

Device Management

Remember: Everything in UNIX is a file.

Hardware connected to your computer has a corresponding file in */dev*

• /dev/hda IDE drive

/dev/sda SCSI drive

/dev/fd0 floppy drive

/dev/console monitor/serial

connection

• /dev/ttyp0 pseudo terminal

• /dev/ttyS0 serial port (COM

port)



Device Management

Devices are associated with *major* and *minor* device numbers

Major number specifies the type of device so the kernel knows which driver to use

Minor number usually specifies which device

In the "olden days", each new device added to the system meant creating a new *device node* in */dev*

This is now done for you



Device Management

Block Devices

Hard drives, floppy drives and CDROM drives are *block* drives which buffer their I/O. These are the devices you will deal with most often.

Two main controller types for hard drives and CDROM drives:

- IDE
- SCSI

The names for the device files in */dev* will differ depending on the type of controller



Device Management

Block Devices

IDE drives have names beginning with "hd".

SCSI drives have names beginning with "sd".

Each drive has a letter corresponding to its location on the controller:

hda – primary IDE master drive

hdb – primary IDE slave drive

hdc – secondary IDE master drive

sda – first SCSI drive

sdb - second SCSI drive



Device Management

Partitions

Drives can be divided into *partitions* or slices of the disk which can be used for different purposes.

Partitions make your data safer. If you lose one partition, you only lose the data on that partition.

Partitions on drives are signified by a number

/dev/hdb1 – first partition on IDE drive

/dev/sda5 - fifth partition on SCSI drive



Device Management

Partitions

Partitions can be of two types – *primary* or *extended*.

Primary partitions are partitions which may be made bootable.

There may only be four primary partitions. This is an historic artifact imposed by the Master Boot Record

Extended partitions are partitions which are "contained" inside one of the four allowed primary partitions.



Device Management

Partitioning

Partitions are created using the fdisk command

fdisk /dev/hda

Most important **fdisk** commands:

- m prints help
- p prints the current partition table
- n create a new partition
- w write the partition table
- q quit without saving changes

None of your work will be remembered unless you write the table



Device Management

Partitioning

Exercise

Use the fdisk command to view the partitions on your disk drive

fdisk /dev/hda

•

•

•

Command (m for help): p



Device Management

File Systems

Partitions do you no good if you cannot access them

File systems allow you to store and access the data of a file via inodes

You must create the file system on the partition

Some common types of file systems:

- swap
- ext2 older but still used
- ext3 now standard, provides journaling
- iso9660 CDROM



Device Management

File Systems

File systems are created with **mkfs.ext2**, **mkfs.ext3** (or **mkfs** -t *type*)

mkfs.ext2 /dev/<drive><partition>

Example:

mkfs.ext2 /dev/sdb3



Device Management

Repairing File Systems

When a system reboots, the file systems are checked before they are mounted using the command **fsck**

Usually the file systems will be clean

After a crash or pressing the reset button, you may see this

/dev/sdc1: UNEXPECTED
INCONSISTENCY; RUN fsck
MANUALLY.^M

In this case, you will need to run **fsck** yourself:

fsck /dev/sdc1



Device Management

Mount Points

Partitions still do no good if the system does not know about them.

Mount points are directories which the system knows correspond to partitions

File systems are mounted and unmounted using mount and umount

Most common syntax:

mount t <*type*> <*device*> <*mount point*>

Example:

mount t ext3 /dev/hda2 /usr

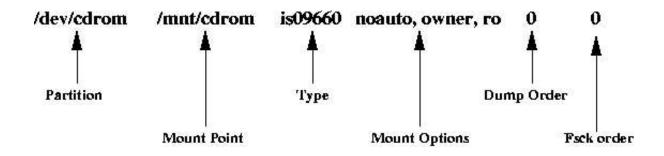


Device Management

/etc/fstab

On boot, the system checks the file /etc/fstab for all file systems to be mounted.

A typical entry in /etc/fstab looks as follows:





Device Management

/etc/fstab

/etc/fstab makes life easier

Entering **mount -a** mounts all file systems in /etc/fstab

Enter **mount** < mount point > mounts the file system associated with that mount point in /etc/fstab

Example:

mount /mnt/cdrom



Device Management

Mounted File Systems

You can see the file systems which are currently mounted using

- the **df** command
- the **mount** command with no arguments
- viewing the contents of /proc/mounts and
- viewing the contents of /etc/mtab

df is more useful as it not only lists the mounted file systems, it gives you space usage information as well



Device Management

Mounted File Systems

Exercises

Determine the partitions mounted on your system:

```
# df
# mount
# cat /proc/mounts
# cat /etc/mtab
```