive or extracted from the archive.

If a directory is in the list, the subtree under that directory is archived.

Links are archived, but not followed.

Archives may span media (several CDs or tapes).

tar Details

Some common options:
`c`: create an archive
`x`: extract from an archive
`t`: list the contents of an archive
`f`: specify archive file/device
`L`: media size in 1K blocks
`z`: compress using gzip
`j`: compress using bzip2 (large files)
`J`: compress using xz

`p`: extract all protection information
`C`: change directory

Examples:

```
tar -cf /dev/nrst0 /u1
```
back up the `/u1` directory to the SCSI tape

```
tar -xf ftp/gcc.tar
```
Extract everything from the archive file `gcc.tar`
The leading `/` is removed by tar so you are able to extract into any local directory.

```
tar -xf /dev/nrst0 u1/john/.cshrc
```
Extract one file from the tape archive

`tar -tzf ftp/gcc.tgz`
list the archive contents (archive is compressed)

`tar -cfL /dev/nrst0 1000000 /u1`
back up to one or more 1GB tapes

`tar -cfL jaguar:/dev/nrst0 1000000 /u1`
the tape drive is on another machine (`jaguar`).

`tar -xpf ftp/gcc.tar`
Preserve the protection and ownership information.
You must be root to effectively use this.

`tar -xjf /usr/local/src/kernel.bz2`
Extract a file using bzip2 decompression
Convention: `.bz2` is a bzip'd tar.

`tar -xf bob.tar -C /home/bob`
Normally tar extracts into the current directory, here we tell it to extract into bob's home directory.

`tar -cJf etcbu.txz /etc`
This creates a backup usings xz compression.

Compression

Several compression programs exist and can be run independent of tar.

`zip` Compatible with windows.

`gzip` Unix version of that compresion.

`bzip2` Compresses a block at at time. Does better than zip/gzip on large files.

`xz` The most recent (advanced) compression routine. Compresses by default. Removes old file by default.
`xz junk`
compresses `junk`, creates `junk.xz`, removes `junk`
`xz -d junk.xz`
decompresses `junk.xz` into `junk`, removes `junk.xz`
`xz -z <file1 >file2`
compresses stdin (`file1`), outputs to stdout (`file2`)