

HPRC Cluster: Quick Start User Guide

THIS PAGE CONTAINS OLD INFORMATION - HPC staff will work on updating it soon.

This page is intended as a quick introduction for new users submitting their first job to the HPRC Cluster.

In the typical work flow the user:

1. Logs into zodiac.hpc.jcu.edu.au
2. Prepares a submission script for their jobs
3. Submits their jobs to the Job Scheduler
4. Monitors their jobs
5. Collects the output of their jobs.

A few things that new users should be aware of:

- Typically, jobs are not run in an interactive manner, except when:
 - users are running small one off jobs
 - evaluating the resources required for bigger jobs
 - using graphical applications like MATLAB
- Examples of interactive jobs:

Content by label

There is no content with the specified labels

- HPRC Cluster software is not run in a window on their desktop, neither is it launched by clicking on it in a network drive.
- Users need to [log into the cluster](#) and inform the job scheduler about their job and it will run it when it can.



Quick Start

- [Logging In](#)
- [Software Packages](#)
- [Running Jobs](#)
 - [Example: paup with the ML_analysis.nex sample file](#)
 - [Creating the job script](#)
 - [Submitting the Job - qsub](#)
 - [Monitoring the Job - qstat](#)
 - [Deleting a job - qdel](#)
 - [Your job's Output](#)
 - [Other Examples](#)
- [Job Resources](#)
- [Further Reading](#)

Logging In

The first step in using the HPRC Cluster is to log in to a login node - `ssh1.hpc.jcu.edu.au` or `ssh2.hpc.jcu.edu.au` (zodiac.hpc.jcu.edu.au for people still wanting to access the old cluster).

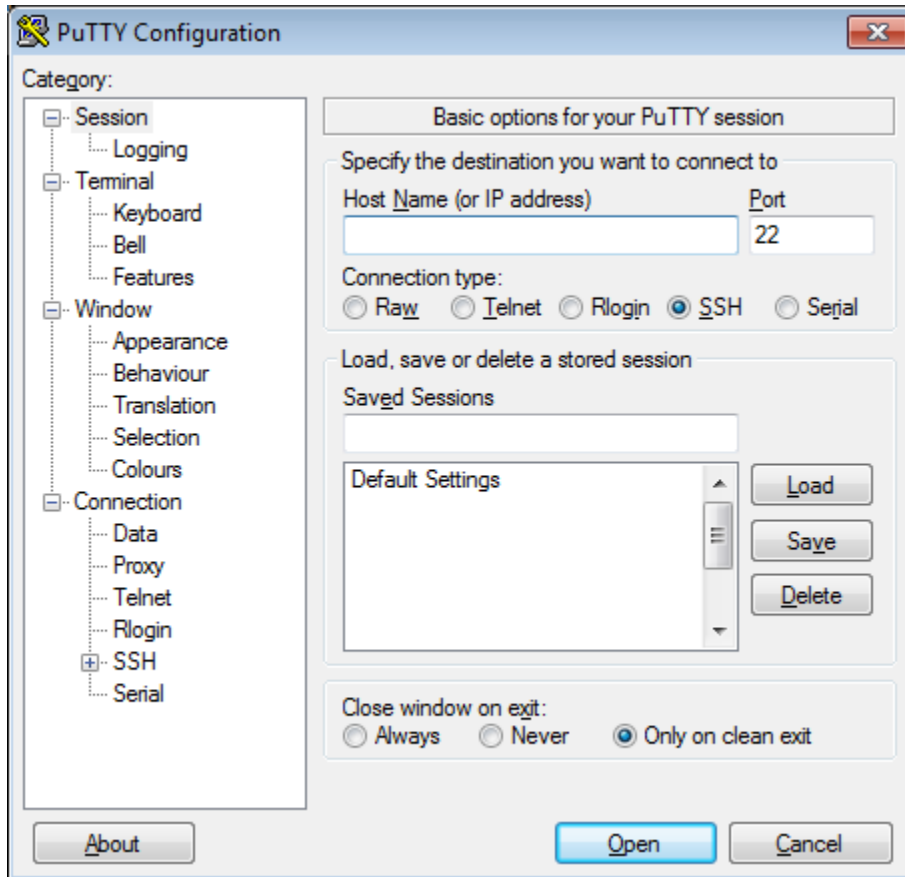
HPRC Desktop Software - Logging into zodiac.hpc.jcu.edu.au

The HPC interactive nodes are accessible via the server named `zodiac.hpc.jcu.edu.au`. See the relevant tabs below of instructions on how to log in to `zodiac`. `Zodiac` is a linux based system. To learn more about the linux shell, see the [Software Carpentry Unix Shell tutorials](#).

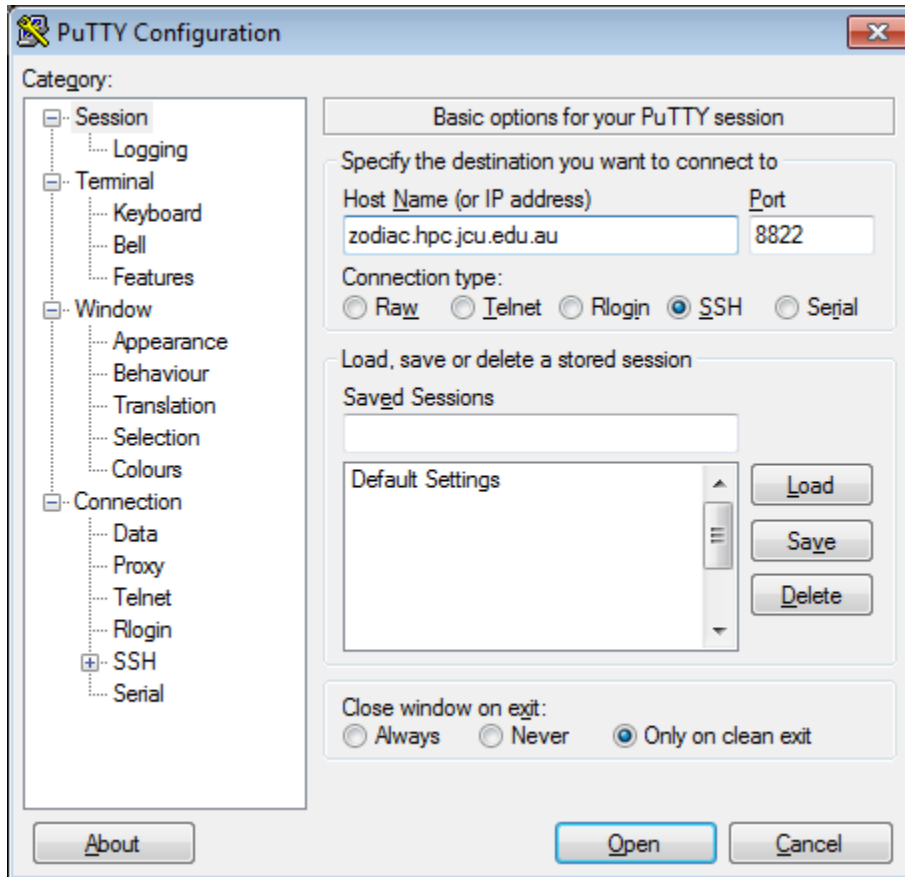
Windows

1. Unable to render {include} The included page could not be found.

2. Starting PuTTY will show this window:

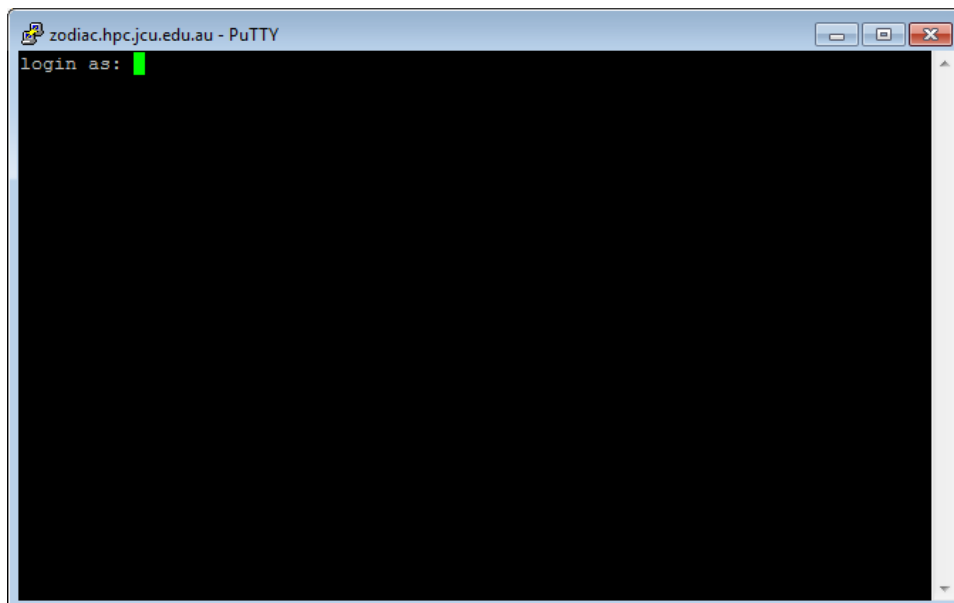


3. Enter the hostname (zodiac.hpc.jcu.edu.au) and the port 8822

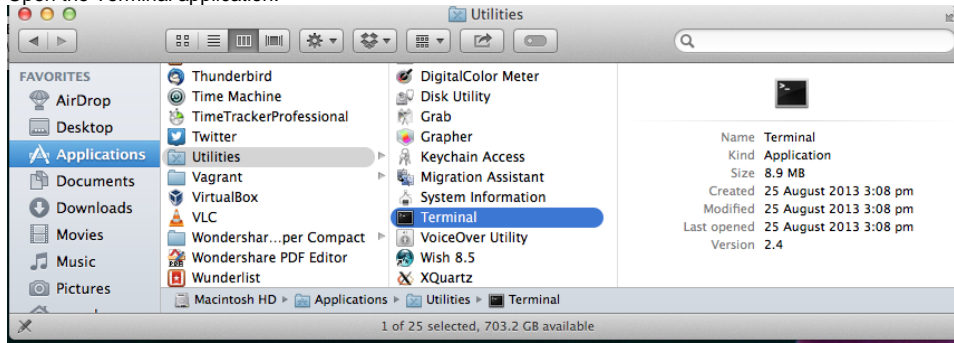


The default port for ssh is port 22, which you can use if you are accessing the cluster from on campus, however if you are accessing it from off campus you need to use port 8822.

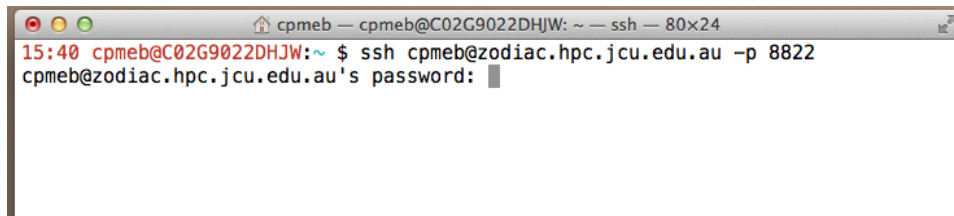
4. Then you will be prompted for your username and password, use your standard JCU credentials (username and password)



1. Open the Terminal application:



2. In the Terminal window run the ssh command: `ssh <username>@zodiac.jcu.edu.au` (add `-p 8822` if you are connecting from outside the JCU network) and you will be asked for your password



3. You are now logged in to the HPC interactive node.

Software Packages

The HPRC Cluster uses environment modules to manage the available software packages. This allows multiple versions of the same software to be installed without interfering with each other. To enable the `environment` module system the following command needs to be executed on the command line:

```
-bash-4.1$ source /etc/profile.d/modules.sh
```

The software that is available on the HPRC cluster is listed here: [HPRC User Software](#). Alternately you can query the software available on the cluster with the following commands:

Command	Result
<code>module avail</code>	A list of available software is displayed
<code>module help <software></code>	Version number and brief synopsis is displayed for <software>

```

-bash-4.1$ module avail
----- /usr/share/Modules
/modulefiles -----
MPInside/3.5.1    compiler/gcc-4.4.5 module-cvs    modules    mpich2-x86_64
null            perfcatcher
chkfeature      dot                module-info  mpi/intel-4.0  mpt/2.05
perfboost       use.own
----- /etc
/modulefiles -----
compat-openmpi-x86_64 openmpi-x86_64
----- /sw
/modules -----
4ti2            blast/2.2.23      crimap_Monsanto    hdf5
migrate/3.6(default)  picard-tools      tmap/1.1
BEDTools       blast/2.2.29(default)  dx                hmmer
mira           proj              tmhmm
EMBOSS         bowtie            elph              ima2
modeltest      pvm               topali
GMT            bwa/0.7.4(default)  enmtools          jags
molphy         r8s               towhee
Macaulay2     caftools          fasta              java
mpich2         rainbowcrack      towhee-openmpi
Python/2.7     cap3              fastme             jcusmart
mrbayes        rpfits            trans-abyss
R/2.15.1(default)  carthagene/1.2.2(default)  ffmpeg           jmodeltest
mrmodeltest    ruby/1.9.3         tree-puzzle
R/3.0.0        carthagene/1.3.beta  fftw2             lagan
msbayes        ruby/2.0.0         trinityrnaseq
abyss          casacore           fftw3             lamarc
ncar           samtools          udunits
ariadne        cernlib           garli              lapack
netcdf         scalapack         udunits2
arlequin       cfitsio           gdal               libyaml/0.1.4
netphos        scipy             velvet
asap           chlorop           glimmer            matlab/2008b
numpy          seadas/6.2        wcslib
atlas          clipper           glpk               matlab/2012a
oases          seg               wise2
bayesass       clustalw          gmp                matlab/2012b
octave         signalp           wwatch3
beagle         cluster           gnu/4.1.2          matlab/2013a(default)
openbugs       sprng             yasm
beast          cns               gnu/4.4.0          maxent
openjdk        ssaha2            zonation
beast-1.5.4    coils            gnuplot            maxima
openmpi        stacks
bfast          colony2           grass              merlin
pari           structure
blacs          consel            gromacs            migrate/3.2.15
paup           targetp
blas           crimap            hdf                migrate/3.5.1
phyml          tcldreadline/2.1.0

```

Running Jobs

To run a job on the cluster create a shell script containing the PBS Directives containing the information required by the scheduler to schedule the job, and the job commands.

Example: [paup](#) with the [ML_analysis.nex](#) sample file

Running a job on the HPRC Cluster

In this example we will run paup with the [ML_analysis.nex](#) sample file provided on the [paup sample nexus files](#) page.

After logging into the cluster download the example file with the command:

```
-bash-4.1$ wget http://paup.csit.fsu.edu/data/ML_analysis.nex
--2014-03-11 13:08:16-- http://paup.csit.fsu.edu/data/ML_analysis.nex
Resolving paup.csit.fsu.edu... 144.174.50.3
Connecting to paup.csit.fsu.edu|144.174.50.3|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 2990 (2.9K) [text/plain]
Saving to: "ML_analysis.nex"

100%[=====] 2,990 --.-K/s in 0s

2014-03-11 13:08:17 (70.7 MB/s) - "ML_analysis.nex" saved [2990/2990]
```

Creating the job script

Using a text editor – examples include vim and nano – create your shell script with the filename: `ML_analysis.sh` and the following contents (the colours are only used for illustration purposes below):

```
#!/bin/bash

#
# Checkpointing is to be done on a job at pbs_mom shutdown.
#
#PBS -c s

#
# Merge standard output and standard error streams into the named file.
#
#PBS -j oe

#
# Set the name of the job
#
#PBS -N ML_analysis

#
# Advise the scheduler that the job requires one cpu (ppn) from one node.
#
#PBS -l nodes=1:ppn=1
```

```

#
# Advise the scheduler about the amount of physical memory required.
# kb for kilobytes, mb for megabytes, gb for gigabytes.
#
#PBS -l pmem=5gb

#
# Advise the scheduler that this job will have completed within 10 minutes.
#
#PBS -l walltime=00:10:00

#
# Send mail at batch job abort/exit to the Email address provided.
#
#PBS -m ae
#PBS -M your.name@jcu.edu.au

ncpu=`wc -l $PBS_NODEFILE | awk '{print $1}'`
echo "-----"
echo " This job is allocated "$ncpu" CPU cores on "
cat $PBS_NODEFILE | uniq
echo "-----"
echo "PBS: Submitted to $PBS_QUEUE@$PBS_O_HOST"
echo "PBS: Working directory is $PBS_O_WORKDIR"
echo "PBS: Job identifier is $PBS_JOBID"
echo "PBS: Job name is $PBS_JOBNAME"
echo "-----"

cd $PBS_O_WORKDIR
source /etc/profile.d/modules.sh
module load paup
paup -n ML_analysis.nex

```

Legend:

1. The very first line of the script file is the **Shebang**, the shebang must always be the first line.
2. The second section contains the PBS directives. For more information on PBS directives please see the [HPC PBSPro script files page](#).
3. The third section outputs information about the job, and is only included as an example of what can be done.
4. The fourth section contains the commands that are actually run in the job. In this case we are using a **bash** shell.

Submitting the Job - qsub

The final step is to submit the job to the job scheduler:

```

-bash-4.1$ qsub ML_analysis.sh
148122.jobmgr.hpc.jcu.edu.au

```

Monitoring the Job - qstat

Once the job has been submitted you can monitor its progress by using the qstat command.

When you first submit your job it is placed into the job queue, and its status column contains Q, meaning the job is in the queue:

```

-bash-4.1$ qstat 148122.jobmgr.hpc.jcu.edu.au
Job ID          Name          User          Time Use S Queue
-----
148122.jobmgr  ML_analysis   jcxxxxxxx    0 Q normal

```

Once your job starts running its status changes to R:

```

-bash-4.1$ qstat 148122.jobmgr.hpc.jcu.edu.au
Job ID          Name          User          Time Use S Queue
-----
148122.jobmgr  ML_analysis   jcxxxxxxx    0 R normal

```

- You can display all of the jobs you are running:

```
-bash-4.1$ qstat -u jcxxxxx -ln
jobmgr.hpc.jcu.edu.au:
```

Req'd	Elap	Username	Queue	Jobname	SessID	NDS	TSK	Memory	Time
148185	00:00:17	jobmgr.hpc.jcu.jcxxxxx	normal	ML_analysis	0	1	1	5gb	500:00:00

- All of the jobs running on the cluster:

```
-bash-4.1$ qstat
```

Job ID	Name	User	Time Use	S	Queue
132501	jobmgr	DDWoodpca1L	jcxxxxxx	581:37:3	R normal
132502	jobmgr	DDWoodpca0.1L	jcxxxxxx	581:11:0	R normal
132503	jobmgr	DDWoodpca0.01L	jcxxxxxx	581:02:3	R normal
132504	jobmgr	DDWoodpca0.001L	jcxxxxxx	581:05:1	R normal
132560	jobmgr	IBWoodpca1L	jcxxxxxx	575:36:2	R normal
132561	jobmgr	IBWoodpca0.1L	jcxxxxxx	550:28:1	R normal
132562	jobmgr	IBWoodpca0.01L	jcxxxxxx	573:21:3	R normal
132563	jobmgr	IBWoodpca0.001L	jcxxxxxx	575:25:3	R normal
142918	jobmgr	DDpca0.001pcb1LL	jcxxxxxx	275:22:4	R normal
144744	jobmgr	...fw0_PBS_CG1.3	jcxxxxxx	3597:28:	R bigmem
144745	jobmgr	...fw0_PBS_CG1.3	jcxxxxxx	2279:22:	R bigmem
144746	jobmgr	...fw0_PBS_CG1.3	jcxxxxxx	1098:06:	R normal
144747	jobmgr	...fw0_PBS_CG1.3	jcxxxxxx	928:26:1	R normal
144748	jobmgr	...fw0_PBS_CG1.3	jcxxxxxx	942:19:5	R normal
144753	jobmgr	...fw0_PBS_CG1.3	jcxxxxxx	1434:04:	R normal
144754	jobmgr	...fw0_PBS_CG1.3	jcxxxxxx	927:20:1	R normal
144756	jobmgr	...fw0_PBS_CG1.3	jcxxxxxx	938:07:5	R normal
145377	jobmgr	...fw0_PBS_CG1.3	jcxxxxxx	1145:52:	R normal
145379	jobmgr	...fw0_PBS_CG1.3	jcxxxxxx	1286:49:	R normal
145381	jobmgr	...fw0_PBS_CG1.3	jcxxxxxx	1034:06:	R normal
145382	jobmgr	...fw0_PBS_CG1.3	jcxxxxxx	740:05:5	R normal
145384	jobmgr	...fw0_PBS_CG1.3	jcxxxxxx	1285:28:	R normal
145386	jobmgr	...fw0_PBS_CG1.3	jcxxxxxx	1028:30:	R normal
145387	jobmgr	...fw0_PBS_CG1.3	jcxxxxxx	1070:39:	R normal
145390	jobmgr	...fw0_PBS_CG1.3	jcxxxxxx	1238:34:	R normal
145391	jobmgr	...fw0_PBS_CG1.3	jcxxxxxx	2018:17:	R normal
145392	jobmgr	...fw0_PBS_CG1.3	jcxxxxxx	1585:10:	R normal
145708	jobmgr	...fw0_PBS_CG1.3	jcxxxxxx	1159:11:	R normal
145756	jobmgr	...fw0_PBS_CG1.3	jcxxxxxx	835:32:4	R normal
145790	jobmgr	...fw0_PBS_CG1.3	jcxxxxxx	1083:20:	R normal
146395	jobmgr	...001pcb2500LL	jcxxxxxx	145:38:3	R normal
146396	jobmgr	...001pcb7500LL	jcxxxxxx	145:00:3	R normal
146397	jobmgr	...001pcb12500LL	jcxxxxxx	152:09:2	R normal
146398	jobmgr	...001pcb5000LL	jcxxxxxx	151:49:3	R normal
146399	jobmgr	...001pcb15000LL	jcxxxxxx	143:16:2	R normal
146527	jobmgr	...fw0_PBS_CG1.3	jcxxxxxx	547:57:0	R normal
147055	jobmgr	...fw0_PBS_CG1.3	jcxxxxxx	362:59:2	R normal
147059	jobmgr	..._567_bfw0_PBS	jcxxxxxx	46:57:34	R normal
147060	jobmgr	..._547_bfw0_PBS	jcxxxxxx	46:56:16	R normal
147063	jobmgr	..._543_bfw0_PBS	jcxxxxxx	46:59:03	R normal
147065	jobmgr	...fw0_PBS_CG1.3	jcxxxxxx	297:58:1	R normal
147070	jobmgr	..._573_bfw0_PBS	jcxxxxxx	51:54:44	R normal
147071	jobmgr	..._549_bfw0_PBS	jcxxxxxx	51:45:14	R normal
147072	jobmgr	..._550_bfw0_PBS	jcxxxxxx	46:32:31	R normal
147073	jobmgr	..._532_bfw0_PBS	jcxxxxxx	51:36:30	R normal
147074	jobmgr	..._575_bfw0_PBS	jcxxxxxx	50:08:14	R normal
147096	jobmgr	..._582_bfw0_PBS	jcxxxxxx	51:10:26	R normal
147097	jobmgr	..._578_bfw0_PBS	jcxxxxxx	51:08:43	R normal
147101	jobmgr	..._568_bfw0_PBS	jcxxxxxx	46:43:19	R normal
147106	jobmgr	..._539_bfw0_PBS	jcxxxxxx	46:40:11	R normal
147115	jobmgr	...es.1km_1km.sh	jcxxxxxx	50:46:46	R normal

147203.jobmgr	NarrowBoxParent	jcxxxxxx	140:35:0	R	normal
147206.jobmgr	NarrowBoxParent	jcxxxxxx	150:53:4	R	normal
147207.jobmgr	NarrowBoxParent	jcxxxxxx	125:10:5	R	normal
147212.jobmgr	NarrowBoxParent	jcxxxxxx	151:57:4	R	normal
147213.jobmgr	NarrowBoxParent	jcxxxxxx	164:20:4	R	normal
147215.jobmgr	NarrowBoxParent	jcxxxxxx	160:11:4	R	normal
147216.jobmgr	NarrowBoxParent	jcxxxxxx	157:02:1	R	normal
147217.jobmgr	NarrowBoxParent	jcxxxxxx	163:39:0	R	normal
147218.jobmgr	NarrowBoxParent	jcxxxxxx	156:11:5	R	normal
147219.jobmgr	NarrowBoxParent	jcxxxxxx	140:37:4	R	normal
147220.jobmgr	NarrowBoxParent	jcxxxxxx	149:46:1	R	normal
147221.jobmgr	NarrowBoxParent	jcxxxxxx	141:12:4	R	normal
147222.jobmgr	NarrowBoxParent	jcxxxxxx	136:42:2	R	normal
147224.jobmgr	NarrowBoxParent	jcxxxxxx	132:20:5	R	normal
147225.jobmgr	NarrowBoxParent	jcxxxxxx	153:33:0	R	normal
147274.jobmgr	NarrowBox	jcxxxxxx	138:18:2	R	normal
147275.jobmgr	NarrowBox	jcxxxxxx	142:08:0	R	normal
147318.jobmgr	..._bfw0_PBS_1.2	jcxxxxxx	39:59:20	R	normal
147325.jobmgr	NarrowBox	jcxxxxxx	148:00:2	R	normal
147331.jobmgr	...fw0_PBS_CG1.3	jcxxxxxx	248:00:3	R	normal
147332.jobmgr	...fw0_PBS_CG1.3	jcxxxxxx	229:03:3	R	normal
147391.jobmgr	NarrowBox	jcxxxxxx	126:32:5	R	normal
147398.jobmgr	NarrowBox	jcxxxxxx	125:48:5	R	normal
147470.jobmgr	...fw0_PBS_CG1.3	jcxxxxxx	232:50:4	R	normal
147471.jobmgr	...fw0_PBS_CG1.3	jcxxxxxx	196:34:1	R	normal
147472.jobmgr	...fw0_PBS_CG1.3	jcxxxxxx	415:56:4	R	normal
147494.jobmgr	...or.lkm_lkm.sh	jcxxxxxx	25:36:45	R	normal
147495.jobmgr	...us.lkm_lkm.sh	jcxxxxxx	26:03:45	R	normal
147496.jobmgr	...us.lkm_lkm.sh	jcxxxxxx	26:04:44	R	normal
147498.jobmgr	...us.lkm_lkm.sh	jcxxxxxx	25:59:16	R	normal
147499.jobmgr	...ii.lkm_lkm.sh	jcxxxxxx	11:43:42	R	normal
147502.jobmgr	...fw0_PBS_CG1.3	jcxxxxxx	131:28:4	R	normal
147576.jobmgr	NarrowBox	jcxxxxxx	60:24:11	R	normal
147577.jobmgr	NarrowBox	jcxxxxxx	60:36:39	R	normal
147584.jobmgr	NarrowBox	jcxxxxxx	52:12:36	R	normal
147590.jobmgr	NarrowBox	jcxxxxxx	49:06:43	R	normal
147616.jobmgr	NarrowBox	jcxxxxxx	33:08:43	R	normal
147619.jobmgr	NarrowBox	jcxxxxxx	30:28:18	R	normal
147621.jobmgr	NarrowBox	jcxxxxxx	30:33:22	R	normal
147629.jobmgr	...es.lkm_lkm.sh	jcxxxxxx	07:04:15	R	normal
147630.jobmgr	...ae.lkm_lkm.sh	jcxxxxxx	07:15:13	R	normal
147631.jobmgr	...ra.lkm_lkm.sh	jcxxxxxx	0	Q	normal
147632.jobmgr	...ii.lkm_lkm.sh	jcxxxxxx	0	Q	normal
147634.jobmgr	...ta.lkm_lkm.sh	jcxxxxxx	0	Q	normal
147635.jobmgr	...mi.lkm_lkm.sh	jcxxxxxx	0	Q	normal
147636.jobmgr	...ii.lkm_lkm.sh	jcxxxxxx	0	Q	normal
147637.jobmgr	...ei.lkm_lkm.sh	jcxxxxxx	07:02:52	R	normal
147647.jobmgr	NarrowBox	jcxxxxxx	13:28:24	R	normal
147651.jobmgr	run_RSEM.sh	jcxxxxxx	21:46:18	R	normal
147652.jobmgr	NarrowBox	jcxxxxxx	19:52:35	R	normal
147657.jobmgr	NarrowBox	jcxxxxxx	18:04:53	R	normal
147665.jobmgr	NarrowBox	jcxxxxxx	16:17:01	R	normal
147702.jobmgr	NarrowBox	jcxxxxxx	13:03:45	R	normal
147820.jobmgr	NarrowBox	jcxxxxxx	10:16:38	R	normal
148021.jobmgr	NarrowBox	jcxxxxxx	08:41:26	R	normal
148022.jobmgr	NarrowBox	jcxxxxxx	08:42:34	R	normal
148087.jobmgr	NarrowBox	jcxxxxxx	07:45:33	R	normal
148095.jobmgr	NarrowBox	jcxxxxxx	07:03:12	R	normal
148096.jobmgr	NarrowBox	jcxxxxxx	07:10:18	R	normal
148099.jobmgr	NarrowBox	jcxxxxxx	06:54:12	R	normal
148100.jobmgr	NarrowBox	jcxxxxxx	06:38:54	R	normal
148101.jobmgr	NarrowBox	jcxxxxxx	06:46:31	R	normal
148108.jobmgr	NarrowBox	jcxxxxxx	05:57:17	R	normal
148124.jobmgr	NarrowBox	jcxxxxxx	05:11:52	R	normal
148126.jobmgr	NarrowBox	jcxxxxxx	04:49:25	R	normal
148138.jobmgr	NarrowBox	jcxxxxxx	04:30:48	R	normal
148150.jobmgr	NarrowBox	jcxxxxxx	03:46:22	R	normal
148154.jobmgr	NarrowBox	jcxxxxxx	03:19:45	R	normal
148179.jobmgr	NarrowBox	jcxxxxxx	00:10:33	R	normal
148180.jobmgr	NarrowBox	jcxxxxxx	00:10:57	R	normal
148182.jobmgr	NarrowBox	jcxxxxxx	00:30:19	R	normal

Deleting a job - qdel

If you need to your job you can use the qdel command

```
-bash-4.1$ qdel 148122.jobmgr.hpc.jcu.edu.au
```

Your job's Output

Different programs have different ways of outputting their data. If they output data directly to a file then your results will be in whatever file you specified. If, however, the results are printed out to the standard out (as is the case for this example) then PBS captures them into a file for you.

```
-----  
This job is allocated 1 CPU cores on  
n025nfs  
-----
```

```
PBS: Submitted to normal@n029.default.domain  
PBS: Working directory is /home/jcxxxxx/paup  
PBS: Job identifier is 148122.jobmgr.hpc.jcu.edu.au  
PBS: Job name is ML_analysis  
-----
```

```
P A U P *  
Portable version 4.0b10 for Unix  
Tue Mar 11 13:36:52 2014
```

```
-----NOTICE-----  
This is a beta-test version. Please report any crashes,  
apparent calculation errors, or other anomalous results.  
There are no restrictions on publication of results obtained  
with this version, but you should check the WWW site  
frequently for bug announcements and/or updated versions.  
See the README file on the distribution media for details.  
-----
```

```
Processing of file "~/ML_analysis.nex" begins...
```

```
Data read in DNA format
```

```
Data matrix has 8 taxa, 200 characters  
Valid character-state symbols: ACGT  
Missing data identified by '?'  
"Equate" macros in effect:
```

```
R,r ==> {AG}  
Y,y ==> {CT}  
M,m ==> {AC}  
K,k ==> {GT}  
S,s ==> {CG}  
W,w ==> {AT}  
H,h ==> {ACT}  
B,b ==> {CGT}  
V,v ==> {ACG}  
D,d ==> {AGT}  
N,n ==> {ACGT}
```

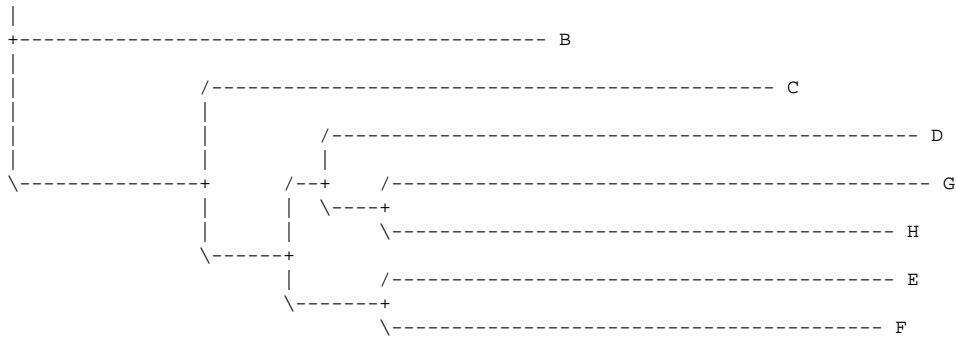
```
Neighbor-joining search settings:
```

```
Ties (if encountered) will be broken systematically  
Distance measure = uncorrected ("p")  
(Tree is unrooted)
```

```
Tree found by neighbor-joining method stored in tree buffer  
Time used = <1 sec (CPU time = 0.00 sec)
```

```
Neighbor-joining tree:
```

```
/----- A
```



Likelihood scores of tree(s) in memory:

Likelihood settings:

Number of substitution types = 2 (HKY85 variant)

Transition/transversion ratio estimated via ML

Assumed nucleotide frequencies (empirical frequencies):

A=0.35000 C=0.28813 G=0.20563 T=0.15625

Among-site rate variation:

Assumed proportion of invariable sites = none

Distribution of rates at variable sites = gamma (discrete approximation)

Shape parameter (alpha) = estimated

Number of rate categories = 4

Representation of average rate for each category = mean

These settings correspond to the HKY85+G model

Number of distinct data patterns under this model = 152

Molecular clock not enforced

Starting branch lengths obtained using Rogers-Swofford approximation method

Branch-length optimization = one-dimensional Newton-Raphson with pass

limit=20, delta=1e-06

-ln L (unconstrained) = 936.27218

Tree	1
-----	-----
-ln L	1646.41982
Ti/tv:	
exp. ratio	4.167819
kappa	8.796257
Shape	0.429541

Time used to compute likelihoods = 1 sec (CPU time = 0.79 sec)

Optimality criterion set to likelihood.

Heuristic search settings:

Optimality criterion = likelihood

Likelihood settings:

Number of substitution types = 2 (HKY85 variant)

Transition/transversion ratio = 4.16782 (kappa = 8.7962568)

Assumed nucleotide frequencies (empirical frequencies):

A=0.35000 C=0.28813 G=0.20563 T=0.15625

Among-site rate variation:

Assumed proportion of invariable sites = none

Distribution of rates at variable sites = gamma (discrete approximation)

Shape parameter (alpha) = 0.429541

Number of rate categories = 4

Representation of average rate for each category = mean

These settings correspond to the HKY85+G model

Number of distinct data patterns under this model = 152

Molecular clock not enforced

Starting branch lengths obtained using Rogers-Swofford approximation method

Trees with approximate likelihoods 5% or further from the target score are rejected without additional iteration

Branch-length optimization = one-dimensional Newton-Raphson with pass

limit=20, delta=1e-06

-ln L (unconstrained) = 936.27218

Starting tree(s) obtained via stepwise addition

Addition sequence: random

Number of replicates = 5

Starting seed = 1412047148

Number of trees held at each step during stepwise addition = 1

Branch-swapping algorithm: tree-bisection-reconnection (TBR)

```
Steepest descent option not in effect
Initial 'MaxTrees' setting = 100
Branches collapsed (creating polytomies) if branch length is less than or
equal to 1e-08
'MulTrees' option in effect
Topological constraints not enforced
Trees are unrooted
```

```
Heuristic search completed
Total number of rearrangements tried = 128
Score of best tree(s) found = 1645.76314
Number of trees retained = 1
Time used = 4 sec (CPU time = 3.49 sec)
```

Tree-island profile:

Island	Size	First tree	Last tree	Score	First replicate	Times hit
1	1	1	1	1645.76314	1	5

```
Processing of file "~/ML_analysis.nex" completed.
```

Other Examples

- [Running a job on the HPRC Cluster](#)
- [Running R Jobs on the HPRC Cluster](#)

Job Resources

It is important to match resources requested with the PBS Directives in your script and the resource usage of your job. There can be consequences for incorrectly specifying these resource requirements

- Walltime: your job can be killed if it exceeds the specified wall time.
- Memory: overusing memory can cause the compute node's memory to be pushed into swap space, slowing down all jobs on that node. This has also killed compute nodes in the past, destroying
- CPUs: using more cpus than requested can slow down all jobs running on that node.

Furthur Reading

1. [HPRC Cluster Explained](#)
2. [HPRC Cluster Job Management Explained](#)
3. [HPC PBSPro script files](#)