# ACEnet Introduction

UNB - Bioinformatics

# Introduction to ACEnet

# What is ACEnet?

- Consortium of Atlantic Canadian universities
- Canada Foundation for Innovation (CFI) is the principal funding source
- High performance computing facilities
  - Hardware
    - Computing
    - Visualization
    - Communication
  - Software
  - Expertise

# Where is ACEnet?

Acadia University
University of Prince Edward Island
Mount Allison University
Memorial University
University of New Brunswick
University of New Brunswick, Saint John
St. Francis Xavier University
Cape Breton University
Dalhousie University
St. Mary's University

# Where is ACEnet?

| | |
|---|---|
| Acadia University | |
| University of Prince Edward Island | |
| Mount Allison University | |
| Memorial University | placentia |
| University of New Brunswick | fundy |
| University of New Brunswick, Saint John | |
| St. Francis Xavier University | |
| Cape Breton University | |
| Dalhousie University | glooscap |
| St. Mary's University | mahone |

# Available Hardware

Processors - 2.6GHz dual-core Opteron 285SE

| Cluster | Available Cores | RAM per Core | Node Connection |
|---|---|---|---|
| fundy | 636 | 4 or 8 | GigE<br>Infiniband |
| mahone | 532 | 4 | GigE<br>Myrinet 2000 |
| placentia | 3756 | 4 | GigE<br>Myrinet 10G |
| glooscap | 2196 | 1 | GigE |

# Cluster Layout

- Clusters are structured with a single node (the head node) controlling access
- A job submission system (N1GE or grid engine) controls access to compute nodes
- A single username/password provides access to all clusters
- File storage is not shared across the clusters

# Cluster Storage

| Cluster | /home/username | ./scratch | /scratch/tmp (on the nodes) |
|---|---|---|---|
| fundy | 47GB | 238GB | 38GB |
| mahone | 13GB | 238GB | 42GB |
| placentia | 13GB | 238GB | 31GB |
| glooscap | 13GB | 48GB | 13GB |

# NQS - No Quota Scratch

- Some jobs require even more storage
- NQS provides unlimited temporary storage
- No guarantee that files will remain there
- You can setup a warning system for when files will get deleted

| mahone | brasdor | placentia | glooscap | courtenay | fundy |
|---|---|---|---|---|---|
| 13TB | 12TB | 10TB | 2.7TB | 2TB | 13TB |

# Software

## http://www.ace-net.ca/wiki/Software

| Scientific Computing | Scientific Libraries | Parallel APIs | Compilers and Languages | Debuggers and Profilers |
|---|---|---|---|---|
| Star Lab | FFTW | OpenMPI | C/C++ | gdb |
| Octave | ACML | CRE | FORTRAN 77/90 | dbx |
| Gaussian | LAPACK | openMP | Java | Sun Studio |
| Gromacs | GSL | pyMPI | Perl,Python | TotalView |

# Other Facilities

- Internet Videoconferencing (inSORS/AccessGrid)
- Located at
  - Dalhousie University
  - Memorial University
  - Mount Allison University
  - Saint Mary's University
  - St. Francis Xavier University
  - Acadia University
  - University of New Brunswick
- 3D data cave - available at St. Mary's

# How do I get started?

Can get an account at

- http://www.ace-net.ca/wiki/Get_an_Account
- http://ccdb.computecanada.ca

Account types

- Primary Project Account - Primary Project Accounts are held by the Research Project Leader who is typically the principal investigator for a project or the head of a research group. A project leader is responsible for all other user accounts associated with the project.
- User Account - User accounts will only be issued on the authority of the Research Project Leader for a particular project

# How do I connect?

All connections are over secure links

- Logging in
  - ssh - Secure SHell
- Moving data around
  - scp - Secure CoPy
  - sftp - Secure File Transfer Protocol

All tools are available under Linux and Mac OSX by default.
Windows users need to download extra tools

- ssh - putty
- scp - WinSCP

# Taking Turns - the job queue

- Login nodes are for file management, compiling, testing - small stuff
- "Production" work is distributed using queue management software - Sun Grid Engine (N1GE)
- Relevant commands - qsub, qstat, qacct, qdel, qalter
- More information available through
  - wiki - http://wiki.ace-net.ca
  - man pages - man qsub

# Types of jobs

Series vs. Parallel

- Multiple serial jobs can be run in parallel
- Parallel jobs can be run in multiple configurations
- The aim is to complete a calculation in the shortest possible time
- Should all existing code be made parallel? Is it possible?

# Where to go for help

ACEnet User Guide

- http://www.ace-net.ca/wiki/User_Guide

ACEnet User Wiki

- http://www.ace-net.ca

Research / User support personnel

- email: support@ace-net.ca

# Introduction to Linux and Shell Scripting

# What is Linux?

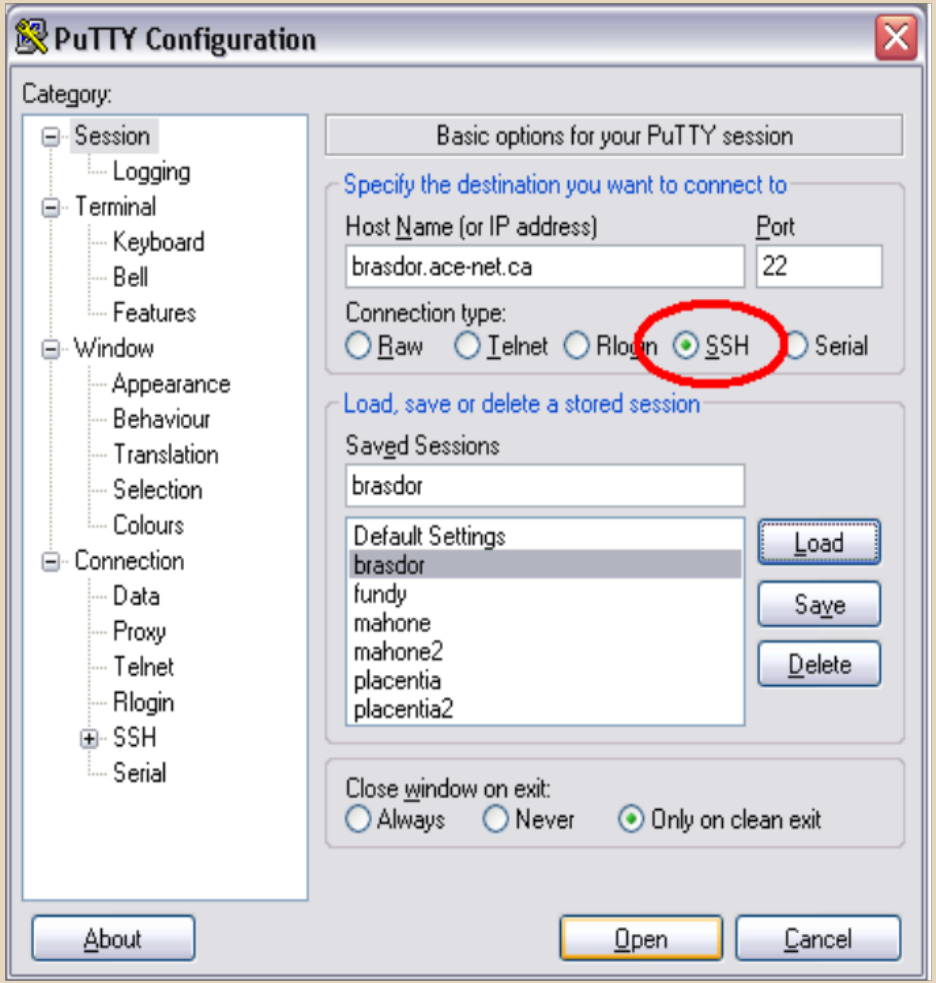
# Getting Started

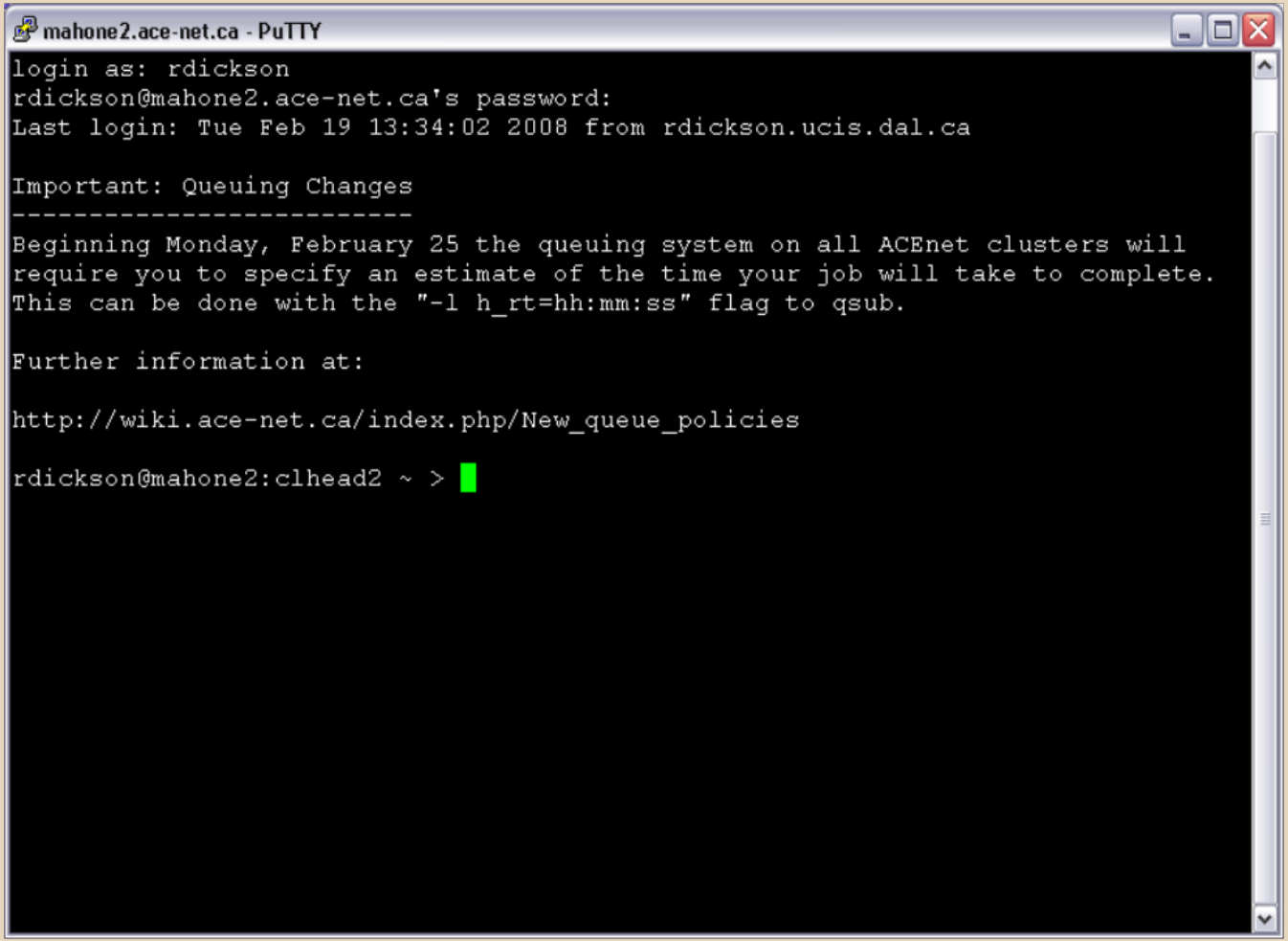
- How to connect
  - ssh
  - sftp
- Basic Unix or Linux
  - commands
  - files and directories
  - environment variables

*...not necessarily in that order*

# Getting connected

- ssh *Secure SHell*
  - provided with Mac OS X and Linux
    - **ssh -Y *username@somehost.*ace-net.ca**
  - Windows users need PuTTY or Cygwin or MobaXterm
    - http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html
    - http://www.cygwin.com
    - http://mobaxterm.mobatek.net/


PuTTY Configuration
Category:
Session
Logging
Terminal
Keyboard
Bell
Features
Window
Appearance
Behaviour
Translation
Selection
Colours
Connection
Data
Proxy
Telnet
Rlogin
SSH
Serial
Basic options for your PuTTY session
Specify the destination you want to connect to
Host Name (or IP address)
Port
brasdor.ace-net.ca
22
Connection type:
Raw
Telnet
Rlogin
SSH
Serial
Load, save or delete a stored session
Saved Sessions
brasdor
Default Settings
brasdor
fundy
mahone
mahone2
placentia
placentia2
Load
Save
Delete
Close window on exit:
Always
Never
Only on clean exit
About
Open
Cancel


mahone2.ace-net.ca - PuTTY
login as: rdickson
rdickson@mahone2.ace-net.ca's password:
Last login: Tue Feb 19 13:34:02 2008 from rdickson.ucis.dal.ca
Important: Queuing Changes
Beginning Monday, February 25 the queuing system on all ACEnet clusters will require you to specify an estimate of the time your job will take to complete. This can be done with the "-l h_rt=hh:mm:ss" flag to qsub.
Further information at:
http://wiki.ace-net.ca/index.php/New_queue_policies
rdickson@mahone2:clhead2 ~ >

# Looking Around

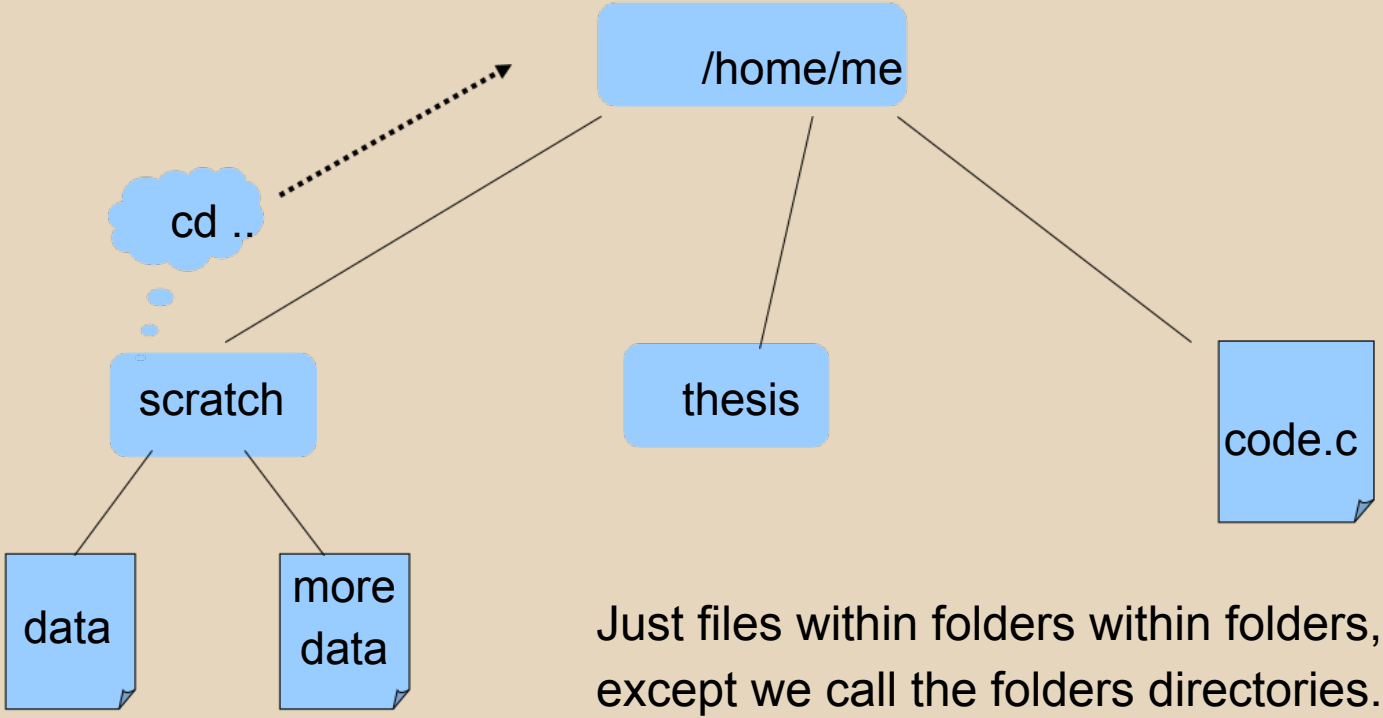
- pwd
  - “print working directory”
  - where am I?
- ls
  - “list”
  - what files are here? what subdirectories?

# Getting around

- mkdir *newdir*
  - “make directory”
- cd *directory*
  - “change directory”
- cd ..
  - “..” means “parent directory” or “one level up”
  - “.” means “right here”
  - “~” means “my home directory”

# Directories grow on trees

*Ok, upside down trees...*




Just files within folders within folders,
except we call the folders directories.

# Example

- > mkdir workshop
- > cd workshop
- > ls
- > pwd
- /home/*userid*/workshop
- >

# Moving stuff around

- mv *from to*
  - “move”, also “rename”
- cp *from to*
  - “copy”
- rm *file*
  - “remove”
  - Careful --- There's no recycle bin!
- rmdir *directory*

# Paths, directories, abbreviations

- directory separator is “/”
- /home/jsmith/code/test is an *absolute path*
  - begins with “/”
- code/test is a *relative path*
  - no “/” at the beginning
  - same thing if pwd is /home/jsmith
- *
  - “wildcard”

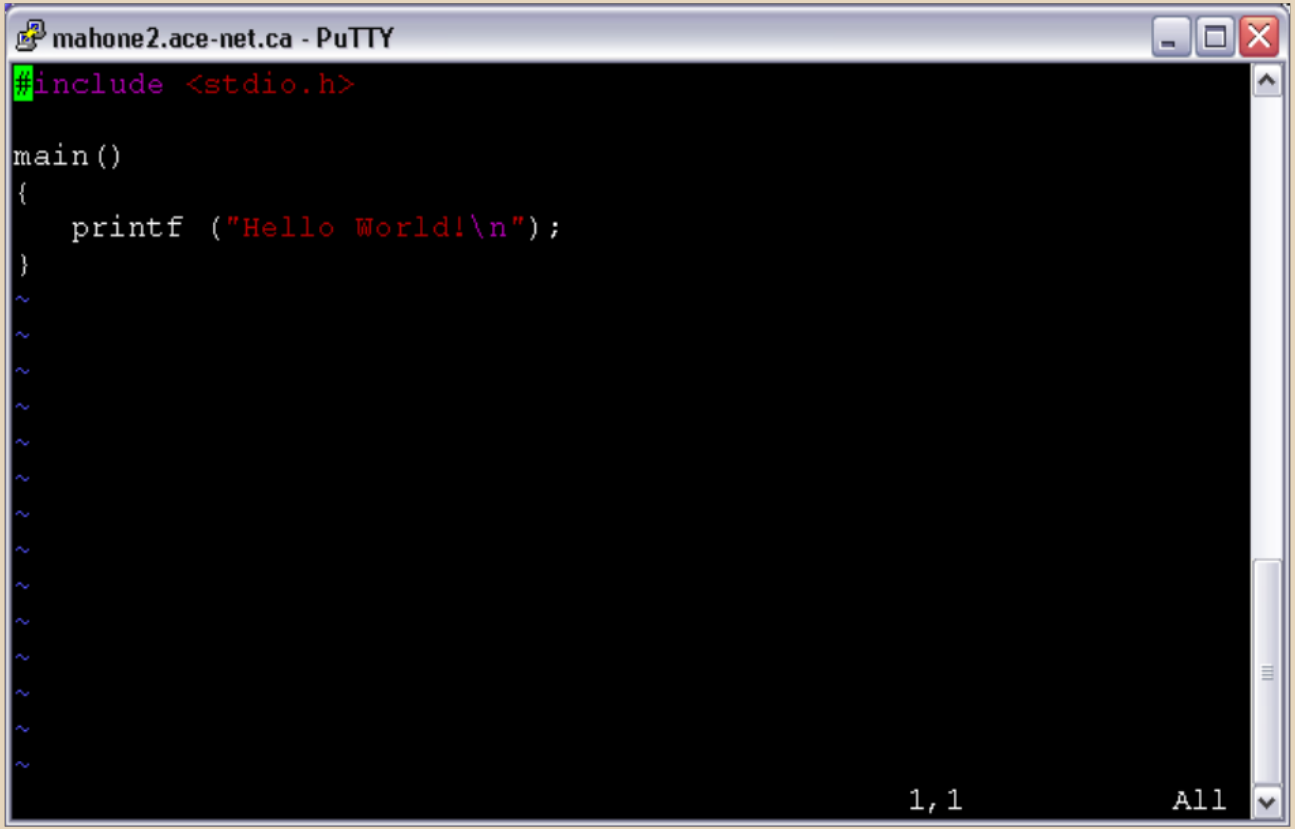
# Example

- > pwd
- /home/*userid*/workshop
- > cp /home/rdickson/demo/C/* .
- > ls
- function.c hello.c input int1.c
- int2.c int3.c int4.c Makefile
- >

# What's inside?

- more *file*
- less *file*
  - scroll through file, one screen at a time
  - Spacebar to continue, q to quit, ctrl-B to back up
- Editors
  - **vim *file***
  - **emacs *file***
  - or use your computer and transfer with sftp

# vim


mahone2.ace-net.ca - PuTTY
#include <stdio.h>
main()
{
printf ("Hello World!\n");
}
1,1
All

# File Permissions

Sometimes you need to edit the permissions of a file. The permissions of a file dictate who can execute, read and write to a file.

If you do **ls - l** you might see something like the following:

**-rwxr-xr-x 1 staveley TechTeam 271 Jul 12 2007 Makefile**

- To see a directory listing you must have *execute* permission
- To change permissions, use **chmod**
  - **chmod +x *file*** to turn on execute permission for you
  - **chmod o-r *file*** to turn off read permisson for others

# What happens if I ...?

- which *program*
  - shows absolute path of *program*
- echo $PATH
  - what directories are searched for executables?
  - $PATH is an *environment variable*
- setenv or set
  - “set environment”
  - shows or sets variables

# Miscellany

More information can be found out about the system you are working on with the following commands:

**du** - disk usage - how much disk space is being used by the current directory

**du -h** - “human readable”

**df** - how much disk space is available

**ps** - show the programs that you are running

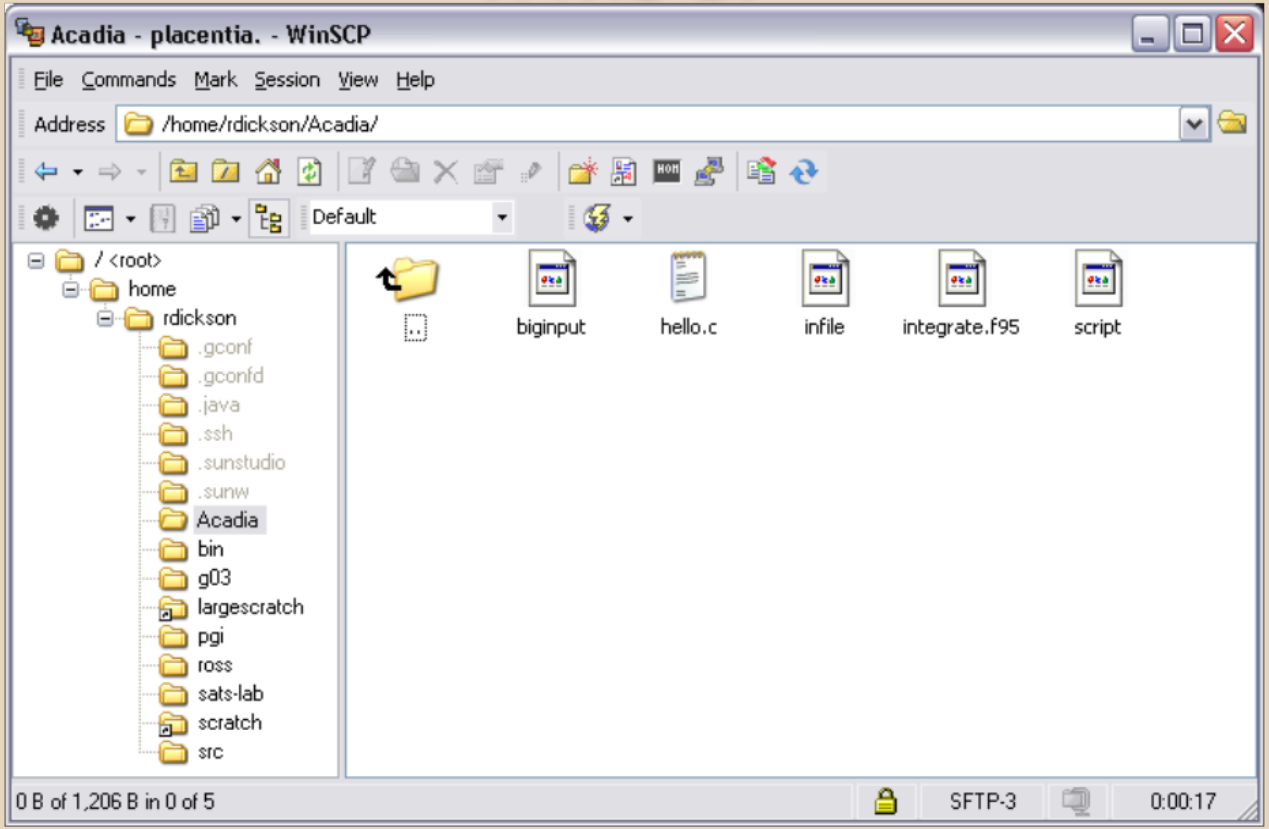
**top** - shows information about processor and memory usage of the machine you’re on

**date** - displays the current date and time (machine clock)

# Data from the outside world
## or from one cluster to another

- sftp *Secure File Transfer Protocol*
  - provided with Mac OS X and Linux
    - **sftp username@somehost.ace-net.ca**
    - Easiest way to move data between clusters
  - MS Windows? Use WinSCP or Cygwin
    - http://winscp.net/eng/index.php
    - http://www.cygwin.com

# WinSCP


Acadia - placentia. - WinSCP
File Commands Mark Session View Help
Address /home/rdickson/Acadia/
Default
/ <root>
home
rdickson
.gconf
.gconfd
.java
.ssh
.sunstudio
.sunw
Acadia
bin
g03
largescratch
pgi
ross
sats-lab
scratch
src
biginput
hello.c
infile
integrate.f95
script
0 B of 1,206 B in 0 of 5
SFTP-3
0:00:17

# sftp commands

- ls
  - List files already *there*
- put *filename*
  - Copy file from *here* to *there*
  - Can use wildcards: “put *” copies everything
- get *filename*
- cd *directory*
  - Change directory *there*
- lcd *directory*
  - “Local cd”, change directory *here*

# Where to go for help

- man *command*
  - “manual”
- man -k *keyword*
  - searches online manual pages
- Online Linux tutorials
  - Google “Unix command line tutorial”

# Where else to go for help

- ACEnet User Wiki
  - http://www.ace-net.ca/
- User support line
  - email: support@ace-net.ca


brasdor.ace-net.ca - PuTTY
login as: rdickson
Using keyboard-interactive authentication.
Password:
Last login: Fri Jun 8 12:14:12 2007 from n2.pub.hpcvl.qu
Sun Microsystems Inc. SunOS 5.10 Generic January 2005
rdickson@brasdor: ~ >

# What is the shell?

- The shell provides the interface to the system
- It is not a part of the kernel
  - The kernel only provides access to the hardware
- This means you can choose how you interact with your computer
- The default on most systems is ‘bash’

# Shell Scripts

- Scripts are text files with lists of commands
- How to run them?
- Run explicitly

  /bin/bash my_script.sh
- Run directly - you need to make it executable

  ./my_script.sh
- If you want to run it directly, be sure to add

  #!/bin/bash

# First Script

```
echo "The date and time today is" \ `date` \
echo
who
echo
echo "The number of login windows is" $(who | wc -l)
echo
echo "What is my user name:" $(who am i) "and where am i:" $(pwd)
```

# Output

```
The date and time today is  Thu Oct 9 12:40:07 ADT 2008

srazul   pts/0        Oct  9 11:24 (141.109.52.35)
evorobyo pts/1        Oct  9 07:34 (76.11.66.169)
sgupta   pts/2        Oct  9 09:12 (puffin.bioc.mun.ca)
mcoates  pts/3        Sep 10 12:23 (134.153.141.136)
ghuang   pts/4        Oct  9 11:43 (141.109.36.40)
aelsherb pts/5        Sep 29 16:58 (doc.chem.mun.ca)
jzhu     pts/6        Oct  8 15:46 (coriolis.physics.mun.ca)
ishort   pts/7        Oct  8 09:12 (crux.smu.ca)
rkarsten pts/8        Sep 30 12:28 (mathfluids.acadiau.ca)
ghuang   pts/9        Oct  9 11:47 (141.109.36.40)
jchaffey pts/10       Oct  9 11:10 (0xtp.mar.dfo-mpo.gc.ca)
pyuet    pts/11       Oct  9 11:20 (landau-four-forty-six.mit.edu)
jbernard pts/12       Oct  9 11:25 (bernardmac.cs.unb.ca)
aelsherb pts/13       Oct  6 14:08 (doc.chem.mun.ca)
rdickson pts/14       Oct  9 11:57 (df01a.vpn.dal.ca)
yhuang   pts/16       Oct  9 12:02 (141.109.36.42)
jramsey  pts/23       Oct  8 11:51 (ttauri.smu.ca)
jzhu     pts/26       Oct  8 15:53 (coriolis.physics.mun.ca)

The number of login windows is 18

What is my user name: srazul pts/0 Oct 9 11:24 (141.109.52.35) and where am i: /home/srazul
```

# Cutting Text

test_cut.txt -> a#b#c#d#e#f#g#h#i#j#

cut -1 test_cut.txt -> a

cut -c1-5 test_cut.txt -> a#b#c

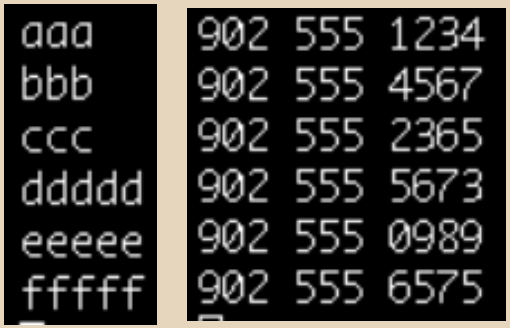
cut -c1-5,15 test_cut.txt -> a#b#ch

cut -d# -f2 test_cut.txt -> b

# Pasting it Back Together

paste test1 test2 > new

```
aaa     902 555 1234
bbb     902 555 4567
ccc     902 555 2365
ddddd   902 555 5673
eeeee   902 555 0989
fffff   902 555 6575
```




paste -d '%' test1 test2 > new2

```
aaa%902 555 1234
bbb%902 555 4567
ccc%902 555 2365
ddddd%902 555 5673
eeeee%902 555 0989
fffff%902 555 6575
```

# Searches

grep -v ‘d’ paste1

```
aaa
bbb
ccc
eeeee
fffff
```

grep -n ‘d’ paste1

```
4:ddddd
```

# Grep Patterns

PATTERNS MATCHED STRINGS

a.c a[any single character]c, e.g. abc, a1c, a c.

a[xyz]c or a[x-z]c axc, ayc, and azc only

a[^xyz]c a[any single character but x, y, or z]c

ab*c a[0 or more b]c, e.g. ac, abc, abbbc

ab+c a[1 or more b]c, e.g. abc, abbbc

^abc abc only at the beginning of a line

abc$ abc only at the end of a line

a(bc|de)f abcf and adef

# Sed - Stream Editing

sed ‘s/9/*/g’ paste2

```
*02 555 1234
*02 555 4567
*02 555 2365
*02 555 5673
*02 555 0*8*
*02 555 6575
```

sed ‘5d’ paste2

```
902 555 1234
902 555 4567
902 555 2365
902 555 5673
902 555 6575
```

sed ‘/3/d’ paste2

```
902 555 4567
902 555 0989
902 555 6575
```

# Sed - 2

sed ‘s/^/hello /g’ paste2

```
hello 902 555 1234
hello 902 555 4567
hello 902 555 2365
hello 902 555 5673
hello 902 555 0989
hello 902 555 6575
```

sed ‘s/$/ hello/g’ paste2

```
902 555 1234 hello
902 555 4567 hello
902 555 2365 hello
902 555 5673 hello
902 555 0989 hello
902 555 6575 hello
```

# If Statements

if

  …

then

  …

else

  …

fi

```
user="$1"

if who | grep "^$user " > /dev/null
then
     echo "$user is logged ON"
else
     echo "$user is not logged on"
fi
```

# Tests

Operator Returns TRUE (exit status of 0) if

-d *file ---------> file is a directory*

-e *file ---------> file exits*

-f *file ---------> file is an ordinary file*

-r *file ---------> file is readable by the process*

-s *file ---------> file has nonzero length*

-w *file ---------> file is writable by the process*

-x *file ---------> file is executable*

-L *file ---------> file is a symbolic link*

# Tests - 2

```
file=$1
if [ -f $1 ]
then
     echo "yes '$1' is here"
else
     echo "No '$1' is not here"
fi
```

# Greeting Examples

```
hour=$(date | cut -c12-13)
echo $hour
if [ "$hour" -ge 0 -a "$hour" -le 11 ]
then
        echo "Good Morning"
elif [ "$hour" -ge 12 -a "$hour" -le 17 ]
then
      echo "Good Afternoon"
else
      echo "Good evening"
fi
```

# Comparison Operations

What other's are there?

-gt - greater than

-lt - less than

-eq - equal to

-ne - not equal to

# Chaining Commands

Commands can be chained together, based on whether they completed successfully or not

&&

||

```
user=$1
hour=$(date | cut -c12-13)
echo $hour
#
if [ "$hour" -ge 0 -a "$hour" -le 17 ] &&  [ "$user" =  srazul ]
then
        echo "Good Day"
else
       :
fi
```

# Arithmetic

bash can do basic integer arithmetic

$((.....))

| | |
|---|---|
| + | addition |
| - | subtraction |
| * | multiplication |
| / | integer division |
| % | modulus |

# While Loop

while

do

....

done

```
#
i=1
while [ "$i" -le 5 ]
do
        echo $i
        i=$((i+1))
done
#
```

# Until Loop

until

do

....

done

```
#
i=1
until [ "$i" -eq 6 ]
do
      echo $i
      i=$((i+1))
done
#
```

# Bioinformatics

# Available Software

ABySS
BIMr
BioPerl
BLAST
Bowtie
Cufflinks
IM
LAMARC
RAPSearch
Migrate
MIRA
MrBayes
PhyML
PLINK
SAMtools
Structure

# BLAST Examples

```
#$ -cwd
#$ -l h_rt=01:00:00
module load blast
blastx -query query.FASTA -db $BLASTDB/nr -out ${JOB_ID}_results.txt -evalue 1E-10 -outfmt 7 -num_alignments 10
```

```
#$ -cwd
#$ -l h_rt=01:00:00
#$ -pe openmp 4

module load blast
blastx -query query.FASTA -db $BLASTDB/nr -out ${JOB_ID}_results.txt -evalue 1E-10 -outfmt 7 -num_alignments 10 -num_threads $NSLOTS
```

# Bowtie Example

```
#$ -cwd
#$ -l h_rt=01:00:00
#$ -pe openmp 4

module load bowtie
bowtie2 -p $NSLOTS -x lambda_virus -U \ $BOWTIE/example/reads/reads_1.fq -S eg1.
sam
```