



Massey University



Linux: Getting Started

Andre Barczak

Chris Messom, updated 29/02/08

Martin Johnson updated 26/03/09



Objectives

- This unit provides basic information about **Linux**
- After completing this unit, you should be able to:
 - Logon
 - Execute basic tasks
 - Describe the basic structure
 - Edit files
 - Manage the file system
 - Submit batch jobs



Contents

- 1 - Introduction.
- 2 - Login, work, logout.
- 3 - Structure: file system, process.
- 4 - Commands.
- 5 - Editors.
- 6 - Compile and run a program.
- 7 - Finding help.





1 - Introduction



Where to get more information...

- www.linux.org
- www.redhat.com



What is Linux?

- Open Source Operating System.
- Freely distributed
- Based on Unix
- Being developed by programmers all over the world.
- Very popular Unix brand



History

- 1991 - Linux is created as a hobby by Linus Torvalds, a student at University of Helsinki (Finland)
- 1992 - First public version
- 1993 - First prefabricated Linux distributions
- 1996 - Support for non-Intel processors
- 1999 - Linux 2.2 (sisters' cluster original system)
- 2001 - Linux 2.4 (helix & sisters' current version)
- 2003-2004 Linux 2.6...
- 2004-2008 Rocks cluster





2 - Login, work, logout....



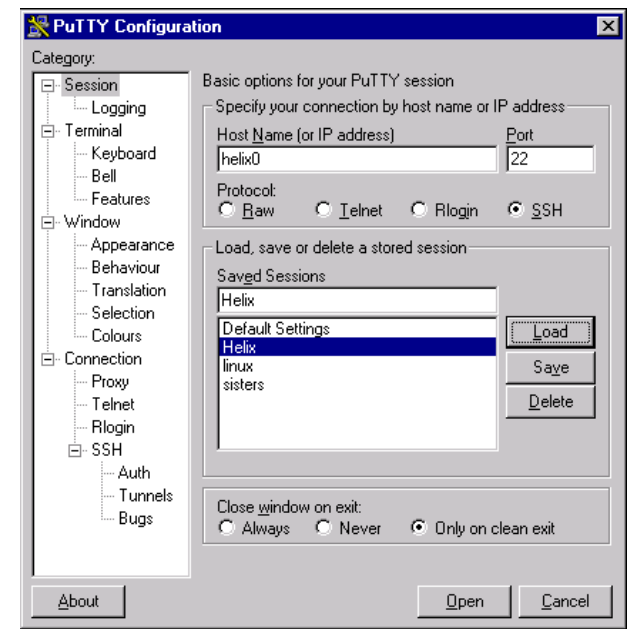
Logging onto iimscluster

- Remotely login using a secure shell (ssh)
 - can start several sessions and also open up Xwindows as long as your terminal can accept connections
- Or
- Login using VNC
 - Start a remote VNC session
 - get a view of your desk top running on the stcluster



Login using ssh

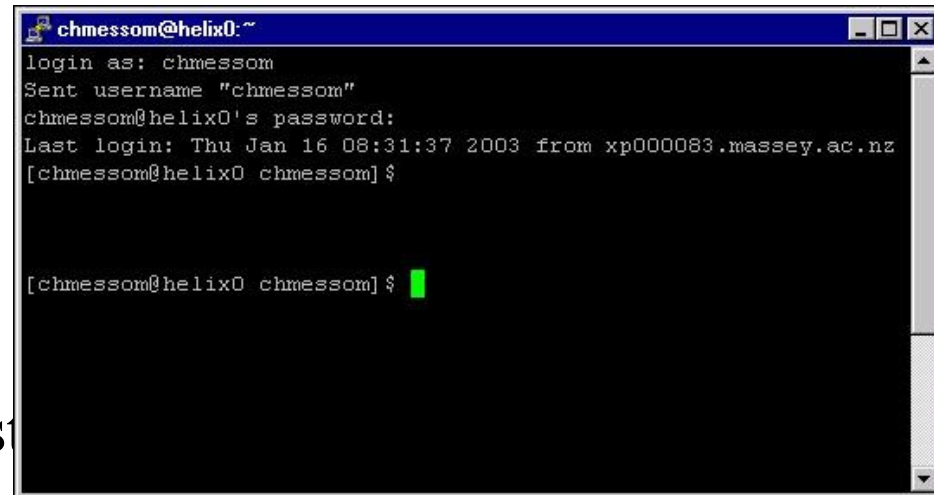
- Get a copy of ssh
 - <http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html> for windows
 - Linux
 - should already have a copy
 - Hostname iimscluster
 - ssh
 - open the connection and login



Logging in

– Login

- Type in username
- Type in password
- start using the iimscluster



```
chmessom@helix0:~  
login as: chmessom  
Sent username "chmessom"  
chmessom@helix0's password:  
Last login: Thu Jan 16 08:31:37 2003 from xp000083.massey.ac.nz  
[chmessom@helix0 chmessom]$  
  
[chmessom@helix0 chmessom]$ █
```

– To open up an xwindow

- run `xwin32` on your windows machine
- Type `export DISPLAY=<machinenum>:0`
 - in the ssh terminal, where `<machinenum>` is your station number e.g. `it001480`
- type `xterm &` to start a new terminal
 - or type `gnome-terminal &`



Logging in using VNC

- Start a VNC server on iimscluster
 - vncserver
 - Note the DISPLAYID
- Start a VNC viewer on your machine (google for vncviewer)
- you'll need to type in
iimscluster:DISPLAYID
this gives you a login dialog box
type your password

For Logging in to VNC from home see ssh tunneling
<http://freesco.no-ip.org/VNC/>

Exiting VNC

- Close the VNC window
- Or log out

- Exiting ssh
- Logout from your ssh session

`logout`

Or

`exit`

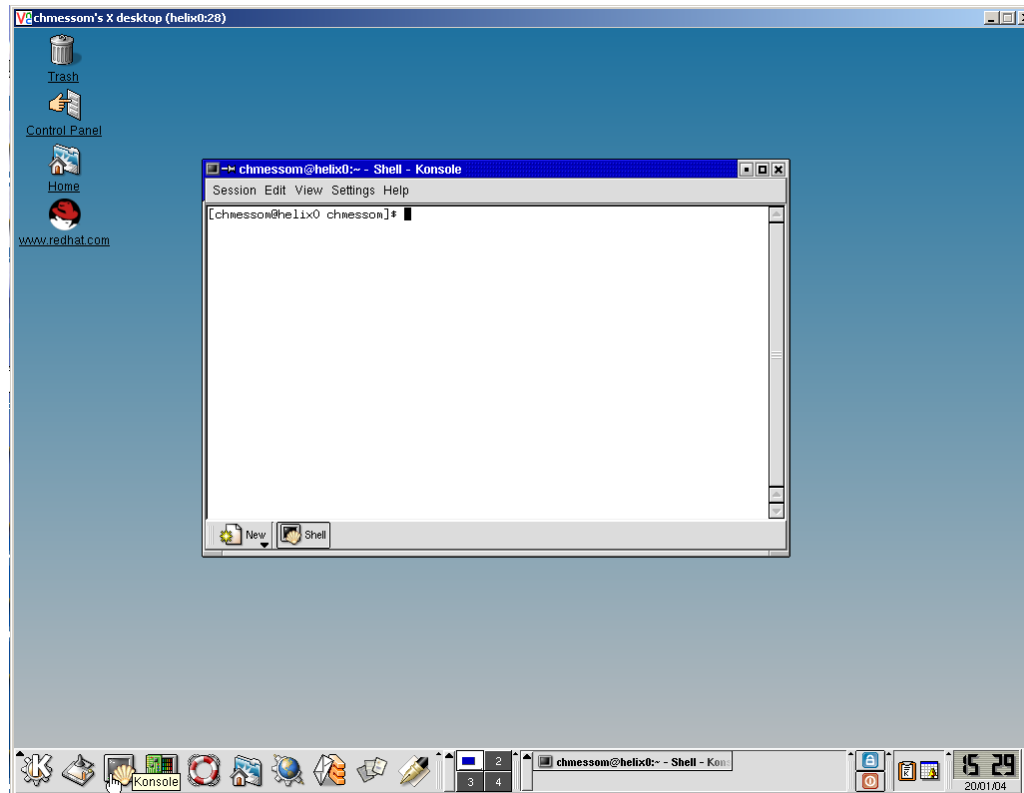


The desktop

- Several desktop environments available to Linux (Gnome and KDE among others).
- Xwindow compatible.
- Drag and drop capabilities.
- Easily customized.



Open a new console window...



Changing Passwords

- Use the `passwd` command
`passwd`
 - note you'll have to type your old password first





3 - Linux Structure

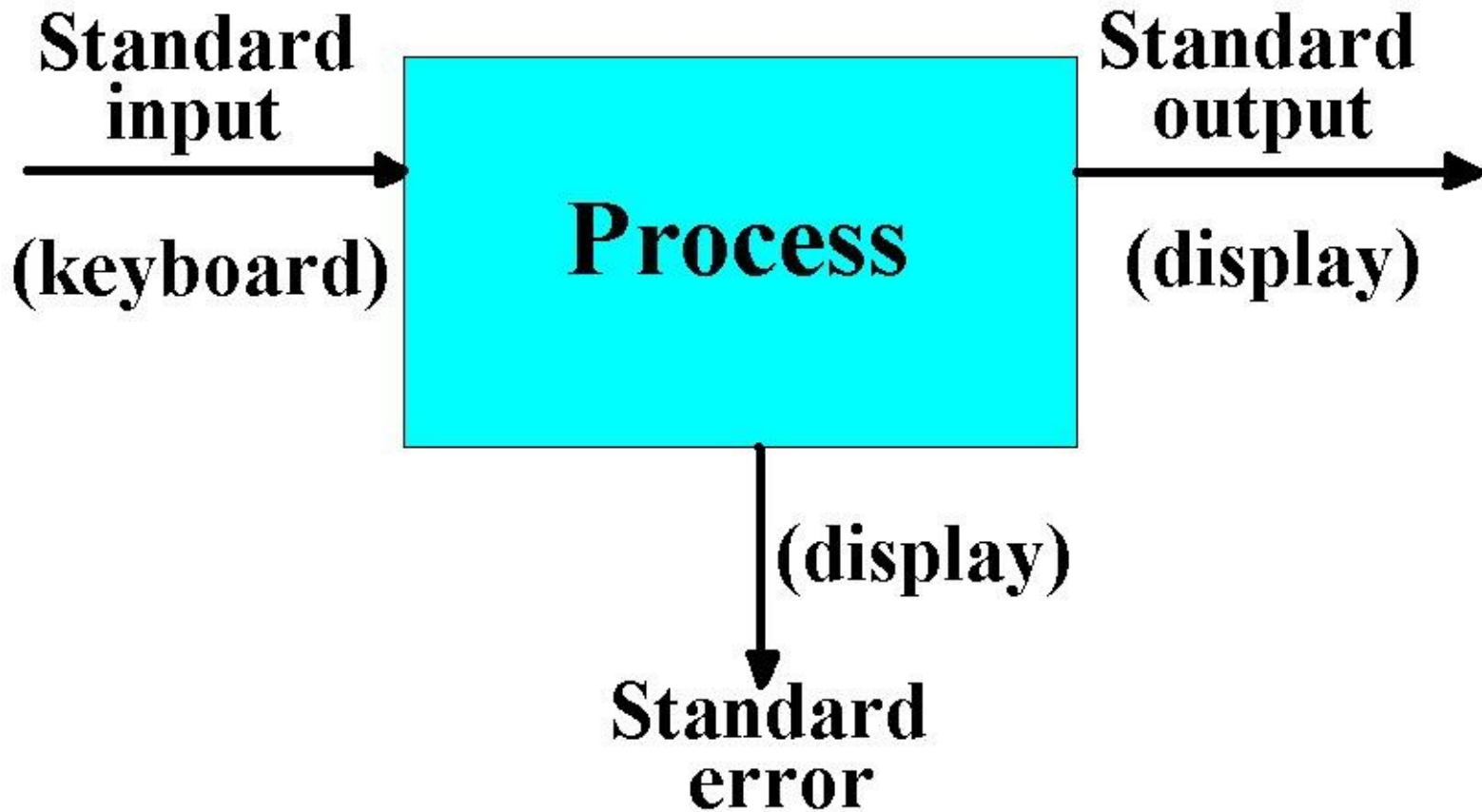


Shells

- Command Interpreters
 - Several ‘shells’ available: `ksh`, `csch`, `bsh`...
- Linux ==> `bash`
- Shell Scripts
 - collections of shell commands that can be run

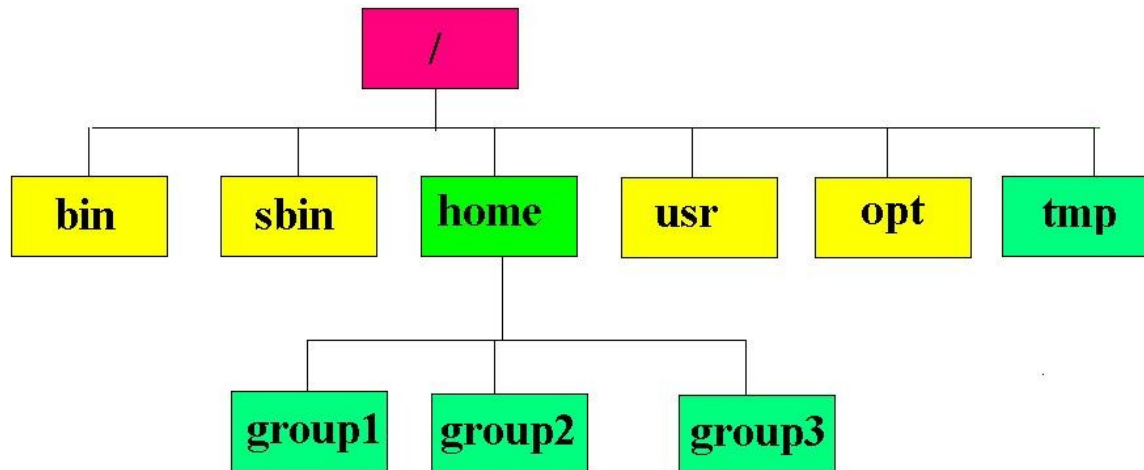


standard input/output



The File system

- Notice the similarities with other OS.
 - But no concept of drives, everything goes from the root directory
 - Look for the cdrom or the floppy under /mnt
 - the cdrom & floppy directories will be empty until they have been



Permissions

- All directories are special files
 - The read/write/execute scheme is used
 - User (owner), group and others

```
[group1@it001484 group1]$ ls -la jo*
-rwxr--r--  1 group1  group1    529 Feb  1 09:05 job.pbs.old
-rwxr--r--  1 group1  group1    526 Feb  8 13:19 job1.pbs
-rw-----  1 group1  group1    382 Feb  1 10:32 job1.pbs.o462
-rw-----  1 group1  group1    382 Feb  1 10:34 job1.pbs.o463
-rw-----  1 group1  group1    382 Feb  1 10:37 job1.pbs.o464
-rw-----  1 group1  group1    382 Feb  1 15:33 job1.pbs.o470
-rw-----  1 group1  group1    382 Feb  1 16:10 job1.pbs.o472
-rw-----  1 group1  group1    382 Feb  8 13:06 job1.pbs.o610
-rw-----  1 group1  group1    424 Feb  8 14:17 job1.pbs.o670
-rw-----  1 group1  group1    424 Feb  8 14:17 job1.pbs.o671

jobdir:
total 8
drwxrwxr-x  2 group1  group1   4096 Feb 12 15:39 .
drwx----- 15 group1  group1   4096 Feb 12 15:14 ..
```





4 - Commands



Command format

- *<command> <options> <arguments>*

- ***Examples:***

```
ls -la /home/group1
```

(this enables the user to see the file structure with dates etc, including hidden files, that is files that begin with a dot ‘.’)

```
who
```

(it shows who is logged in the same machine)

```
ps -ef
```

(it shows what processes are running specifically for this session)



Command format

- Pipes and filters:
- Pipes makes the output of the command to become the input for the next command
 - Filters read from input and write to standard output
 - command | filter



Command format

- *examples:*

```
ps -ef | grep group1
```

(this shows all processes related with userid group1)

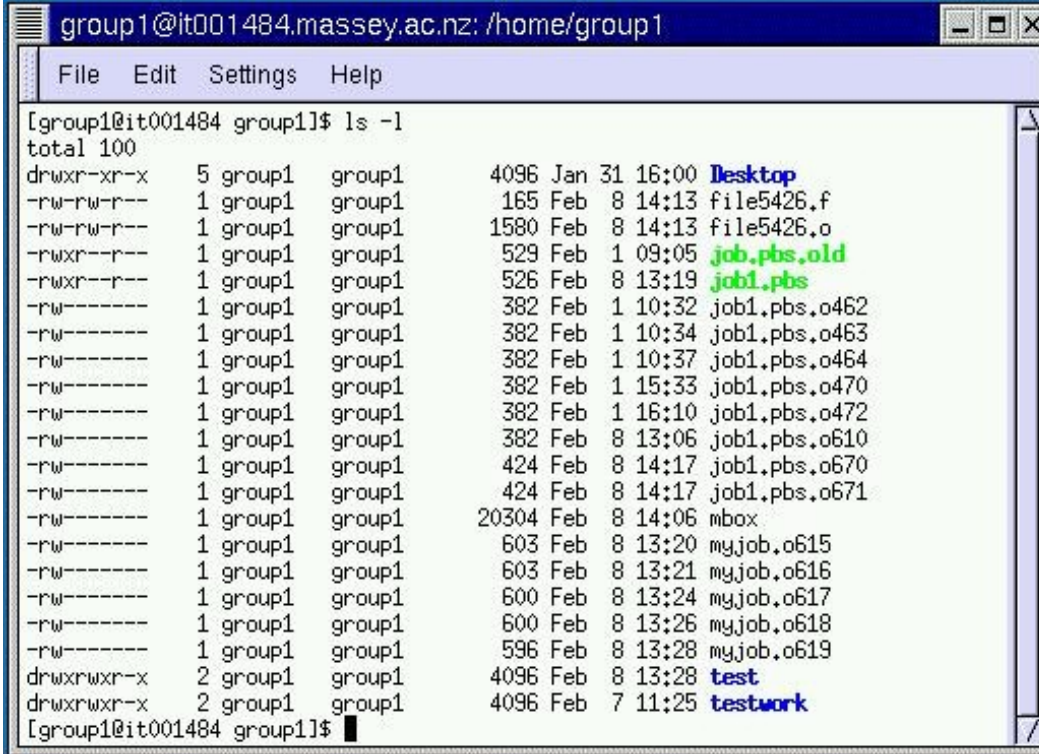
```
ps -ef | grep group1 | wc -l
```

(this will count the number of processes related with group1)



Basic Command...

- List files: `ls -la`



A terminal window titled "group1@it001484.massey.ac.nz: /home/group1" showing the output of the command `ls -l`. The window has a menu bar with "File", "Edit", "Settings", and "Help". The terminal output is as follows:

```
[group1@it001484 group1]$ ls -l
total 100
drwxr-xr-x  5 group1  group1    4096 Jan 31 16:00 desktop
-rw-rw-r--  1 group1  group1    165 Feb  8 14:13 file5426.f
-rw-rw-r--  1 group1  group1   1580 Feb  8 14:13 file5426.o
-rwxr--r--  1 group1  group1    529 Feb  1 09:05 job.pbs.old
-rwxr--r--  1 group1  group1    526 Feb  8 13:19 job1.pbs
-rw-----  1 group1  group1    382 Feb  1 10:32 job1.pbs.o462
-rw-----  1 group1  group1    382 Feb  1 10:34 job1.pbs.o463
-rw-----  1 group1  group1    382 Feb  1 10:37 job1.pbs.o464
-rw-----  1 group1  group1    382 Feb  1 15:33 job1.pbs.o470
-rw-----  1 group1  group1    382 Feb  1 16:10 job1.pbs.o472
-rw-----  1 group1  group1    382 Feb  8 13:06 job1.pbs.o610
-rw-----  1 group1  group1    424 Feb  8 14:17 job1.pbs.o670
-rw-----  1 group1  group1    424 Feb  8 14:17 job1.pbs.o671
-rw-----  1 group1  group1   20304 Feb  8 14:06 mbox
-rw-----  1 group1  group1    603 Feb  8 13:20 my.job.o615
-rw-----  1 group1  group1    603 Feb  8 13:21 my.job.o616
-rw-----  1 group1  group1    600 Feb  8 13:24 my.job.o617
-rw-----  1 group1  group1    600 Feb  8 13:26 my.job.o618
-rw-----  1 group1  group1    596 Feb  8 13:28 my.job.o619
drwxrwxr-x  2 group1  group1    4096 Feb  8 13:28 test
drwxrwxr-x  2 group1  group1    4096 Feb  7 11:25 testwork
[group1@it001484 group1]$
```



Four important tips:

- 1 - Double <tab>
 - Get file or command names with a few characters
- 2 - Arrows
 - Retrieve commands using ↑ or ↓
- 3 - ctrl-C
 - stop programs at once
- 4 `kill -9 <processID>`
 - to kill processes,
 - use `ps -e` to find the process ID number



File commands

`ls -la` shows all files

`mv <name1> <name2>` to rename a file,
move

`cp <name1> <name2>` **copy** a file
or **directory**



File commands (cont.)

<code>rm <name></code>	remove a file or dir
<code>rm -Rf <dir></code>	remove directory and all contents
<code>touch <name></code>	creates a zero size file or if already exists, updates the time stamp
<code>more</code>	shows the file
<code>less</code>	similar to <code>more</code>
<code>cat</code>	dumps the file to terminal



Directory commands

`cd` (**change directory**)

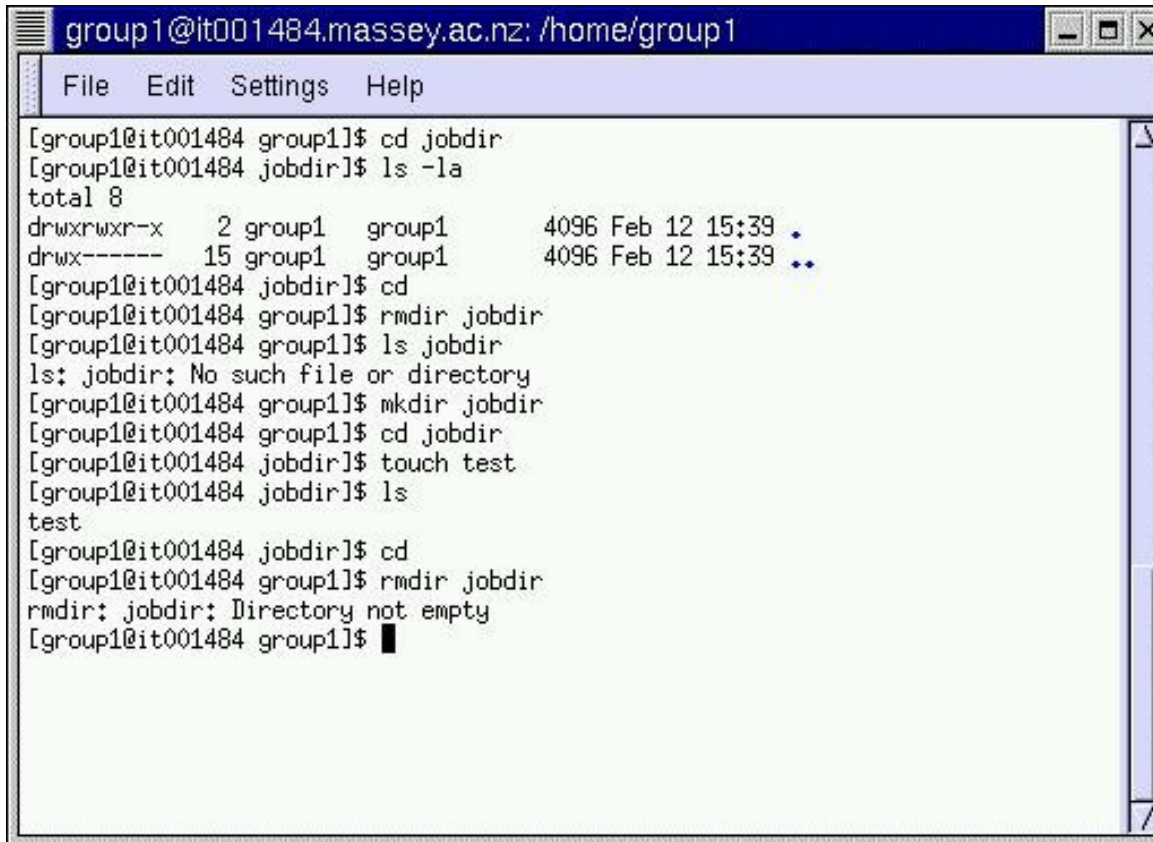
`mkdir` (**make directory**)

`rmdir` (**remove directory**)

`pwd` (which directory?)



cd, mkdir, rmdir



```
group1@it001484.massey.ac.nz: /home/group1
File Edit Settings Help
[group1@it001484 group1]$ cd jobdir
[group1@it001484 jobdir]$ ls -la
total 8
drwxrwxr-x  2 group1  group1   4096 Feb 12 15:39 .
drwx----- 15 group1  group1   4096 Feb 12 15:39 ..
[group1@it001484 jobdir]$ cd
[group1@it001484 group1]$ rmdir jobdir
[group1@it001484 group1]$ ls jobdir
ls: jobdir: No such file or directory
[group1@it001484 group1]$ mkdir jobdir
[group1@it001484 group1]$ cd jobdir
[group1@it001484 jobdir]$ touch test
[group1@it001484 jobdir]$ ls
test
[group1@it001484 jobdir]$ cd
[group1@it001484 group1]$ rmdir jobdir
rmdir: jobdir: Directory not empty
[group1@it001484 group1]$ █
```



Changing Permissions

- Use `chmod` command to change the permissions on a file
 - remove write permission for others

```
chmod o-w <file_name>
```
 - add read permission for user (owner) and group

```
chmod ug+r <file_name>
```
 - make executable for all users

```
chmod a+x <file_name>
```



Searching

- Finding files by name: ‘find’ command

```
find ~ -name "filename"
```

- looks for the file under the home directory (~)

```
find ~ -name "*.zip"
```

- Finding Text in file

```
fgrep -R "myText" .
```

- Looks for “myText” in all files and directories under the current directory (.)



Uploading and downloading files

- sftp is the recommended way to copy and retrieve files
 - linux users will already have a copy, windows users get psftp.exe
- run sftp or psftp
 - open a connection to the helix

```
open helix0
```
 - type in your user name and password



Using sftp

- Get files

```
get <file_name>
```

- Put files

```
put <file_name>
```

- getting and putting multiple files (not using psftp.exe)

```
mget <file_name_start>*
```

```
mput <file_name_start>*
```



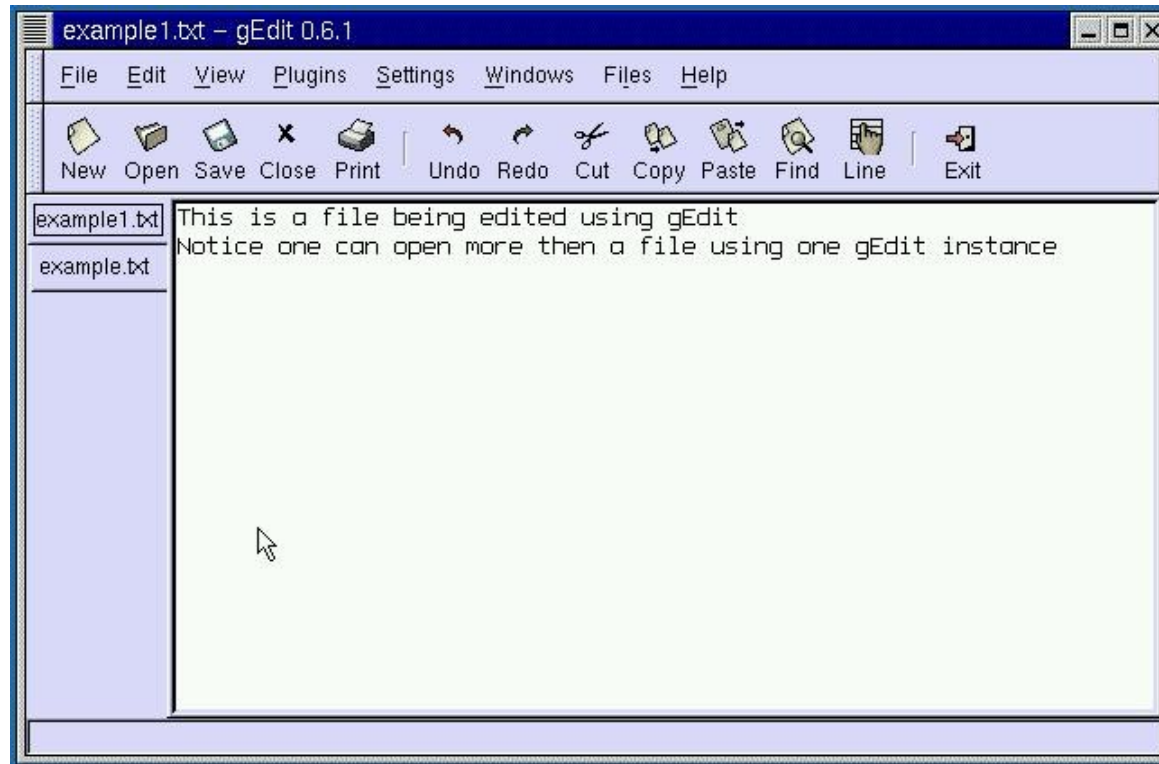


5 - Editing files



Graphical Editors

- There are several editors available. gedit is an example:



Other editors....

- gxedit
- vim, gvim
- emacs
- joe
- jed
- pico, vi, etc.....





6 - How to compile and run a test program



Compile and run (C/C++)

```
gcc <filename.c>
```

– Creates a.out by default

- To run type a.out

```
gcc <filename.c> -o <any-name>
```

– Creates ‘any-name’ executable

- To run type that name



Problems Running

- If your program can't be found then the current directory (.) might not be part of your path.
 - Instead of just `<program-name>` type
`./<program-name>`
- To see the current path
`echo $PATH`
- To change path (add current directory)
`export PATH=$PATH:.`



MPI C compiler

```
mpicc <filename.c>
```

- Creates a.out by default

```
mpicc <filename.c> -o <any-name>
```

- Creates ‘any-name’ executable

- To run use mpiexec with the -n option, number of processors required

```
mpirun -np 2 any-name
```



Batch Processing (Scalable)

↑ PBS options ↑ Redirect output

↑ # of nodes
in this job

```
#PBS -j oe -k eo -l nodes=4:ppn=2
#PBS -q default
cd $PBS_O_WORKDIR
mpirun -machinefile $PBS_NODEFILE -np 8 a.out
# start on 8 PEs
```

of processors
per node



Monitoring the PBS queue

- Submit your PBS job file

```
qsub <pbs_jobfile>
```

- Use the qstat command to see the jobs in the queue

```
qstat
```

- Use the xpbs command to graphically monitor the queue

```
xpbs &
```

- Use xpbsmon to see the state of the cluster

```
xpbsmon &
```

xpbs and xpbsmon

HOSTS

Server	Max	Tot	Que	Run	Hld	Mat	Trn	Ext	Status	PEsInUse	Select All	detail
helix0	0	8	0	8	0	0	0	0	Active	-/-		Submit..

QUEUES Listed By Host(s): helix0

Queue	Max	Tot	Ena	Str	Que	Run	Hld	Mat	Trn	Ext	Type	Server	Select All	detail
q2w1n	0	8	yes	yes	0	8	0	0	0	0	Execution	helix0		
q1w10n	0	0	yes	yes	0	0	0	0	0	0	Execution	helix0		
q1w2n	0	0	yes	yes	0	0	0	0	0	0	Execution	helix0		
q1w65n	0	0	yes	yes	0	0	0	0	0	0	Execution	helix0		
q1w32n	0	0	yes	yes	0	0	0	0	0	0	Execution	helix0		
q1w32p	0	0	yes	yes	0	0	0	0	0	0	Execution	helix0		

JOBS Listed By Queue(s): slow@nohost,OpenPBS.org

Other Criteria Select Jobs

Job id Select All

- detail
- modify..
- delete..
- hold..
- release..
- signal..
- msg..
- move..
- order

INFO

```
[01/22/03 11:40:31] '/opt/pbs/11b/xpbs/bin/xpbs_datadump -t 30 -u chmessom helix0'.....done.
```

Local

helix0

helix1	helix2	helix3	helix4	helix5	helix6	helix7	helix8	helix9	helix10	helix11	helix12
4	helix15	helix16	helix17	helix18	helix19	helix20	helix21	helix22	helix23	helix24	
6	helix27	helix28	helix30	helix31	helix32	helix33	helix34	helix35	helix36	helix37	
9	helix40	helix41	helix42	helix43	helix44	helix45	helix46	helix47	helix48	helix49	
1	helix52	helix53	helix54	helix55	helix56	helix57	helix58	helix59	helix60	helix61	
	helix62	helix63	helix64	helix65	helix0						

Nodes:: Total:65 Used:8 Avail:55 Down:1 Unk:1 VPROCsUsed:8

OFFL RSYD NOINFO INUSE/SHARED

walker.1nodes) INUSE(847.ngwalker.1nodes,842.ngwalker.1nodes) INUSE(845.ngwalker.1n

.....done.



7 - Finding help



'man' command

- 'man' stands for **manual**
- available in most Unix brands

```
man <command>
```



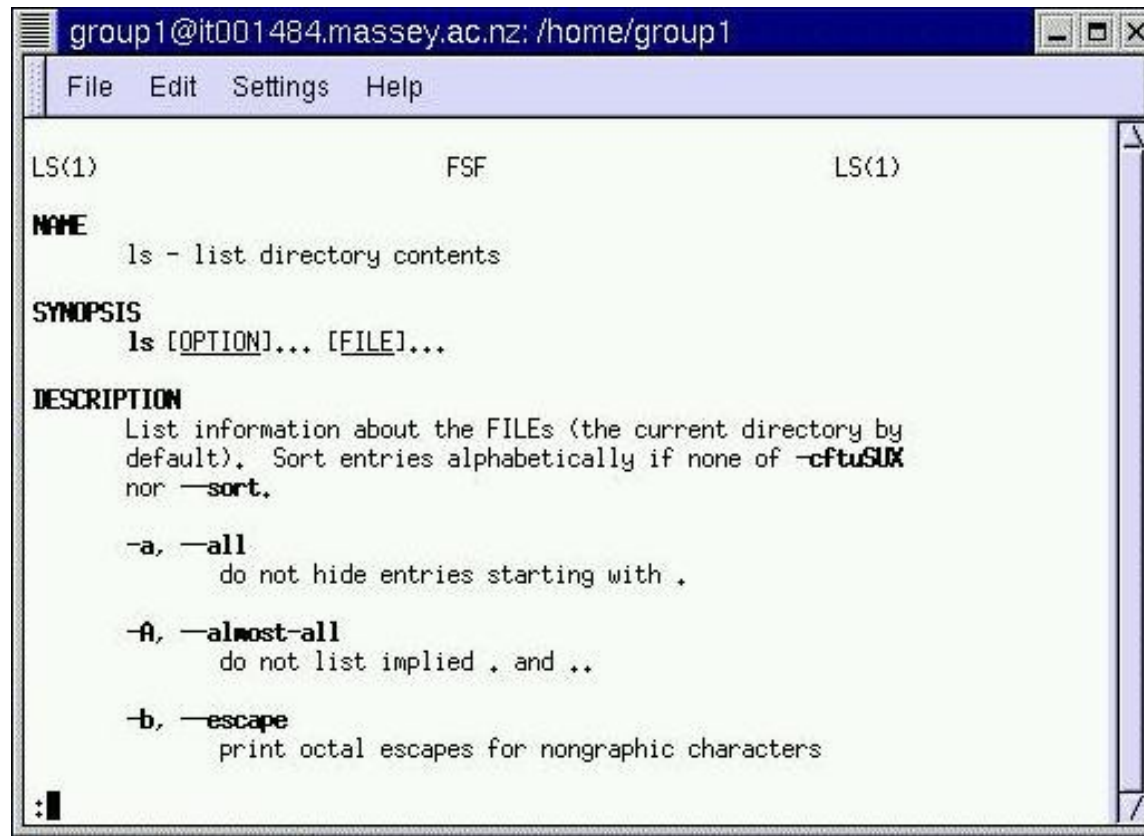
'man' command

- To scroll forward and backward:
 - **press [space] or [page down] to forward**
 - **press 'b' or [page up] to backward**
 - **press 'q' to quit**



man example:

man ls



```
group1@it001484.massey.ac.nz: /home/group1
File Edit Settings Help
LS(1) FSF LS(1)
NAME
  ls - list directory contents
SYNOPSIS
  ls [OPTION]... [FILE]...
DESCRIPTION
  List information about the FILEs (the current directory by
  default). Sort entries alphabetically if none of -cftuSUX
  nor --sort.
  -a, --all
    do not hide entries starting with .
  -A, --almost-all
    do not list implied . and ..
  -b, --escape
    print octal escapes for nongraphic characters
```



'info' command

- Slightly improved on 'man'
 - hierarchical 'menu' based system
- type `info` for a tutorial on info
 - in tutorial type `C-h` means `<control>-h`
 - in tutorial type `M-x` means `<Alt>-X`



Inside Info System

- l Quit this help.
 - q Quit Info altogether.
 - h Invoke the Info tutorial.
- Selecting other nodes:
- n Move to the "next" node of this node.
 - p Move to the "previous" node of this node.
 - u Move "up" from this node.
 - m Pick menu item specified by name. Picking a menu item causes another node to be selected.
 - f Follow a cross reference.



Other Documentation

– Available in html, postscript and pdf formats

- use `netscape` to view html
- `ghostview (gv)` to view ps files
- `xpdf` to view pdf files

– Compressed files

- use `unzip <filename>` to unzip .zip files
- use `tar xvf <filename>` to unzip .tar files
- use `tar xvzf <filename>` to unzip tar.gz or .tgz files
- use `gunzip <filename>` to unzip .gz or .Z files

Compressing Data

- Use tar to compress a directory

```
tar cvf <tar_filename> <directory>
```

- e.g.

```
tar cvf mywork.tar mywork
```

- To tar and gzip a directory

```
tar cvzf <tar.gz_filename> <directory>
```





Conclusions

- 1 - Introduction.
- 2 - Login, work, logout.
- 3 - Structure: file system, process.
- 4 - Commands.
- 5 - Editors.
- 6 - Compile and run a program.
- 7 - Finding help.

