CSI3660 – System Administration

Prof. Fredericks

Mounting and Compressing

Permissions Revisited

- Quick example

Additions to Adding Users

- May be beneficial to create basic directories required for all new users
  - i.e., a *skeleton*

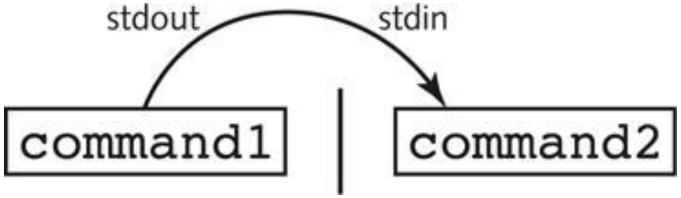
- Add whatever you wish to **/etc/skel** and it will appear for all future users!

Outline

- Piping and redirects
- Filesystems and mounting
- File compression

Piping and Redirects

- Pipes
  - Use output of one program as input to another
  - | operator
  - E.g.,
    - ls –la /etc | less
    - ls –l | sed “s/[aeio]/u/g”



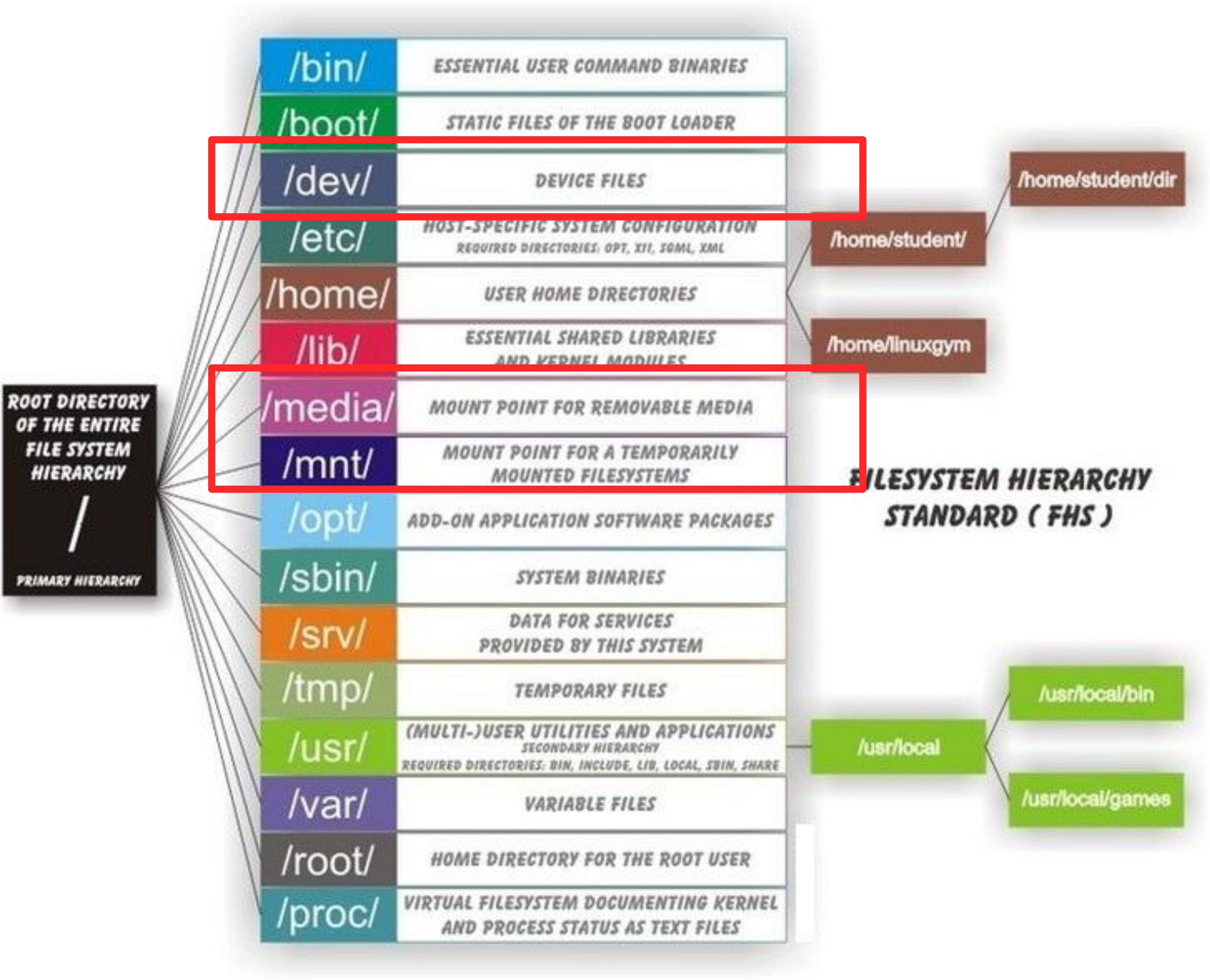
- Simple redirection
  - Output to file
  - > operator
  - E.g.,
    - locate find > locate_results.txt
    - More detail when we get into scripting

Filesystem and Mounting

- /dev → Structure and types of device files
- Common filesystem types and features
- Mount and unmount filesystems
- Manage media (CD, DVD, USB, Firewire, HDD partitions)
-

Chapter 8

Filesystem and Mounting



ROOT DIRECTORY OF THE ENTIRE FILE SYSTEM HIERARCHY

/

PRIMARY HIERARCHY

/bin/	ESSENTIAL USER COMMAND BINARIES
/boot/	STATIC FILES OF THE BOOT LOADER
/dev/	DEVICE FILES
/etc/	HOST-SPECIFIC SYSTEM CONFIGURATION REQUIRED DIRECTORIES: OPT, X11, SGML, XML
/home/	USER HOME DIRECTORIES
/lib/	ESSENTIAL SHARED LIBRARIES AND KERNEL MODULES
/media/	MOUNT POINT FOR REMOVABLE MEDIA
/mnt/	MOUNT POINT FOR A TEMPORARILY MOUNTED FILESYSTEMS
/opt/	ADD-ON APPLICATION SOFTWARE PACKAGES
/sbin/	SYSTEM BINARIES
/srv/	DATA FOR SERVICES PROVIDED BY THIS SYSTEM
/tmp/	TEMPORARY FILES
/usr/	(MULTI-)USER UTILITIES AND APPLICATIONS SECONDARY HIERARCHY REQUIRED DIRECTORIES: BIN, INCLUDE, LIB, LOCAL, SBIN, SHARE
/var/	VARIABLE FILES
/root/	HOME DIRECTORY FOR THE ROOT USER
/proc/	VIRTUAL FILESYSTEM DOCUMENTING KERNEL AND PROCESS STATUS AS TEXT FILES

FILESYSTEM HIERARCHY STANDARD (FHS)

Hard Drives and SSDs

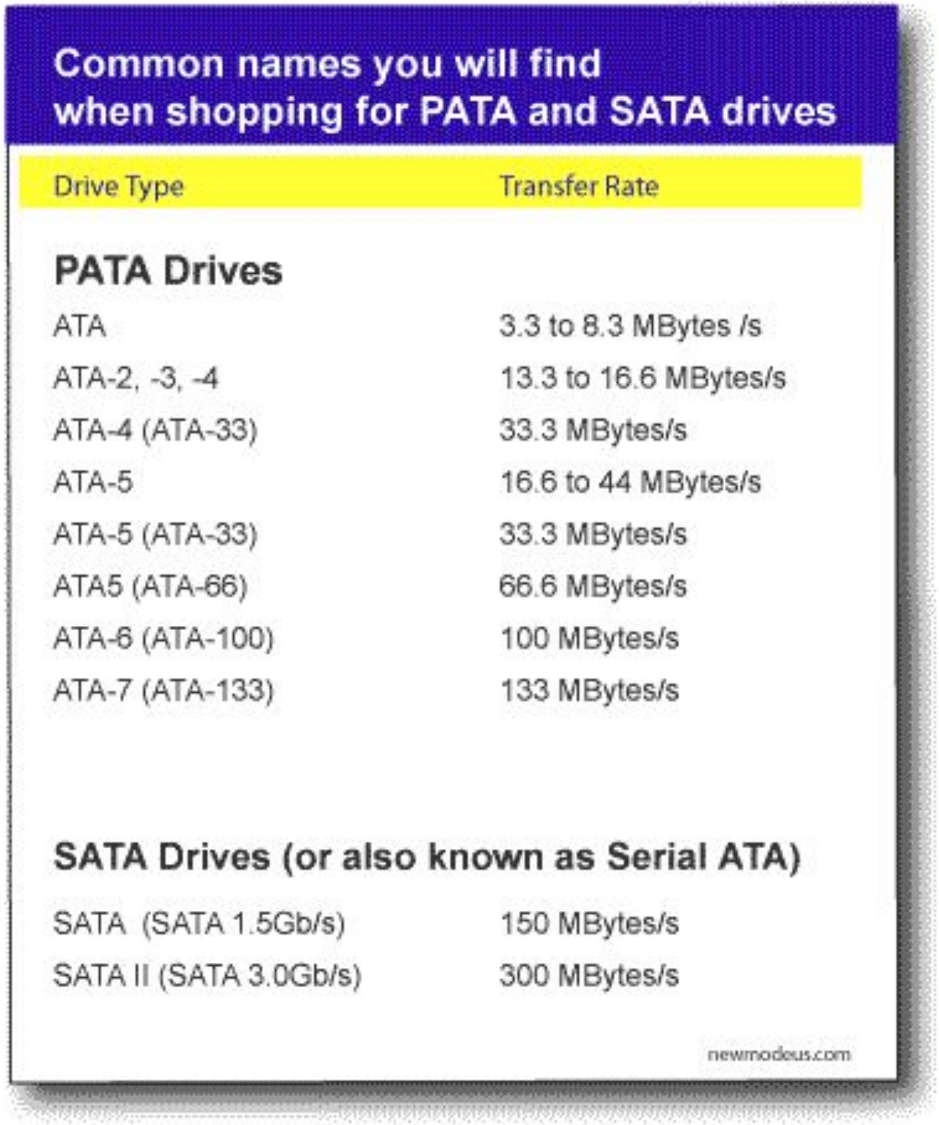
- PATA (parallel ATA) → legacy
  - Master/slave relationship
  - Primary (hda, hdb), Secondary (hdc, hdd)
  - Each has different device file
- SCSI / SATA (serial ATA) → more current
  - Can include solid-state drives
  - First HDD (sda)
  - Second (sdb)
  - …
  - sda could refer to first SATA or first SATA SSD

PATA and SATA cables

http://www.computerhope.com

Hard D

- PATA (pa
  - Master/s
  - Primary (
  - Each ha
- SCSI / SAT
  - Can incl
  - First HDD
  - Second
  - …
  - sda coul



Common names you will find when shopping for PATA and SATA drives

Drive Type	Transfer Rate
PATA Drives	
ATA	3.3 to 8.3 MBytes /s
ATA-2, -3, -4	13.3 to 16.6 MBytes/s
ATA-4 (ATA-33)	33.3 MBytes/s
ATA-5	16.6 to 44 MBytes/s
ATA-5 (ATA-33)	33.3 MBytes/s
ATA5 (ATA-66)	66.6 MBytes/s
ATA-6 (ATA-100)	100 MBytes/s
ATA-7 (ATA-133)	133 MBytes/s
SATA Drives (or also known as Serial ATA)	
SATA (SATA 1.5Gb/s)	150 MBytes/s
SATA II (SATA 3.0Gb/s)	300 MBytes/s

newmodeus.com

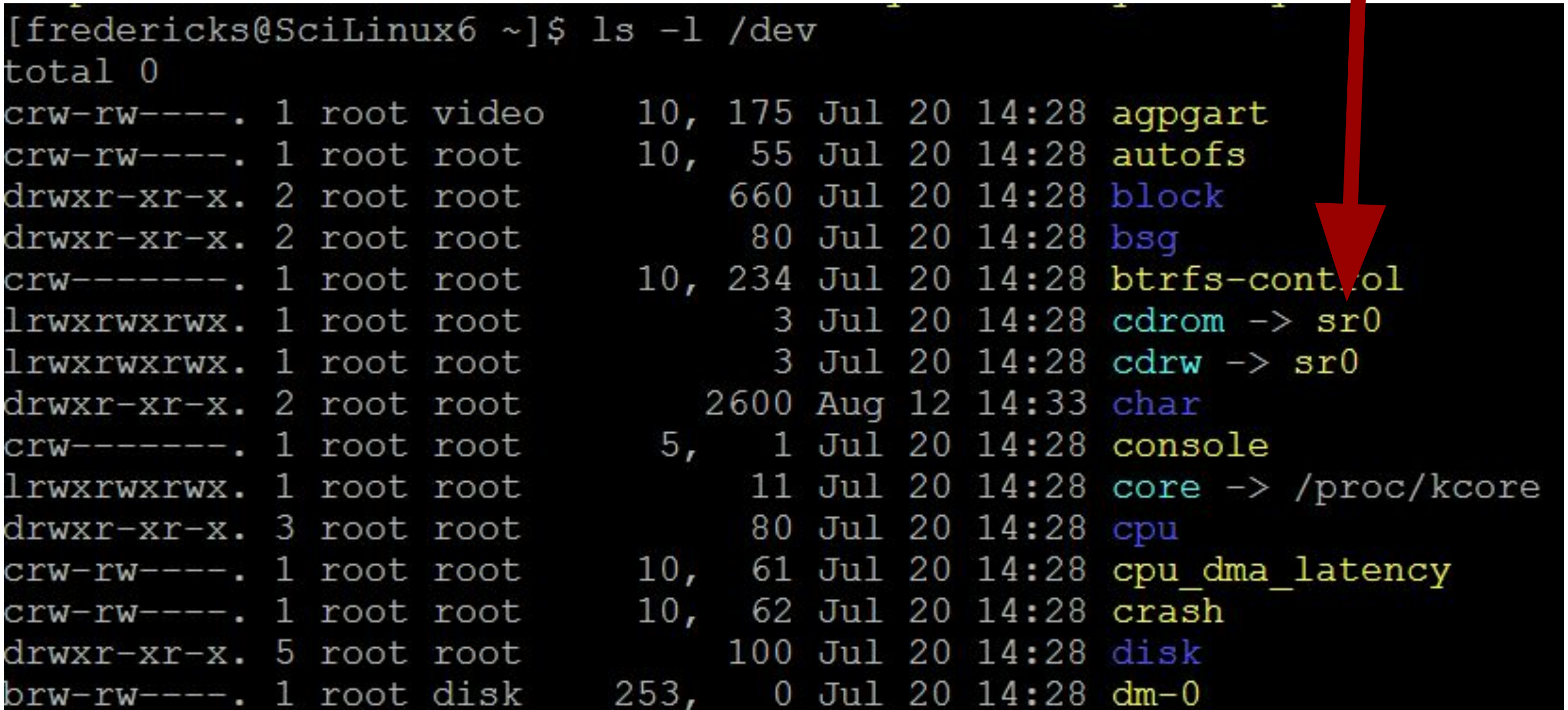
PATA and SATA cables

http://www.computerhope.com

Back to /dev

- Contains device files
  - Represent system device
  - One file per device
  - How to transfer data to/from device

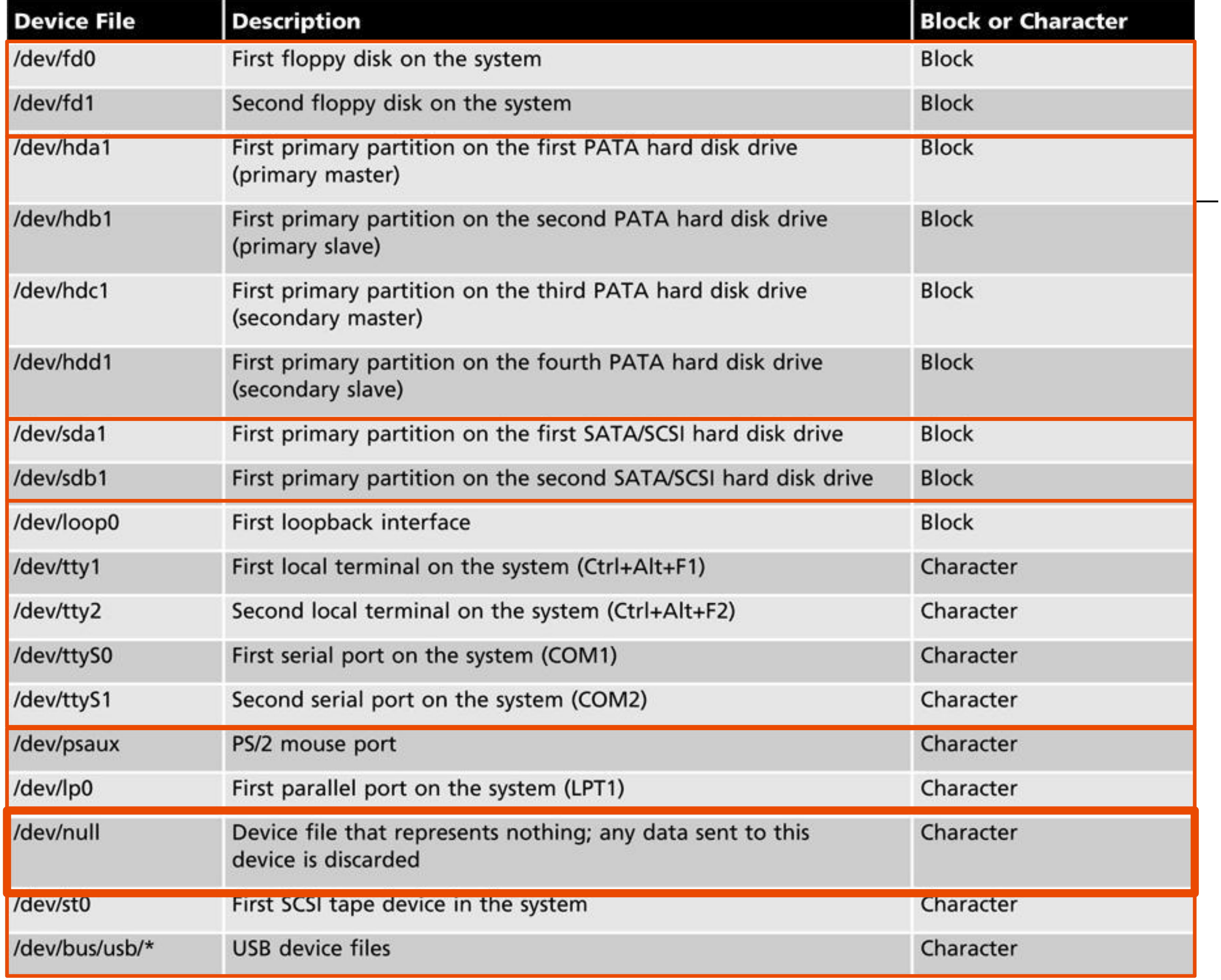
Kernel device controller



```
[fredericks@SciLinux6 ~]$ ls -l /dev
total 0
crw-rw----. 1 root video    10, 175 Jul 20 14:28 agpgart
crw-rw----. 1 root root     10,  55 Jul 20 14:28 autofs
drwxr-xr-x. 2 root root         660 Jul 20 14:28 block
drwxr-xr-x. 2 root root          80 Jul 20 14:28 bsg
crw-------. 1 root root     10, 234 Jul 20 14:28 btrfs-control
lrwxrwxrwx. 1 root root           3 Jul 20 14:28 cdrom -> sr0
lrwxrwxrwx. 1 root root           3 Jul 20 14:28 cdrw -> sr0
drwxr-xr-x. 2 root root        2600 Aug 12 14:33 char
crw-------. 1 root root      5,   1 Jul 20 14:28 console
lrwxrwxrwx. 1 root root          11 Jul 20 14:28 core -> /proc/kcore
drwxr-xr-x. 3 root root          80 Jul 20 14:28 cpu
crw-rw----. 1 root root     10,  61 Jul 20 14:28 cpu_dma_latency
crw-rw----. 1 root root     10,  62 Jul 20 14:28 crash
drwxr-xr-x. 5 root root         100 Jul 20 14:28 disk
brw-rw----. 1 root disk    253,   0 Jul 20 14:28 dm-0
```

Back to /dev

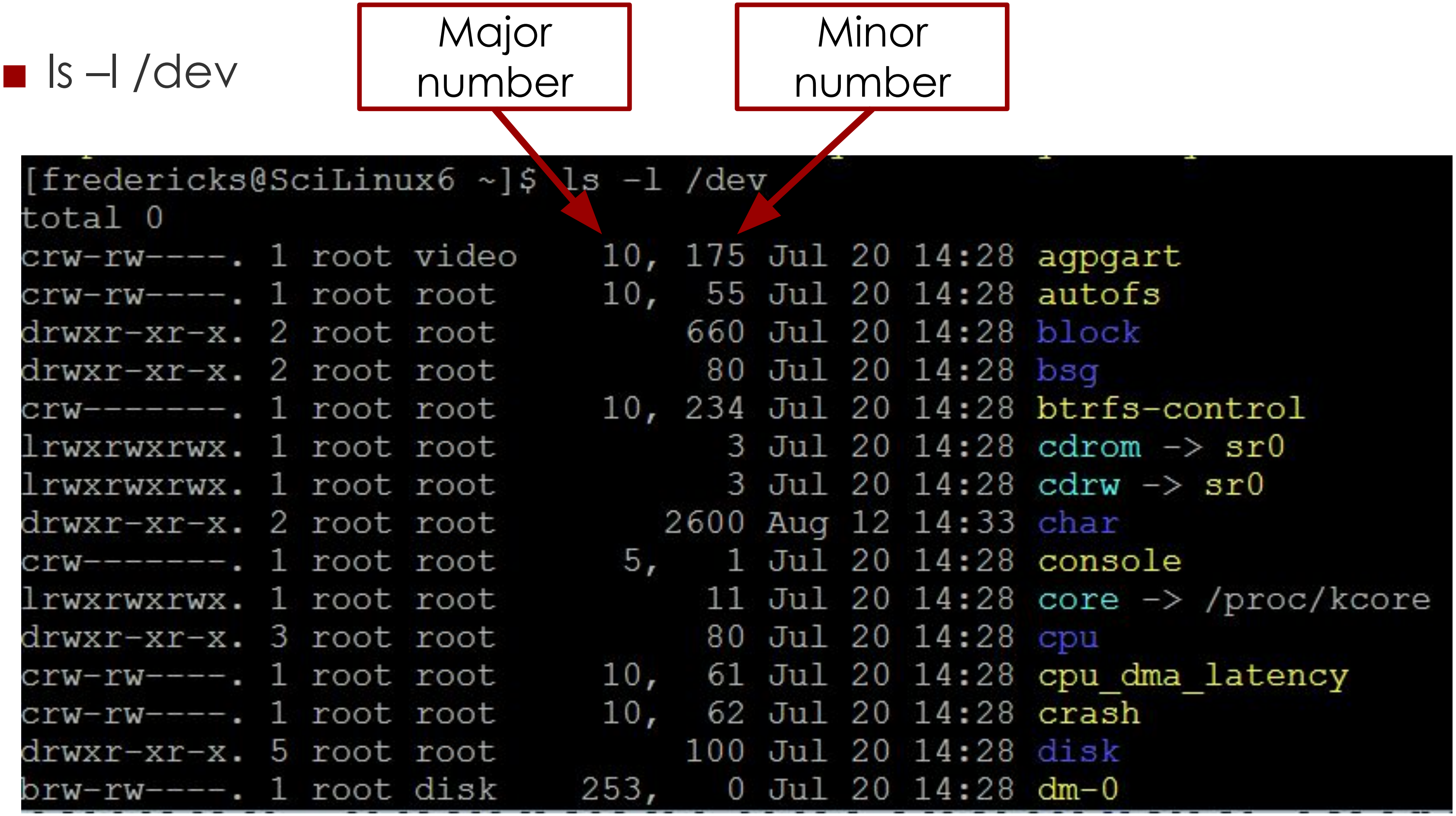
- Three types of devices
  - Character device
    - Transfer data character-by-character
    - Serial/parallel ports, sound cards, terminal, etc.
  - Block device
    - Transfer data in blocks
    - Faster
    - Floppy disks, CD-ROMS, USB, file loopback, etc.
  - Network device
    - Transfer data in packets
    - Network cards / network loopback devices



Device File	Description	Block or Character
/dev/fd0	First floppy disk on the system	Block
/dev/fd1	Second floppy disk on the system	Block
/dev/hda1	First primary partition on the first PATA hard disk drive (primary master)	Block
/dev/hdb1	First primary partition on the second PATA hard disk drive (primary slave)	Block
/dev/hdc1	First primary partition on the third PATA hard disk drive (secondary master)	Block
/dev/hdd1	First primary partition on the fourth PATA hard disk drive (secondary slave)	Block
/dev/sda1	First primary partition on the first SATA/SCSI hard disk drive	Block
/dev/sdb1	First primary partition on the second SATA/SCSI hard disk drive	Block
/dev/loop0	First loopback interface	Block
/dev/tty1	First local terminal on the system (Ctrl+Alt+F1)	Character
/dev/tty2	Second local terminal on the system (Ctrl+Alt+F2)	Character
/dev/ttyS0	First serial port on the system (COM1)	Character
/dev/ttyS1	Second serial port on the system (COM2)	Character
/dev/psaux	PS/2 mouse port	Character
/dev/lp0	First parallel port on the system (LPT1)	Character
/dev/null	Device file that represents nothing; any data sent to this device is discarded	Character
/dev/st0	First SCSI tape device in the system	Character
/dev/bus/usb/*	USB device files	Character

/dev Versioning

- ls –l /dev



Major number

Minor number

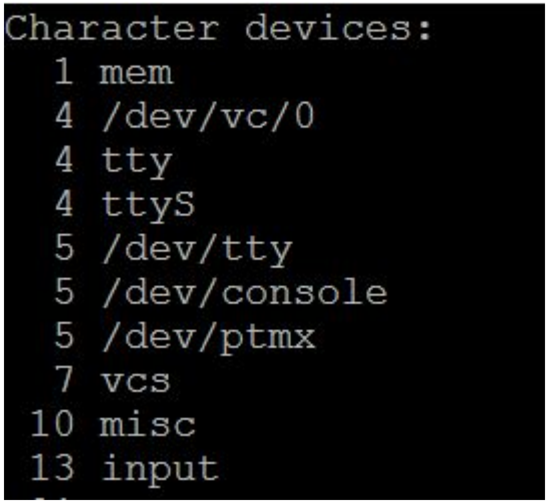
```
[fredericks@SciLinux6 ~]$ ls -l /dev
total 0
crw-rw----. 1 root video    10, 175 Jul 20 14:28 agpgart
crw-rw----. 1 root root     10,  55 Jul 20 14:28 autofs
drwxr-xr-x. 2 root root         660 Jul 20 14:28 block
drwxr-xr-x. 2 root root          80 Jul 20 14:28 bsg
crw-------. 1 root root     10, 234 Jul 20 14:28 btrfs-control
lrwxrwxrwx. 1 root root           3 Jul 20 14:28 cdrom -> sr0
lrwxrwxrwx. 1 root root           3 Jul 20 14:28 cdrw -> sr0
drwxr-xr-x. 2 root root        2600 Aug 12 14:33 char
crw-------. 1 root root      5,   1 Jul 20 14:28 console
lrwxrwxrwx. 1 root root          11 Jul 20 14:28 core -> /proc/kcore
drwxr-xr-x. 3 root root          80 Jul 20 14:28 cpu
crw-rw----. 1 root root     10,  61 Jul 20 14:28 cpu_dma_latency
crw-rw----. 1 root root     10,  62 Jul 20 14:28 crash
drwxr-xr-x. 5 root root         100 Jul 20 14:28 disk
brw-rw----. 1 root disk    253,   0 Jul 20 14:28 dm-0
```

/dev Versioning

- Major number
  - Points to driver in Linux kernel
  - Several devices can share major number if similar type
- Minor number
  - Specific device
  - Minor number differentiates particular device
- Device type (character / block), major number, minor number define unique characteristics of device

Device File

- Device driver
  - Software that operates device
  - Part of monolithic kernel (e.g., Linux)
- Lookup existing devices
  - View content of /proc/devices
    - Major number | Device



```
Character devices:
  1 mem
  4 /dev/vc/0
  4 tty
  4 ttyS
  5 /dev/tty
  5 /dev/console
  5 /dev/ptmx
  7 vcs
 10 misc
 13 input
```

- **mknod** : Re-create corrupted device
  - Requires file type, major number, minor number

Filesystems Revisited

- Organization and management infrastructure for media device
  - HDD, USB, etc.
  - All require a filesystem before it can be used

- Formatting device
  - Create filesystem on device

- Filesystems share three common components:
  - Superblock : filesystem metadata
  - inode table : table of inodes (metadata of files)
  - Data blocks : data referenced by inodes

Filesystem Revisited

- Superblock
  - Filesystem characteristics
    - Size
    - Block size
    - Inode tables (size / count)
    - Disk block map
    - Block usage
    - Etc.

Filesystems Revisited

- inode
  - Each file has entry in inode table
  - Extended info:
    - Filename
    - File size
    - File owner
    - Group
    - Permissions
    - Pointer to disk block that contains data

Filesystems

File System	Max File Size	Max Partition Size	Journaling	Notes
Fat16	2 GB	2 GB	No	Legacy
Fat32	4 GB	8 TB	No	Legacy
NTFS	2 TB	256 TB	Yes	(For Windows Compatibility) NTFS-3g is installed by default in Ubuntu, allowing Read/Write support
ext2	2 TB	32 TB	No	Legacy
ext3	2 TB	32 TB	Yes (ordered)	Standard linux filesystem for many years
ext4	16 TB	1 EB	Yes	Modern iteration of ext3
reiserFS	8 TB	16 TB	Yes (writeback)	No longer well-maintained.
JFS	4PB	32PB	Yes (ordered)	Created by IBM - Not well maintained.
XFS	8 EB	8 EB	Yes (writeback)	Created by SGI. Best choice for a mix of stability and advanced journaling.

https://help.ubuntu.com/community/LinuxFilesystemsExplained

Other Common Filesystems

- vfat
  - DOS filesystem (long filename support)

- iso9660
  - Standard CD / DVD / ISO

Journaling vs. Copy-on-Write

- Journaling
  - Write file changes to temporary file (journal) prior to writing to storage / inode table
  - Journal deleted after write success
  - Why?

- Three common journaling approaches
  - ***Data mode***: inode and file data journaled
    - Good at keeping data, bad at performance

  - ***Writeback mode***: inode only written, no control over file data
    - High risk of data loss, but better performance

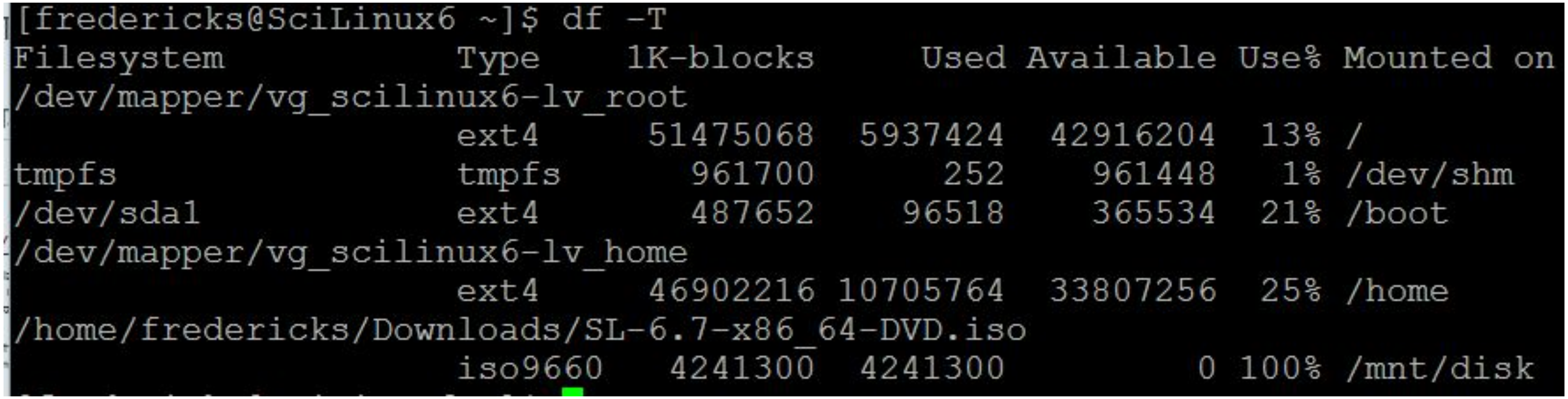
  - ***Ordered mode***: inode only written, but not removed until file data is written
    - Compromise

Journaling vs. Copy-on-Write

- Copy-on-write
  - Snapshots of data rather than writing to temp file
    - Create new file rather than updating over old file
    - Old data not overwritten
    - Snapshot includes superblock and inode metadata
  - ZFS (Sun)
    - Comparable ot Resier4, Btrfs, ext4
    - No GPL license
  - Btrfs (Oracle)
    - Based on Resier4
    - Used as default in openSUSE
    - Offered in RHEL (and therefore, Scientific), but not as default

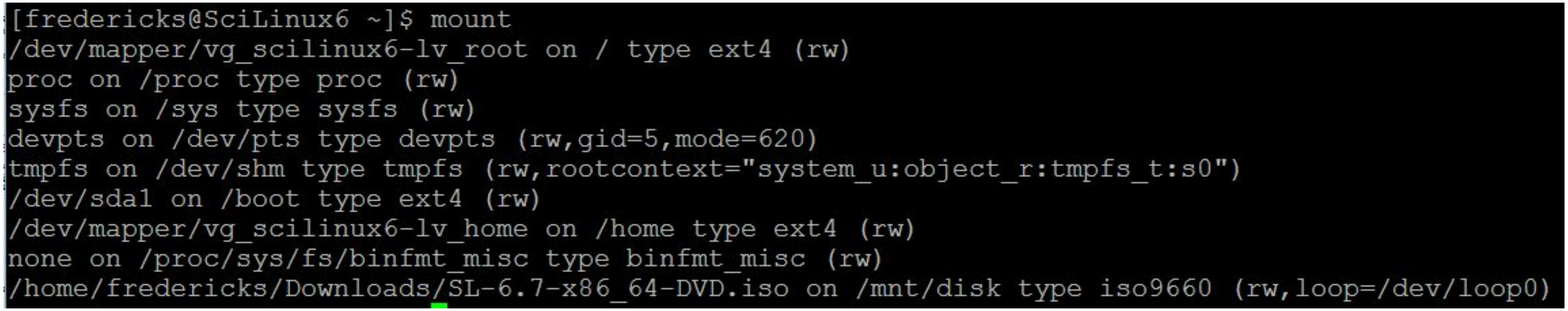
Which Filesystem(s) am I using?

- **df** –T



```
[fredericks@SciLinux6 ~]$ df -T
Filesystem                 Type     1K-blocks      Used Available Use% Mounted on
/dev/mapper/vg_scilinux6-lv_root
                           ext4      51475068   5937424  42916204  13% /
tmpfs                      tmpfs       961700       252    961448   1% /dev/shm
/dev/sda1                  ext4        487652     96518    365534  21% /boot
/dev/mapper/vg_scilinux6-lv_home
                           ext4      46902216  10705764  33807256  25% /home
/home/fredericks/Downloads/SL-6.7-x86_64-DVD.iso
                           iso9660    4241300   4241300         0 100% /mnt/disk
```

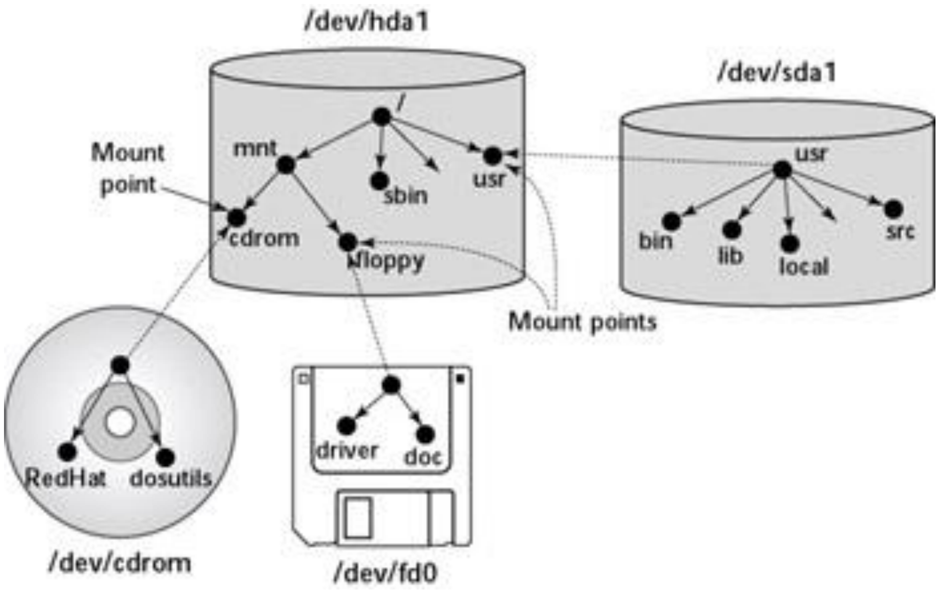
- **mount**



```
[fredericks@SciLinux6 ~]$ mount
/dev/mapper/vg_scilinux6-lv_root on / type ext4 (rw)
proc on /proc type proc (rw)
sysfs on /sys type sysfs (rw)
devpts on /dev/pts type devpts (rw,gid=5,mode=620)
tmpfs on /dev/shm type tmpfs (rw,rootcontext="system_u:object_r:tmpfs_t:s0")
/dev/sda1 on /boot type ext4 (rw)
/dev/mapper/vg_scilinux6-lv_home on /home type ext4 (rw)
none on /proc/sys/fs/binfmt_misc type binfmt_misc (rw)
/home/fredericks/Downloads/SL-6.7-x86_64-DVD.iso on /mnt/disk type iso9660 (rw,loop=/dev/loop0)
```

Mounting

- Make device accessible to user
  - Adding media to virtual directory



Mounting

- Make device accessible to user
  - Adding media to virtual directory

- Mount point
  - Directory where device is attached
  - Any existing directory can be used as a mount point!
    - Create empty directory to prevent issues with existing files

- Graphical desktop tends to auto-mount added devices, discs, etc.
  - Console you get to do it by hand! Yay!

Mounting

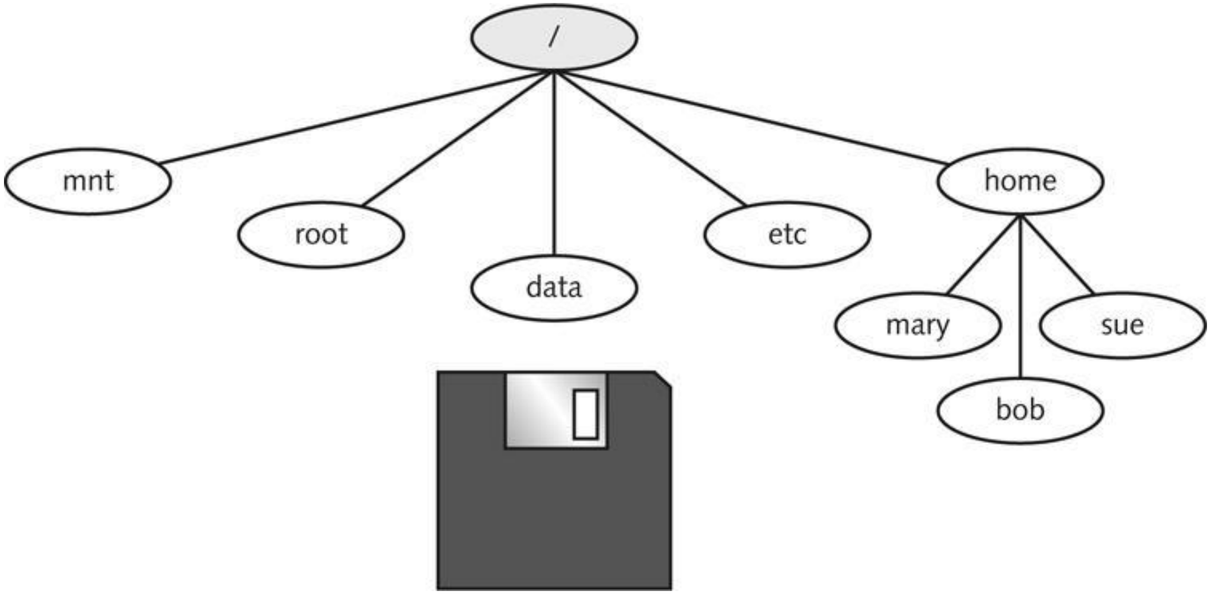


Figure 5-1: The directory structure prior to mounting

Mounting

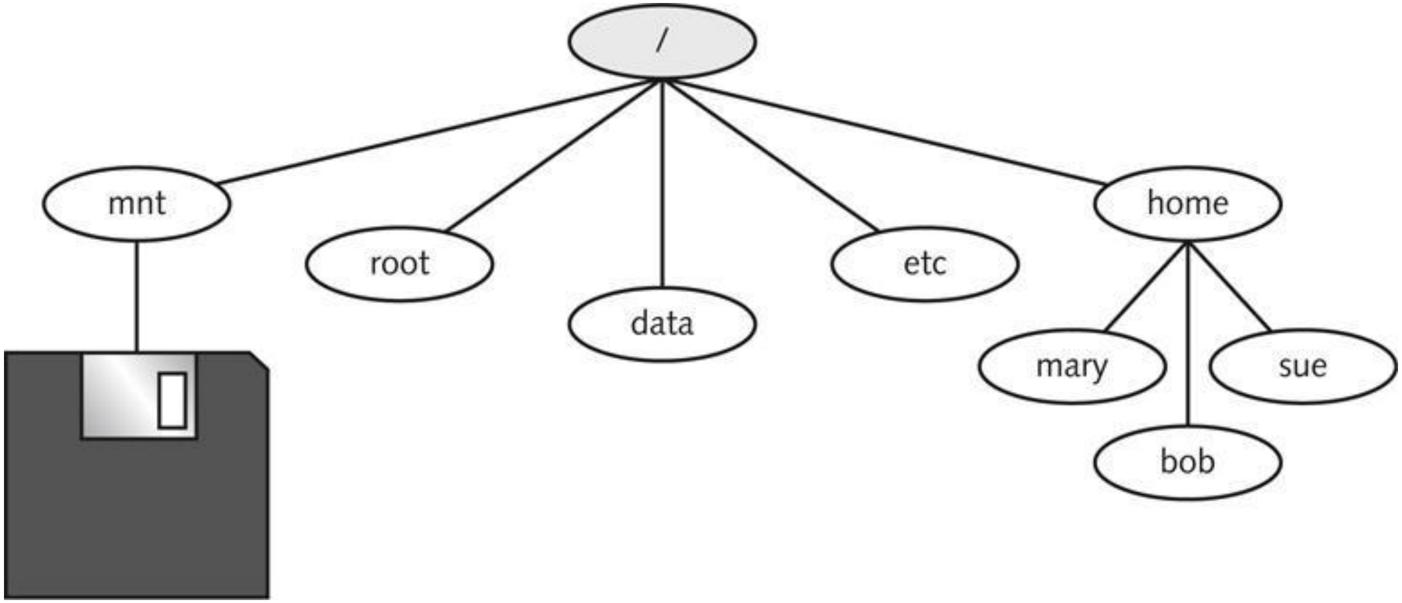


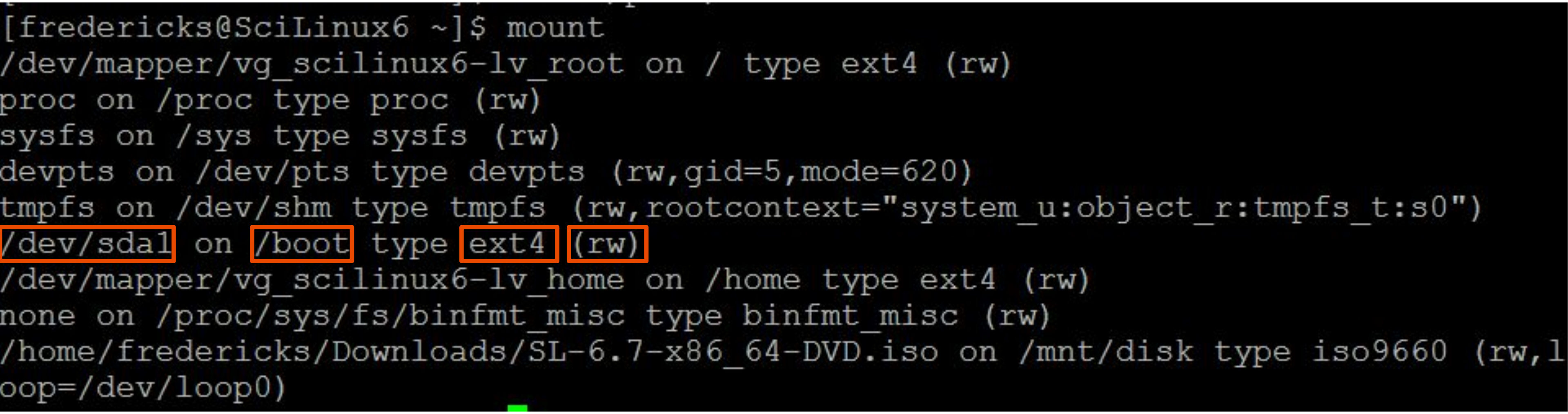
Figure 5-2: The directory structure after mounting a floppy device

Root Filesystem

- System is mounted to / when it is turned on
  - / contains most OS-specific files
- **mount** : Mount device to mount point
  - No options : lists currently mounted devices
    - (1) Device filename
    - (2) Mount point in virtual directory
    - (3) Type of filesystem
    - (4) Access status
- **umount** : Unmount device from mount point

mount

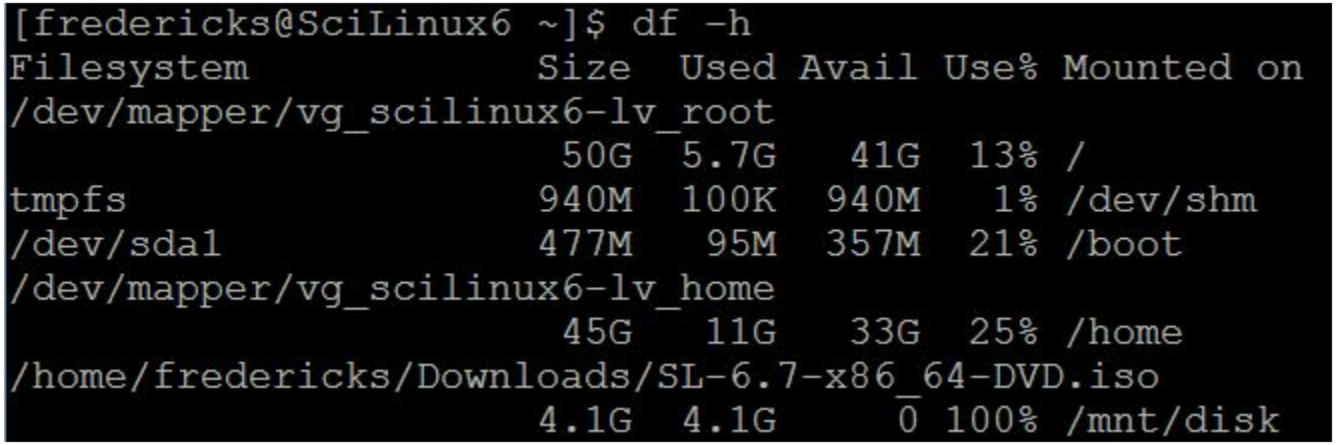
- (1) Device filename
- (2) Mount point in virtual directory
- (3) Type of filesystem
- (4) Access status



```
[fredericks@SciLinux6 ~]$ mount
/dev/mapper/vg_scilinux6-lv_root on / type ext4 (rw)
proc on /proc type proc (rw)
sysfs on /sys type sysfs (rw)
devpts on /dev/pts type devpts (rw,gid=5,mode=620)
tmpfs on /dev/shm type tmpfs (rw,rootcontext="system_u:object_r:tmpfs_t:s0")
/dev/sda1 on /boot type ext4 (rw)
/dev/mapper/vg_scilinux6-lv_home on /home type ext4 (rw)
none on /proc/sys/fs/binfmt_misc type binfmt_misc (rw)
/home/fredericks/Downloads/SL-6.7-x86_64-DVD.iso on /mnt/disk type iso9660 (rw,l
oop=/dev/loop0)
```

What HDDs exist?

- **df** *–h*



```
[fredericks@SciLinux6 ~]$ df -h
Filesystem            Size  Used Avail Use% Mounted on
/dev/mapper/vg_scilinux6-lv_root
                       50G  5.7G   41G  13% /
tmpfs                 940M  100K  940M   1% /dev/shm
/dev/sda1             477M   95M  357M  21% /boot
/dev/mapper/vg_scilinux6-lv_home
                       45G   11G   33G  25% /home
/home/fredericks/Downloads/SL-6.7-x86_64-DVD.iso
                      4.1G  4.1G     0 100% /mnt/disk
```

Mount/Unmount Example

I downloaded an ISO file (Sci Linux 7)

Mounting:

```
sudo mkdir /mnt/scilinux7
```

```
sudo mount -t iso9660 -o loop ~/sci-linux-7.iso
/mnt/scilinux7
```

Unmounting:

```
sudo umount /mnt/scilinux7
```

Overview Review

df -- disk space usage

du -- estimate file space usage

mount -- mount a filesystem (and report mount points)

umask -- _ _ _ _

- 0755 — None of the special bits set
- 1755 — Sticky bit set
- 2755 — SGID bit set
- 4755 — SUID bit set

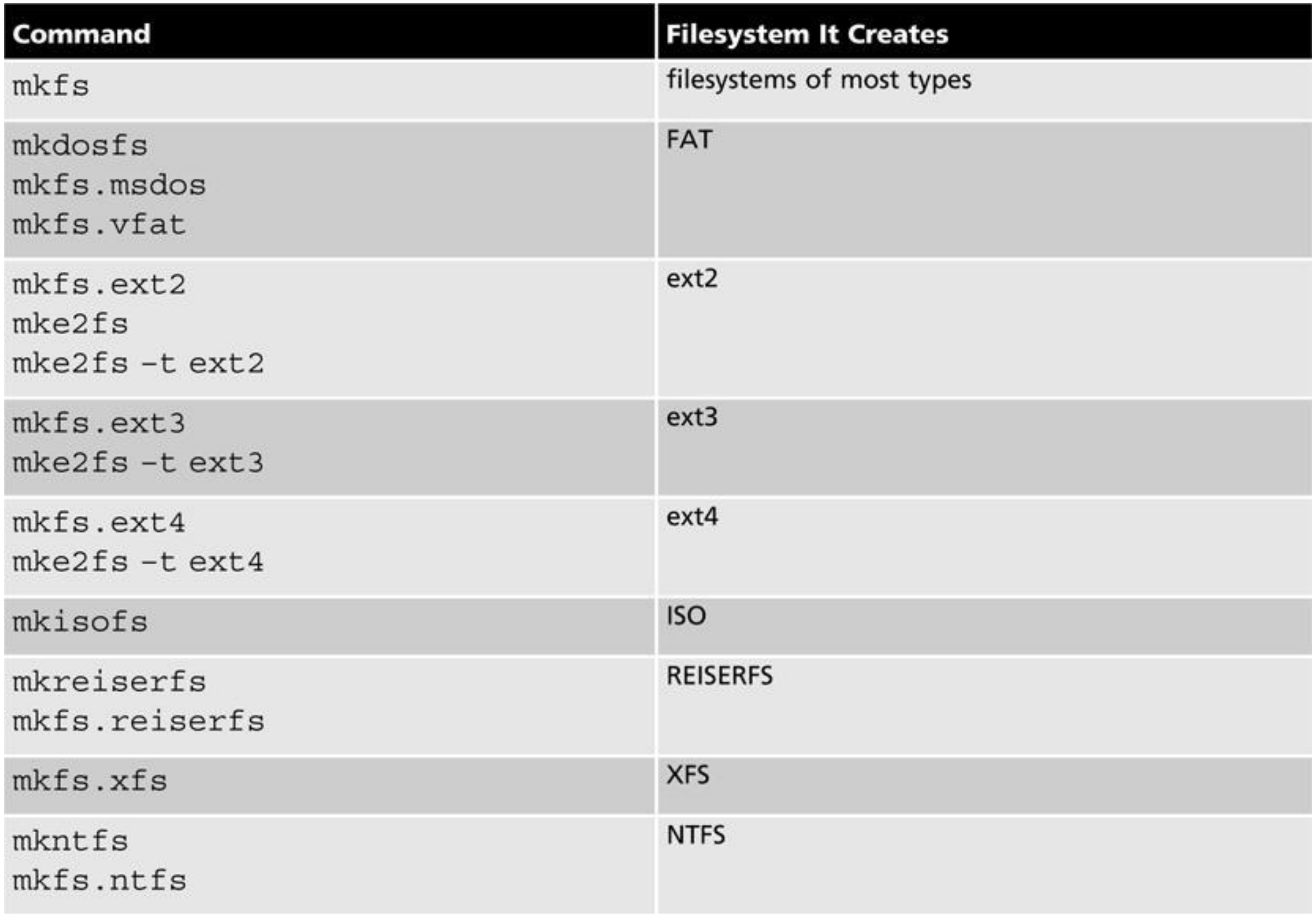
Creating Filesystems

- What filesystems can be created using our system
  - **ls** */sbin/mk**

- **mkfs** : make filesystem
  - Format disk device with specified system
  - -t : filesystem type
    - Default is ext2

  - **mkfs** *–t ext3 /dev/sda8*

Filesystem Creation Commands



Command	Filesystem It Creates
`mkfs`	filesystems of most types

`mkdosfs`
`mkfs.msdos`
`mkfs.vfat`
	FAT

`mkfs.ext2`
`mke2fs`
`mke2fs –t ext2`
	ext2

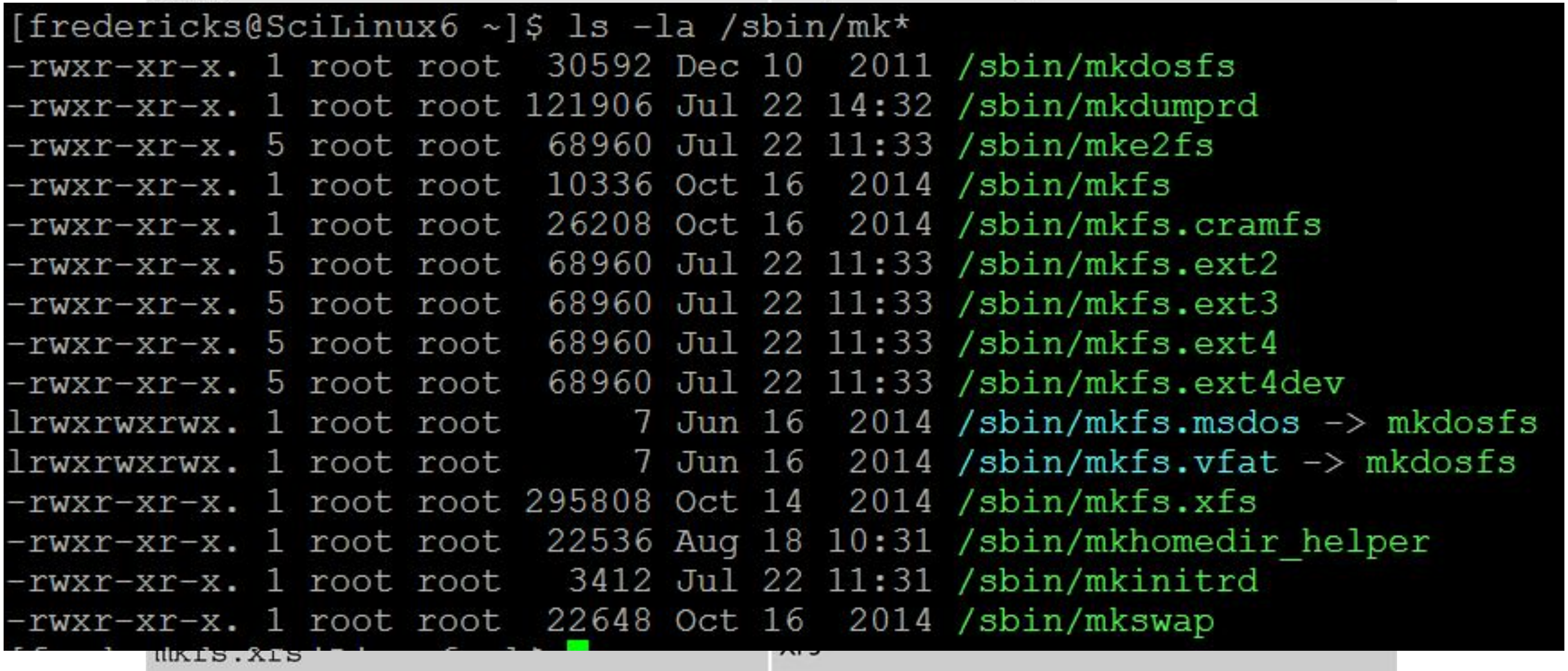
`mkfs.ext3`
`mke2fs –t ext3`
	ext3

`mkfs.ext4`
`mke2fs –t ext4`
	ext4

`mkisofs`	ISO
`mkreiserfs` `mkfs.reiserfs`	REISERFS
`mkfs.xfs`	XFS
`mkntfs` `mkfs.ntfs`	NTFS

Filesystem Creation Commands

Command	Filesystem It Creates
mkfs	filesystems of most types
mkfs.xfs	xfs
mkntfs mkfs.ntfs	NTFS



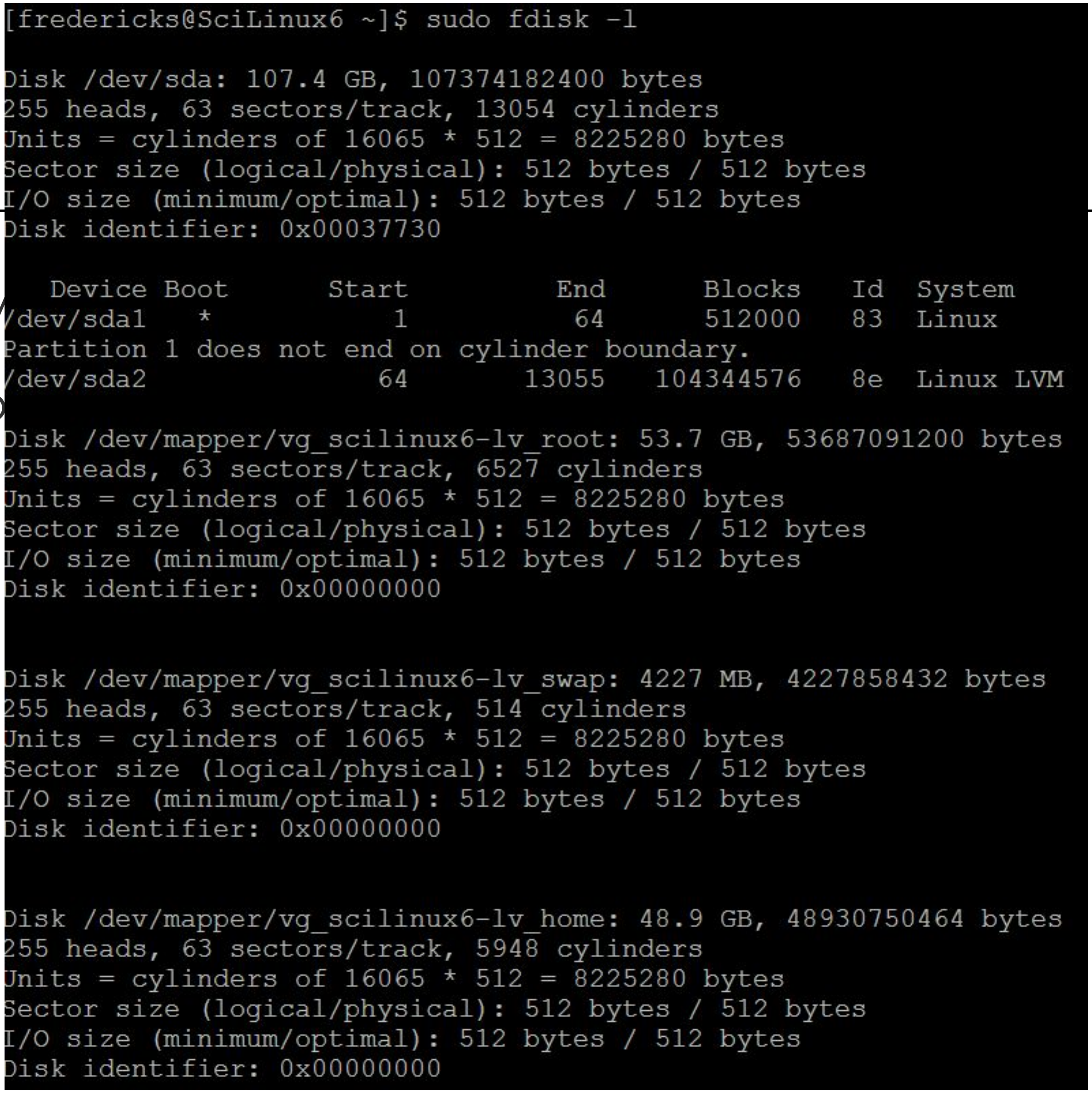
```
[fredericks@SciLinux6 ~]$ ls -la /sbin/mk*
-rwxr-xr-x. 1 root root  30592 Dec 10  2011 /sbin/mkdosfs
-rwxr-xr-x. 1 root root 121906 Jul 22 14:32 /sbin/mkdumprd
-rwxr-xr-x. 5 root root  68960 Jul 22 11:33 /sbin/mke2fs
-rwxr-xr-x. 1 root root  10336 Oct 16  2014 /sbin/mkfs
-rwxr-xr-x. 1 root root  26208 Oct 16  2014 /sbin/mkfs.cramfs
-rwxr-xr-x. 5 root root  68960 Jul 22 11:33 /sbin/mkfs.ext2
-rwxr-xr-x. 5 root root  68960 Jul 22 11:33 /sbin/mkfs.ext3
-rwxr-xr-x. 5 root root  68960 Jul 22 11:33 /sbin/mkfs.ext4
-rwxr-xr-x. 5 root root  68960 Jul 22 11:33 /sbin/mkfs.ext4dev
lrwxrwxrwx. 1 root root      7 Jun 16  2014 /sbin/mkfs.msdos -> mkdosfs
lrwxrwxrwx. 1 root root      7 Jun 16  2014 /sbin/mkfs.vfat -> mkdosfs
-rwxr-xr-x. 1 root root 295808 Oct 14  2014 /sbin/mkfs.xfs
-rwxr-xr-x. 1 root root  22536 Aug 18 10:31 /sbin/mkhomedir_helper
-rwxr-xr-x. 1 root root   3412 Jul 22 11:31 /sbin/mkinitrd
-rwxr-xr-x. 1 root root  22648 Oct 16  2014 /sbin/mkswap
```

Which Device to Format?

- …knowing the approximate size of device helps
- Lookup with (sudo) **fdisk** *–l*
  - Lists all devices
  - sudo fdisk –l <device>
    - Lists particular device

Whic

- …know
- Lookup

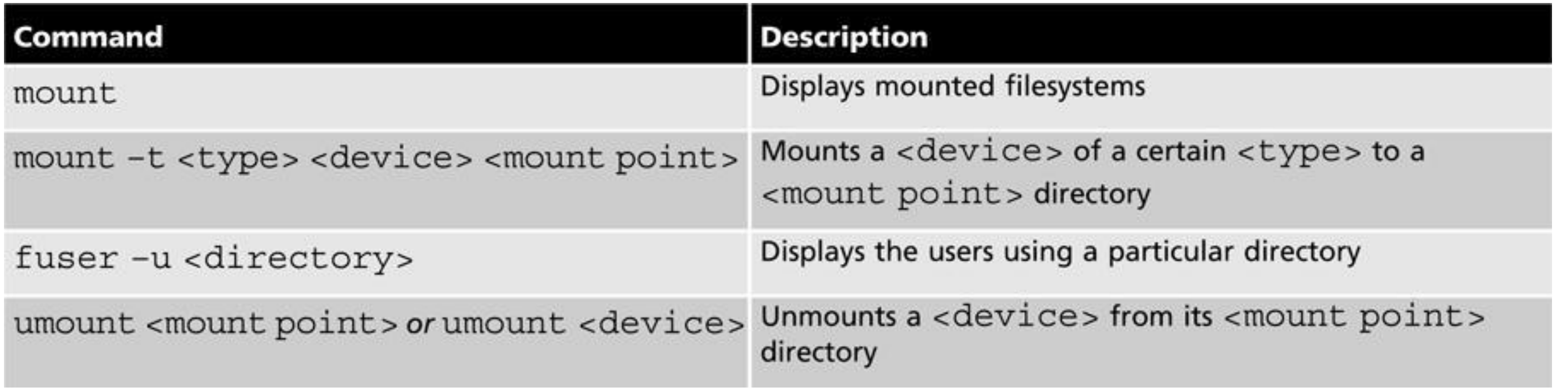


```
[fredericks@SciLinux6 ~]$ sudo fdisk -l

Disk /dev/sda: 107.4 GB, 107374182400 bytes
255 heads, 63 sectors/track, 13054 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00037730

   Device Boot      Start         End      Blocks   Id  System
/dev/sda1   *           1          64      512000   83  Linux
Partition 1 does not end on cylinder boundary.
/dev/sda2              64       13055   104344576   8e  Linux LVM

Disk /dev/mapper/vg_scilinux6-lv_root: 53.7 GB, 53687091200 bytes
255 heads, 63 sectors/track, 6527 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000


Disk /dev/mapper/vg_scilinux6-lv_swap: 4227 MB, 4227858432 bytes
255 heads, 63 sectors/track, 514 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000


Disk /dev/mapper/vg_scilinux6-lv_home: 48.9 GB, 48930750464 bytes
255 heads, 63 sectors/track, 5948 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000
```

Working with Filesystems

- /etc/fstab
  - Mounting is temporary!
  - Mount devices when system boots
    - Add filesystem entry to file
  - Also checked if user doesn’t specify appropriate options when mounting a filesystem

Mount Commands



Command	Description
`mount`	Displays mounted filesystems
`mount -t <type> <device> <mount point>`	Mounts a `<device>` of a certain `<type>` to a `<mount point>` directory
`fuser -u <directory>`	Displays the users using a particular directory
`umount <mount point>` *or* `umount <device>`	Unmounts a `<device>` from its `<mount point>` directory

CD and DVD

- Just another type of media

- Mounted and unmounted as normal (mount / unmount)
  - Device file can be different → dependent on drive technology
  - Common is iso9660 for CD/DVD
    - Read-only → mount with *–r* option
  - Can't be ejected until unmounted
    - **mkdir** */media/cdrom*
    - **mount –t** *iso9660 /dev/scd0 /media/cdrom*

- GUI desktop
  - Automounted
  - Named by CD/DVD label
    - Shortcut created on user's desktop

/mnt vs. /media

- Why do we have different mount points?
  - /media → intended for removable devices
    - E.g., USB key, ISO
  - /mnt → typically for persistent manual mount
    - E.g., internal HDD

- Nothing in the OS to prevent you from using them differently!
  - You can technically mount a device from *any* location
  - Though, not all graphical desktops show devices mounted in /mnt

ISO

- Image (archive) file
  - Typically a backup of a CD/DVD, but can be used as an archive of other files

- Mounted with iso9660 as loopback device
  - su –
  - mkdir –p /media/disk
  - mount –o loop CIT348.iso /media/disk
    - (Double click on ISO in GUI will also auto-mount file)

- **mkisofs** : create ISO from directory
  - Args: name of ISO, directory to create from

Mounting Reference

- **mount –t** *type device directory*
  - **mount –t** *vfat /dev/sdb1 /mnt/disk1*

- ISO:
  - **su –**
  - **mkdir –p** */mnt/disk*
  - **mount –o** *loop linux.iso /media/linux-iso*
    - loop → makes file accessible as block device (accessible by file system)

- If mount point is not available…need to create virtual directory!
  - **mkdir** */media/cdrom*
  - **mount –t** *iso9660 /dev/scd0 /media/cdrom*

- Normal user may not have permission to mount!
  - Ensure user is either part of appropriate group, or sudo your way through it

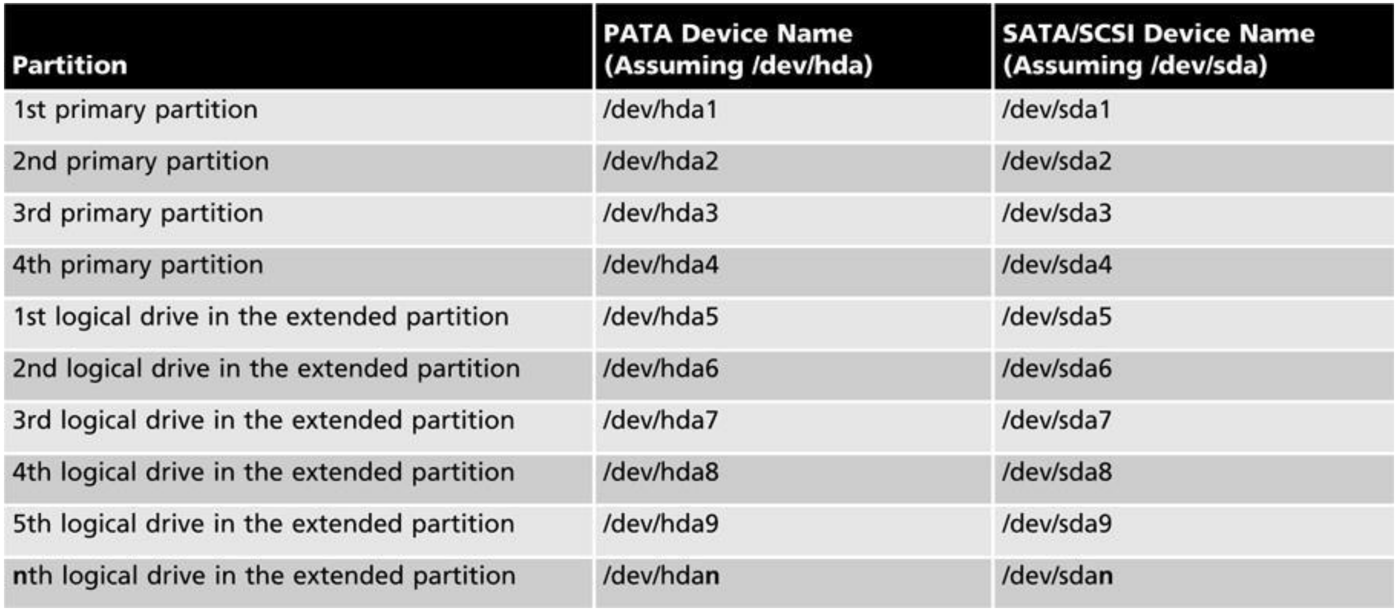
Standard Hard Disk Partitioning

- Partition
  - Division of an HDD
  - Can have its own filesystem

- Linux requires at least two partitions
  - Root (Linux OS / C: drive)
  - Swap (virtual memory)

- It is good practice to use more than two partitions
  - Segregate different types of data
  - Allow for use of multiple filesystem types on one HDD
  - Reduce chance that filesystem corruption will render a system unusable
  - Speed up access to stored data
  - E.g., home directory

Standard Hard Disk Partitioning



Partition	PATA Device Name (Assuming /dev/hda)	SATA/SCSI Device Name (Assuming /dev/sda)
1st primary partition	/dev/hda1	/dev/sda1
2nd primary partition	/dev/hda2	/dev/sda2
3rd primary partition	/dev/hda3	/dev/sda3
4th primary partition	/dev/hda4	/dev/sda4
1st logical drive in the extended partition	/dev/hda5	/dev/sda5
2nd logical drive in the extended partition	/dev/hda6	/dev/sda6
3rd logical drive in the extended partition	/dev/hda7	/dev/sda7
4th logical drive in the extended partition	/dev/hda8	/dev/sda8
5th logical drive in the extended partition	/dev/hda9	/dev/sda9
nth logical drive in the extended partition	/dev/hdan	/dev/sdan

Table 5-5: Common hard disk partition device files for /dev/hda and /dev/sda

Partition Types

- Primary
    - Boot an operating system
- Extended
    - Contain logical drives
    - Logical drive
        - Files unrelated to operating system
            - Music, photos, games, etc.

Standard Hard Disk Partitioning

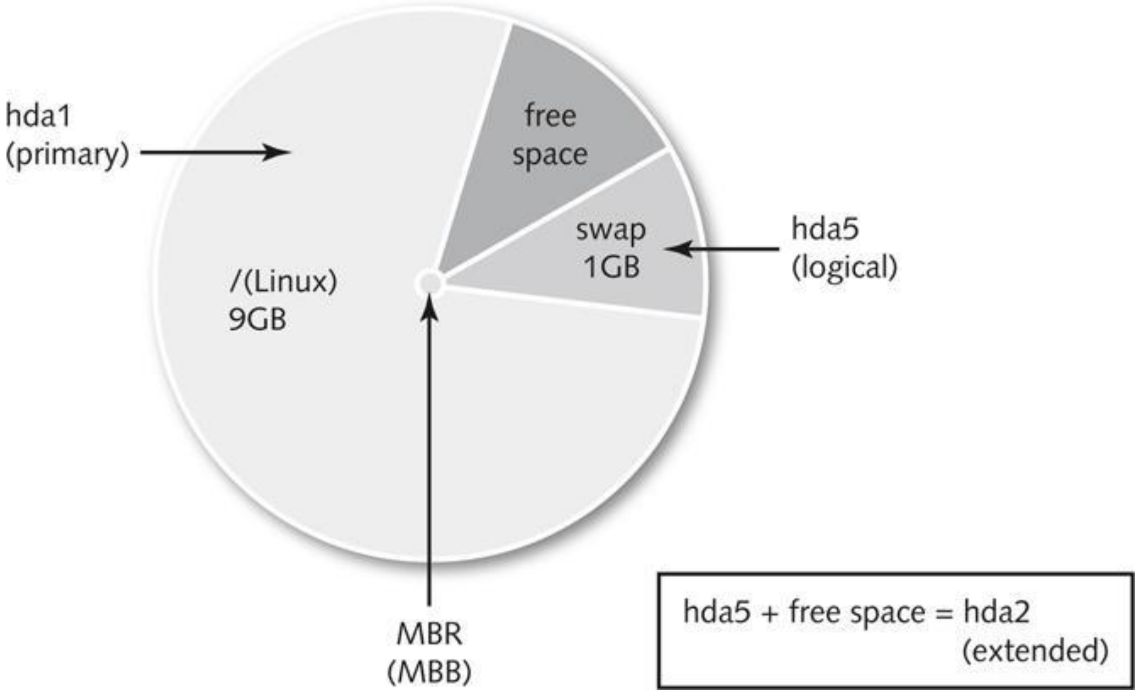


Figure 5-5: A sample Linux partitioning strategy

Standard Hard Disk Partitioning

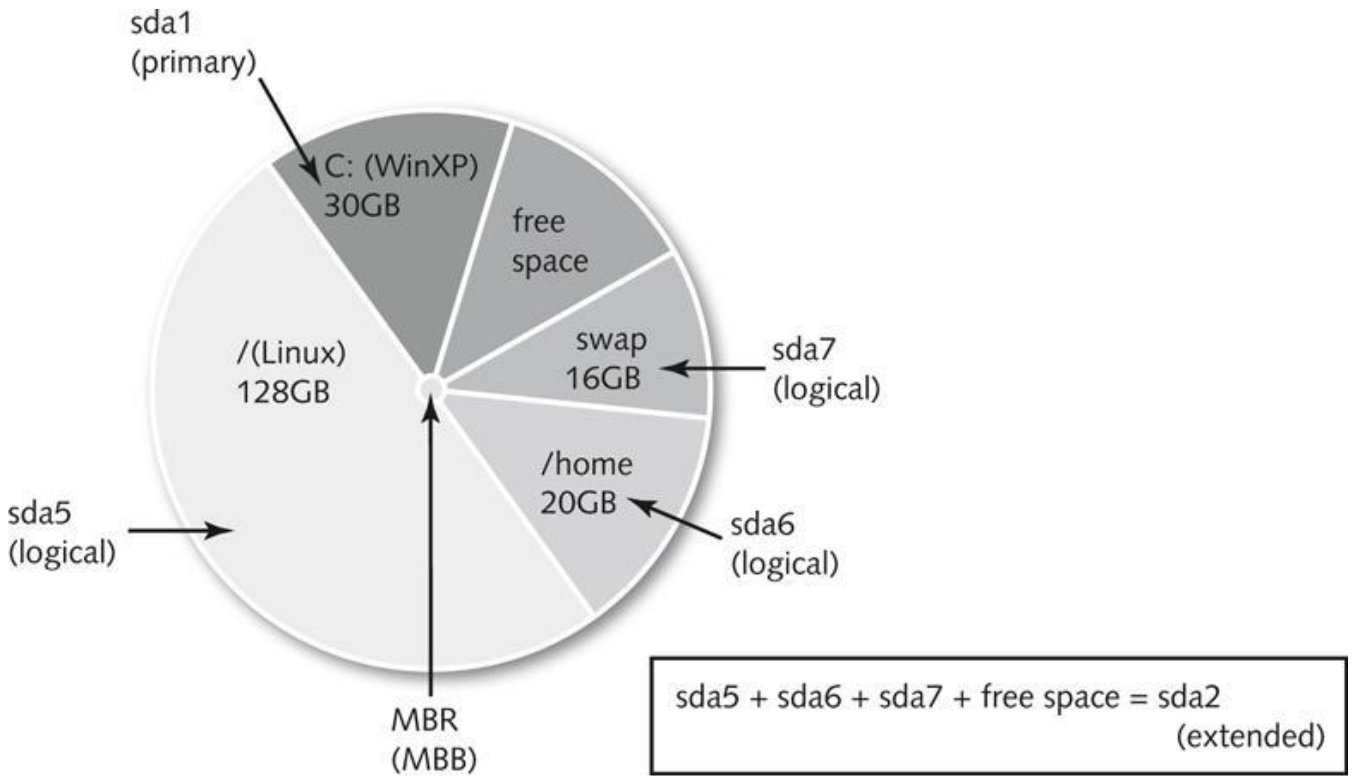


Figure 5-6: A sample dual-boot Linux partitioning strategy

Red Hat Recommended

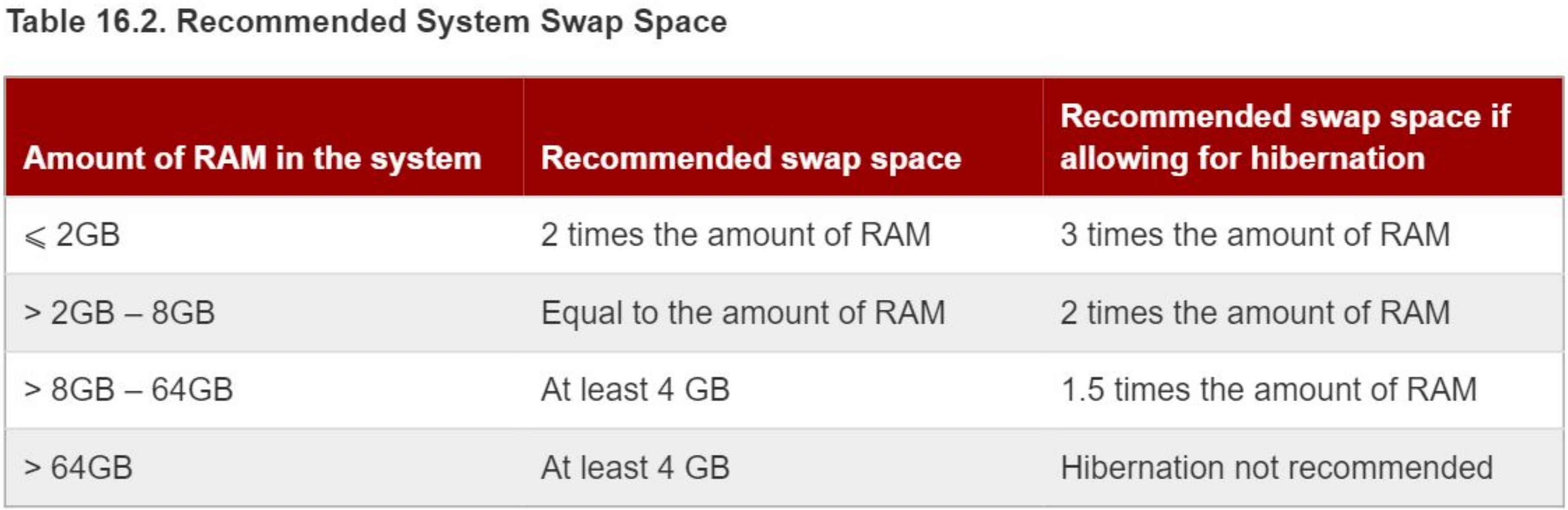


Table 16.2. Recommended System Swap Space

Amount of RAM in the system	Recommended swap space	Recommended swap space if allowing for hibernation
⩽ 2GB	2 times the amount of RAM	3 times the amount of RAM
> 2GB – 8GB	Equal to the amount of RAM	2 times the amount of RAM
> 8GB – 64GB	At least 4 GB	1.5 times the amount of RAM
> 64GB	At least 4 GB	Hibernation not recommended

https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/Installation_Guide/s2-diskpartrecommend-ppc.html#id4394007

Working with Standard Hard Disk Partitions

- **fdisk** command: Create partitions
- Specify hard disk partition as an argument
  - Variety of options for **fdisk** prompt to achieve different tasks
  - Reboot computer after using **fdisk** to ensure proper reloading into memory
  - E.g., **fdisk** /dev/sda
    - Create partition using all space
  - After process is completed, ensure you issue (**w**)rite command to finalize

Working with Standard Hard Disk Partitions

- Edit /etc/fstab file to allow system to mount new filesystems automatically at boot time
- Swap partitions do not contain a filesystem
  - You must still prepare and activate them for Linux
  - Why additional swap over the standard installed swap?
- **mkswap** command: Prepare the swap partition
- **swapon** command: Activate the swap partition
- **swapoff** command: Deactivate the swap partition
- Edit /etc/fstab file to ensure that new swap partition is activated as virtual memory

Setting up a Filesystem (Process Reference)

- (1) Prepare device (partition)
  - Use **fdisk** command
- (2) Create filesystem inside partition (format)
  - Use **mkfs** command
    - **fsck** to check for errors
- (3) Mount new file system (mount)
  - Use **mount** command to temporarily mount filesystem
  - Edit /etc/fstab to mount upon startup

- If creating swap partition (additional swap to the standard install)
  - (1) **fdisk**
  - (2) **mkswap**
  - (3) **swapon**

In-class work

- Get together with your team and discuss what each person’s responsibility will be
    - You can’t have one person be the “documentation” person --- everybody must do equivalent technical work

- On that note, consider (and write down) what technologies you think you’d need to build your project
    - Labs will include a LAMP stack, firewall work, scripts for managing users, etc.

- Lastly, what are the major milestones you’ll need to hit to achieve your project goals?

Working with the LVM

- Logical Volume Manager (LVM)
  - Easy way to adjust disk space

  - Used to create volumes
    - Volumes can contain filesystems and can be mounted to directories
    - More flexible than standard partitions – allows use of free space across multiple hard disks
    - Has error correction abilities

  - Components
    - Physical volumes (PVs)
    - Volume group (VG)
    - Logical volumes (LVs)

Working with the LVM

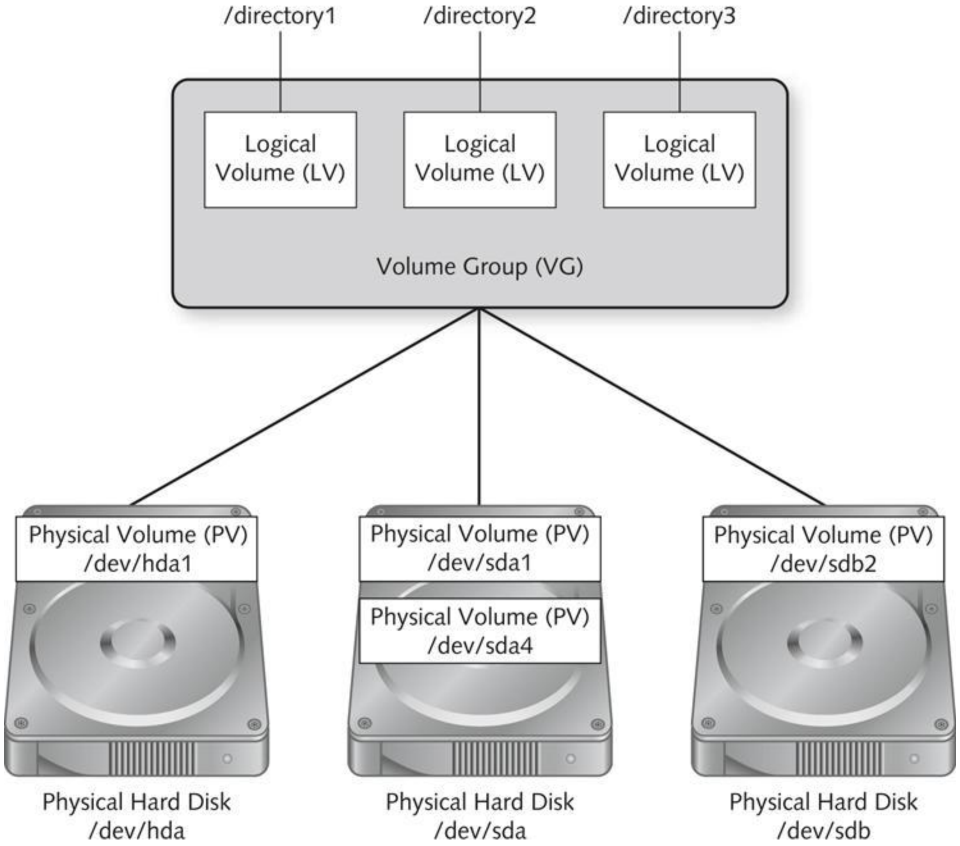

/directory1
/directory2
/directory3
Logical Volume (LV)
Logical Volume (LV)
Logical Volume (LV)
Volume Group (VG)
Physical Volume (PV) /dev/hda1
Physical Volume (PV) /dev/sda1
Physical Volume (PV) /dev/sda4
Physical Volume (PV) /dev/sdb2
Physical Hard Disk /dev/hda
Physical Hard Disk /dev/sda
Physical Hard Disk /dev/sdb

Figure 5-8: A sample LVM configuration

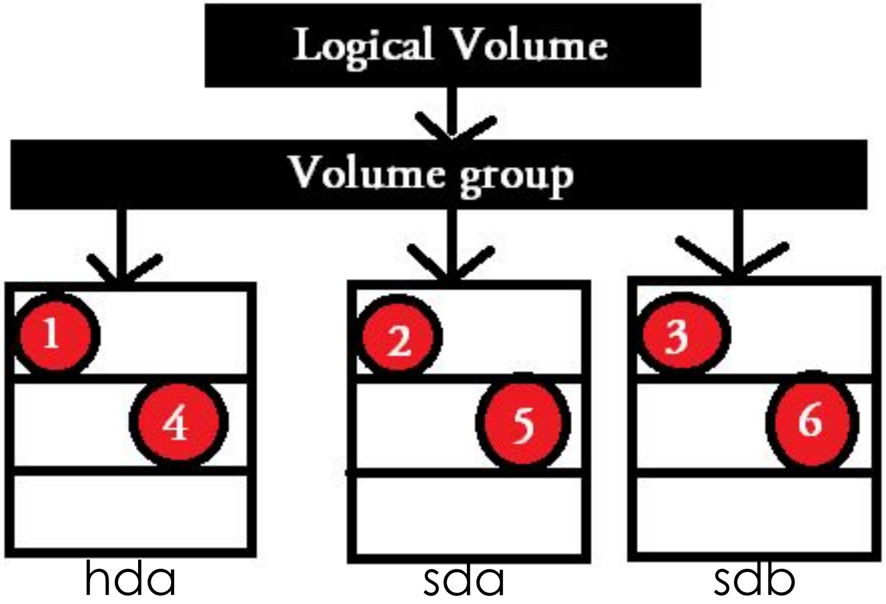
Working with the LVM

- LVM consists of different components:
  - Physical volumes (PVs)
    - Unused partitions on hard disks that the LVM can use to store information
  - Volume groups (VGs)
    - Contain one or more PVs
  - Logical volumes (LVs)
    - Usable volumes that are created by the LVM from the available storage space within a VG

- Linux treats an LVM as any other partition

LVM

- Striping
  - LVM created across multiple HDDs
  - LVM striping != RAID striping
  - No parity
  - Improves disk performance, but can increase chance of data loss
    - If one hard drive goes down, multiple LVMs may be corrupt



- Mirroring
  - Copy of LVM updated in real time
  - Slow performance (large file writes)
  - But, hey, fault tolerance!

Working with the LVM

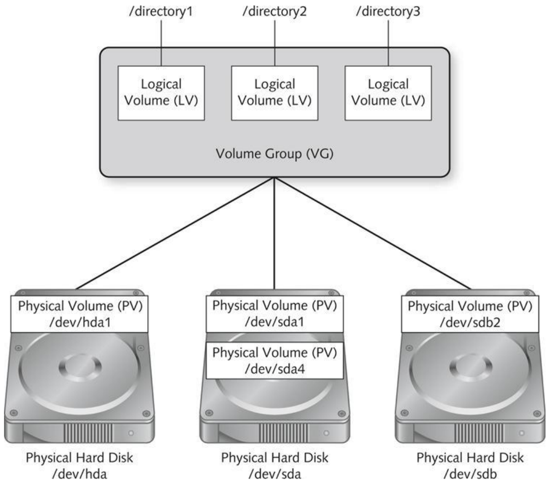
- **pvcreate**
  - Create PVs
- **pvdisplay**
  - Display detailed information about each PV
- **vgcreate**
  - Create a VG that uses the space in PVs
  - Arguments are name of the VG and PVs to be used
- Physical Extent (PE) size: block size for saving data in a VG
  - Should be set when creating a VG
  - Can use **vgcreate** -s to set the PE

Working with the LVM

- **vgdisplay**
  - Display detailed information about each VG
- **lvcreate**
  - Create LVs from available space in a VG
- **lvdisplay**
  - Display information about each LV
- You work with mount points of LVs as you would work with any other hard disk partition device file
  - Edit /etc/fstab to ensure that LVs are automatically mounted at system startup

Working with the LVM

- **pvscan**, **vgscan**, and **lvscan**
  - Display information about PVs, VGs, and LVs, respectively
- **vgextend**
  - Used to add a new PV to an existing VG
- **lvextend**
  - Used to increase the size of an LV
    - E.g., to use space extended onto an existing VG


/directory1
/directory2
/directory3
Logical Volume (LV)
Logical Volume (LV)
Logical Volume (LV)
Volume Group (VG)
Physical Volume (PV) /dev/hda1
Physical Volume (PV) /dev/sda1
Physical Volume (PV) /dev/sda4
Physical Volume (PV) /dev/sdb2
Physical Hard Disk /dev/hda
Physical Hard Disk /dev/sda
Physical Hard Disk /dev/sdb

LVM or Manual Partitioning?

- Manual partitioning
  - Direct control over memory space
  - Simple for single-disk systems
  - Custom memory layouts
- LVM
  - Layer of abstraction between OS and storage devices
    - Combine multiple drives into single space (logical volume)
  - Easy modification of storage space
    - ext4 can be modified without need for LVM
  - Snapshots
  - “Another thing that can go wrong”

Storage Devices

- Removable storage devices are commonly used today
  - Typically connect to your PC via USB or Firewire
  - Examples include:
    - Flash memory drives
    - External hard drives
    - Digital cameras
    - Media players
    - Smartphones
    - Tablets

Working with USB and Firewire-Based Storage Devices

- Most removable storage devices emulate SCSI protocol in the firmware of the device
- Devices are automatically mounted to a new directory under the /media directory named for the label on the device
  - When running GUI
- You must mount in command line!
  - **sudo mkdir** */media/usb*
  - **sudo mount** */dev/sdb1 /media/usb*

Monitoring Filesystems

- Check mounted filesystems periodically for:
  - Errors
  - Disk space usage
  - Inode usage

- Minimizes problems that can occur as a result of a damaged filesystem
  - Reduces the likelihood that a file cannot be saved due to insufficient disk space

Disk Usage

- More filesystems == less hard disk space
  - May result in errors when filesystems fill up with data
  - Periodically remove obsolete files such as old log files to make room for new ones

- **df** (disk filesystem) command
  - Monitor free space used by mounted filesystems
  - `-h` option: to view a more user friendly format
  - To get information about different filesystems, you must mount them prior to using df command

Disk Usage

- **du** (directory usage)
  - View size of a directory and contents in Kilobytes
    - –s option: Summarizes output
    - –h option: More user friendly format

Checking Filesystems for Errors

- Filesystem corruption
  - Errors in filesystem structure preventing retrieval of data
  - Commonly occurs due to improper system shutdown

- Syncing error
  - Process of writing data stored in RAM to the HDD

- Bad blocks
  - Unusable areas of a physical disk
  - Cannot hold a magnetic charge

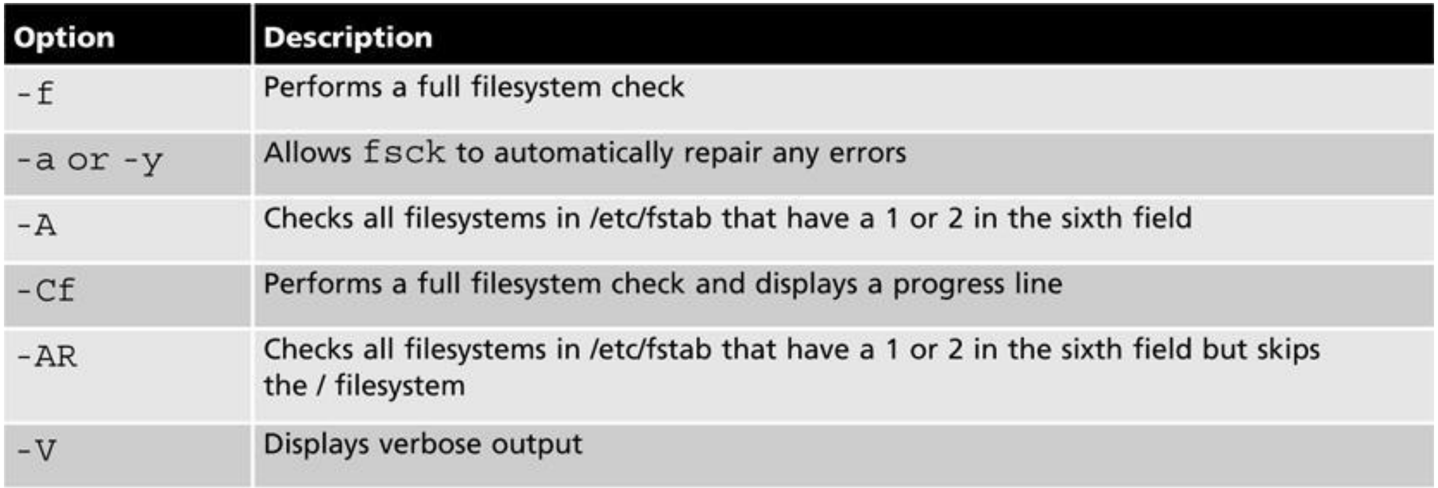
Checking Filesystems for Errors

- **fsck** (filesystem check)
  - Check a filesystem for errors
  - Filesystem must be unmounted
  - –f option used to perform full check

- **e2fsck**
  - Check an ext2, ext3, or ext4 filesystem
  - -c option checks for bad blocks

- **tune2fs**
  - Used to change filesystem parameters
  - -i option sets interval to forcing full system check

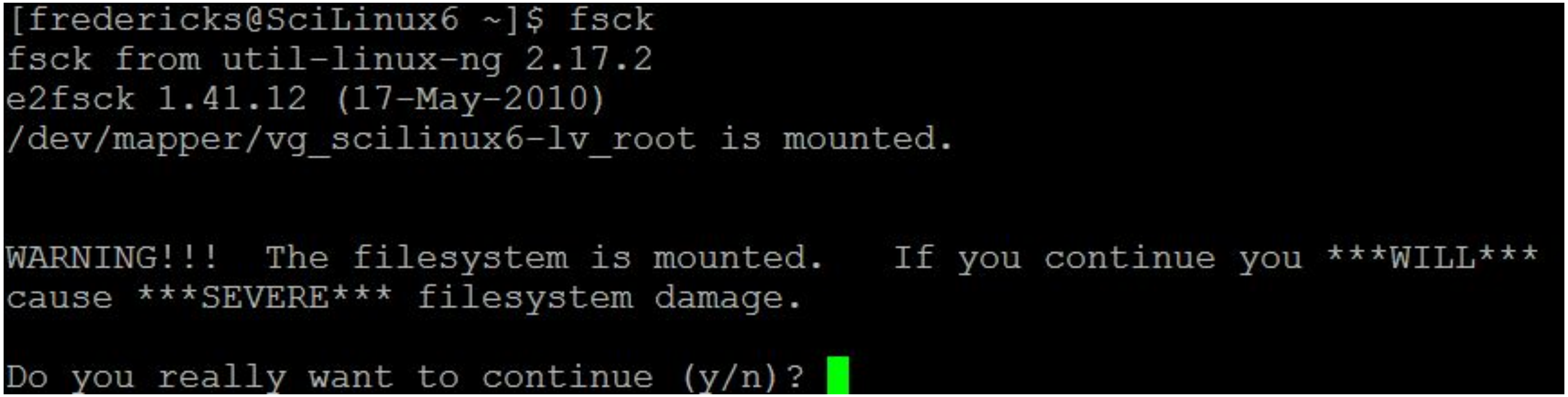
Checking Filesystems for Errors



Option	Description
`-f`	Performs a full filesystem check
`-a` or `-y`	Allows `fsck` to automatically repair any errors
`-A`	Checks all filesystems in /etc/fstab that have a 1 or 2 in the sixth field
`-Cf`	Performs a full filesystem check and displays a progress line
`-AR`	Checks all filesystems in /etc/fstab that have a 1 or 2 in the sixth field but skips the / filesystem
`-V`	Displays verbose output

Table 5-6: Common options to the **`fsck`** command

fsck



```
[fredericks@SciLinux6 ~]$ fsck
fsck from util-linux-ng 2.17.2
e2fsck 1.41.12 (17-May-2010)
/dev/mapper/vg_scilinux6-lv_root is mounted.


WARNING!!!  The filesystem is mounted.   If you continue you ***WILL***
cause ***SEVERE*** filesystem damage.

Do you really want to continue (y/n)?
```

- File checker separate from main system
  - 2 programs potentially reading/modifying data at same time
  - Change filesystem metadata on the fly

- Filesystem mounted as read-only, fsck is read-only, then it should only report issues, not fix them!

Hard Disk Quotas

- If several users on a system, must be enough hard disk space for each user's files
- Hard disk quotas: user limits on filesystem usage
  - Quotas can restrict number of files/directories or total disk space usage
- Soft limit: user may exceed quota briefly
  - For a certain period of time (seven days by default)
- Hard limit: limit cannot be exceeded

Hard Disk Quotas

- **# yum install quota**
- **quotaon** and **quotaoff**
  - Toggle quotas on and off
- **edquota**
  - Edit user quotas
- **repquota**
  - Report user quotas
- **quota**
  - Allows regular users to view their own quotas and current usage

Compression

- Back to Chapter 4!
    - Archiving and compression

Archiving and Compression

- What is the difference between archiving and compression?
  - Archive : set of files stored in single file
    - No space savings
    - **tar** command
      - Historically used to write data to tape backup

  - Compressed file : set of files stored in single file and reduced space
    - gzip, bzip2, zip, compress, 7z, ………………

Archiving

- Files stored in single file without compression
  - **tar** –cvf dir1.tar dir1
    - -c→ create tar archive
    - -v→ verbose output
    - -f → redirect output to file

- Can compress a tar archive
  - gzip
  - **tar** –zcvf dir2.tar.gz dir2
    - -z → redirect to gzip
  - *.tgz or *.tar.gz common
  - **gunzip** dir2.tar.gz → dir2.tar
  - **tar** –zxvf dir2.tar.gz → dir2/

Compression

- **compress :** original UNIX utility
  - Patented to make money
  - .z extension

- **gzip**
  - Performs better than compress anyway
  - Patent-free
  - **gzip** file1.txt → file1.txt.gz
  - **gzip** –d file1.txt.gz → file1.txt
  - **gunzip** file1.txt.gz → file1.txt

Compression

- **bzip2**
  - Uses different compression algorithm
  - Typically smaller filesize than gzip
  - **bzip2** file3.txt → file3.txt.bz2
  - **bunzip2** file3.txt.bz2 → file3.txt

- **zip**
  - UNIX version of PKZIP made for Windows (based on standard ZIP)
  - **zip** file4.txt → file4.txt.zip
  - **unzip** file4.txt.zip → file4.txt

./usr/bin/end_class

- Lab #2 Monday, Oct 2nd

Break