Anatomy of a Unix Command

Command-name -options(s) *filename(s) or arguments*

Example: **wc -l** */etc/motd*

The first word of the command line is usually the command name. This may be followed by options, filenames, directory name, or other arguments, and then a RETURN. Option(s) are usually preceded by a dash. The examples in this document use **bold** case for command names and options, and *italics* for arguments and filenames.

UNIX is Case Sensitive

Most UNIX commands are lower case. File and directory names can be lower, upper or mixed case, and must be typed exactly as listed. Commands prefaced by a ^ (caret) mean hold down the CONTROL key while pressing the indicated character.

Documentation

man *command*   Manual pages for *command*
Navigation: **SPACEBAR**   moves down a screen
       q   exit
       h   help

info   GNU Project documentation
Navigation: **SPACEBAR**   move down a screen
       DEL   move back a screen
       TAB   skip to next link
       RETURN   follows link
       q   exit
       h   help tutorial

Printing

Setenv PRINTER *printer*   set the default printer
lp *option file(s)*   print filename(s)

Printer lp is the free line printer in BA-134
Printer laser is 10 cents/page in BA-110
Printer scc is 10 cents/page in Student Computing Center

File System Commands

Create (Make) a Directory

mkdir *directory-name*   create a directory
Option: **-p**   Create any non-existing parent directories

Look at a File

less *filename*   display file contents, similar to **more**
Options:
       SPACEBAR   moves down a screen
       b   moves back a screen (up)
       ^F   follow a file as it grows
       /text   find and highlight *text*
       ?*text*   find (going up) and highlight *text*
       q   exit
       h   help

head *file*   display first 10 lines of a file
sort *file*   sort a file alphabetically (**-n** sort numerically
                                         -r reverse sort)
tail *file*   display last 10 lines of a file
tail –f *file*   display new lines as they appear in the file

List Files and Directories

ls   lists contents of current directory
ls *directory-name*   lists contents of directory

Options:
-a   lists all files including files starting with '.'
-l   long list, shows permissions, owner, size
-t   list files chronologically
-F   append '*' to executables, '/' to directories
-R   recursively list files and directories

Change Working Directory

cd   change to your home directory
cd *directory-name*   change to another directory
pwd   print the name of the working directory

Directory Abbreviations

~	home directory (tilde)
~username	another user's home directory
.	current or working directory
..	parent directory

Move (Rename) Files and Directories

mv *old-name new-name*   rename old to new
mv *filename directory*   move file into directory

Copy Files

cp *filename copy-name*   copy a file into a new filename
cp *filename directory*   copy a file into a directory
Options:
       -i   confirm before overwriting existing file.
       -R   recursive copy (including directories)

Remove (Delete) Files and Directories

rm *filename*   to remove a file
rmdir *directory*   to remove an empty directory
Options:
       -i   interactive, prompts for confirmation

Recover Files and Directories

Backups are run every night around 3AM. Files can be recovered if they existed while the backup was running. Change to the directory where the old file used to exist type **recover**. The **recover** program will prompt "recover List files on the most recent backup by typing **ls**. **changetime** *month day* to restore from a given day. Se files for recovery with **add** *filename(s)*. When finished, recover to restore the file(s). Expect to wait 5 or 10 minu Type **quit** to exit.

Change File Access Permissions

chmod [*who op access*] *filename*
    who can be any combination of:
       u   user   (user is the owner of the file)
       g   group
       o   other
       a   all (ie: ugo)
    op add or removes access, and can be:
       +   add access
       -   remove access
    access can be any combination of:
       r   read   4
       w   write   2
       x   execute   1
Ex: **chmod a+x** *filename* (Add execute access for everyo
      chmod 711 *filename* (Set access to rwx for user,
                              execute for group and other.)

Shell Tools

Wild Cards

? single character wild card
* any number of characters

History: Command Repetition

history display list of recent commands
!! repeat the last command line
!$ repeat the last word of the prior command
!*n* repeat command line *n*
!*string* repeat command beginning with *string*

Command Redirection

> *file* redirect output to a new file
>> *file* append output to existing file
< *file* redirect input from file
| pipe output of one command into another
Ex: **ls -l | less**

Aliases

alias *abbreviation command* create alias
alias *abbreviation* show alias
unalias *abbreviation* remove alias

Alias creates an abbreviation for a command line. For multiple commands, enclose the commands in quotes.
Ex: **alias ls ls -F** (execute **ls -F** when **ls** is used)
    alias big "du -sk .[A-z]* * | sort -rn | head"
                (show biggest files and directories)

File Operations

Search for Patterns in Files

grep *pattern [filenames(s)]* display lines matching the
                                   pattern in the *file(s)*.
Options:
  -v display lines that don't match
  -i match upper or lower case

Patterns can be any combination of:
  string   match string anywhere in a line
  ^string   match if string begins a line
  string$ match if string ends a line
  ab[c,d]   match list, ie: abc or abd
  ab[A-D]   match range, ie: abA, abB, abC, abD

Concatenate and Display Files

cat *file(s)* reads each *file* in sequence and displays them.
Options: **-n** number lines, **-vt** display non-printing characters

Compare Files

diff *file1 file2* compares the contents of *file1* and *file2* and displays the lines that are different.

Counting Words in a File

wc *file* counts the number of works, lines, or characters
Options: **-w** words, **-l** lines, **-c** characters

Compress Files

compress *file*   compress *file* and rename it *file.Z*
uncompress *file.Z* uncompress *file.Z* and rename it *file*
gzip *file*   compress *file* and rename it *file.gz*
gunzip *file.gz*   uncompress *file.gz* or *file.Z* and rename *file*

Process Control

Process Status

ps display the status of current shell's processes
Options:
  -e   display every process (owned by you or others)
  -f   display additional info
  -u *user* show all processes owned by *user*

kill *PID*   soft kill, terminate PID with SIGTERM. May be
                trapped. (Process ID is found by using the **ps**
                command) Every command has it's own PID.
kill –9 *PID* hard kill, terminates PID with SIGKILL

Job (Process) Control

A command may be run in the background by appending an **&** to the end of the command. Then other commands can be typed (in the foreground) while the background command is running.

^Z   stop foreground job (interrupt)
jobs   list any background jobs
bg   resume a stopped job in the background
fg   resume a job in the foreground
fg *%n* resume job *n* in the foreground

Compilers

cc *file.c*   C compiler, use **gcc** for GNU C compiler
CC *file.cc*   C++ compiler, **g++** for GNU C++ compiler
f90 *file.f*   Fortran 90 compiler (and f77 code)
f95 *file.f*   Fortran 95 compiler
Options: **-o** *file*   compile as *file* instead of a.out
          -l *library*   include *library* in compilation
javac *file*   Java compiler for **java** interpreter

Debuggers

dbx *file*   compiler command line debugger
workshop *file* compiler GUI debugger
gdb *file*   GNU compiler command line debugger
ddd *file*   GNU compiler GUI debugger

Helpful Commands

^C   abort command (some commands trap ^C)
^L   redraw screen (works in vi, less, man, top)
env   list environmental variables and values
echo $path list the command search path
locate *file*   show where *file* is on system
which *command*   show where *command* is located

chfn   change your finger info
finger *user* show info about *user* (also displays .plan)
passwd   change your login password
who   list users on the system
w   list system load and users
top   list system info, and top CPU users

clear   clear screen
cal *year* show calendar for *year* (use 2003 not 03)
date   show current time and date
du -k   show disk usage in current directory
find . –size +1000 –ls show files over 500KB in size
nsrwatch   show status of tape backup robot
quota –v   show your disk quota
ssh *host* connect to remote system *host* (tunnels X)

emacs   powerful text and GUI editor
pico   simple text editor
vim (vi)   vi improved visual editor
vilearn   tutorial on how to use the vi editor

elm   text based mail program
pine   text based mail program
tin   USENET News reader