0. Introduction

- We have seen the basement foundations of a UNIX system
  - history
  - philosophy
  - architecture
  - kernel

- We have seen system calls
  - entry points for programs and applications to the UNIX kernel

- We will now move one level higher to study UNIX commands from the
  - advanced user point of view. More admin. commands and topics
  - will be studied next year.

1. Moving into directories

- cd
- cd .
- cd ..
- cd ~user
- cd into symlinks [depends on the shell/command interpreter]
- Explain why cd cannot be a process (although /usr/bin/cd exists)

2. Listing files

- ls
  - ls is often an alias in shell/command interpreter
  - cd
  - alias
    - /bin/ls
    - the real unix command
- ls -l
- ls -s
- ls –d
- ls -s | more –c (note different output - ls use isatty())
- ls -a
- ls -a .??*
- ls -lad .??*
- ls –lt
- ls –lu
- ls -lS
- ls -rtl
- ls -rlS
- ls -rlS -h
- ls -rlS –si (--s not on MacOSX)
- man ls

3. File times

- ls -l --time=use,ctime
  - involved system call: stat structure (man 2 stat)
    - stat()
    - lstat()
    - fstat()
- file times
  - access time read data
  - modified time write data
  - change time inode change

4. Moving/Copying file

- mv
  - inode change if changing fs
- cp
  - cp –p (preserve perm/time - useful for config backup)
  - cp -r
  - cp –i
    - Ask confirmation before overwriting

5. User and group ownership

- /etc/passwd, /etc/group
  - uid, gid, gecos
- /etc/shadow, /etc/gshadow
- useradd, groupadd
- chown
- chgrp
- user & group at the same time... user:group

6. File types and protection

- file types
  - directory, regular, symbolic link, named pipes
  - sockets (unix domain), block/character devices
- file protection modes
- special bits
  - setuid for executables
  - setgid for directories
  - sticky bit for files
  - sticky bit for directories
- special cases
  - rwx for directories

7. Devices files (ne fait plus partie de la matière)

- ls -l /dev directory --> Special files
- Rubini figure 1-1
- Each hardware piece has a software component
- The software component is a device driver
- Devices drivers are connected to an array (index = major number)
    - Minor number identifies sub-devices
- Special files are entry points to connect user procs to devices
- Piece of software (dev. driver) is:
    - statically compiled in the kernel
    - dynamically loaded in the kernel
- lsmod, insmod, rmmod
- Real devices
    - /dev/audio, /dev/dsp, /dev/sdax, /dev/hdax, ...
- Pseudo devices
    - /dev/null, /dev/zero, /dev/random, ...
- Some oddities ... network devices not in the filesystem space
- Connections/Relations between device drivers
- e.g. usb disk -> /dev/sda