

# Biology PPC Cluster Information

## Overview

The Biology PPC Cluster comprises 26 Apple dual-processor 2.3-GHz compute-node G5s (15 with 4 GB of RAM and 11 with 8 GB of RAM) connected to a head node and a 4.5-TB Xserve RAID (RAID 50). Job management is handled by the Sun Grid Engine (SGE) package from Sun Microsystems, and the iNquiry Suite from the BioTeam. The Java version is 1.5.

## Access and Accounts

Access to the cluster is limited to the head node (dalmo.biol.mcgill.ca) via SSH (including sFTP), AFP (Mac file sharing), and SMB (Windows file sharing) from the 132.192.0.0/11 subnet only. User accounts are hosted on the RAID.

## Submitting Jobs

The SGE, which schedules and distributes jobs to the compute nodes, is a command-line scheduler. Jobs, by default, are scripts or script-wrappers calling other programs, but may also be actual binaries (see 'qsub -b'). Here is an example of a basic script:

```
---
#!/bin/bash
#Rtest.sh

/common/custom/R-2.9.0/R.framework/Resources/R --version > /Users/scottyb/Rtest.txt

exit 0
---
```

All scripts must be written with Unix line breaks (!!). Here is how you submit a job to the SGE:

```
---
qsub ./myScript.sh
---
```

see the qsub man page for more info. (there are a lot of options). Note that you can also imbed the qsub options within the script, as so:

```
# Request shell
#$ -S /bin/bash

# date-time to run, format [[CC]yy]MMDDhhmm[.SS]
#$ -a 12241200

# If I run on dec_x put stderr in /tmp/foo, if I
```

```
# run on sun_y, put stderr in /usr/me/foo
#$ -e dec_x:/tmp/foo,sun_y:/usr/me/foo

# Export these environmental variables
#$ -v PVM_ROOT,FOOBAR=BAR

# The job is located in the current
# working directory.
#$ -cwd

etc.
```

To check that your job has been successfully submitted and is running on a node, use:

```
---
qstat -f
---
```

You will see under the node identifier your job ID, your job name, your user name, the job status ('r' for running) and the date and time it was started. The job ID is important, because if you decide to kill a job you need that number:

```
---
qdel <job ID> (may need to add '-f' to force deletion)
---
```

Completed jobs generate stdout (.o) and stderr (.e) files (by default in your home directory, unless '-cwd' is specified) in addition to any user-defined output files. More complicated script examples can be found at /common/sgе/examples/jobs. Note that a web interface also exists for job execution, but one is then limited to the provided iNquiry Suite applications (a separate web-based account is also required).

Always use qsub to submit jobs—DO NOT EVER run jobs on the head node.

### **Other Useful SGE Commands**

qhold <job ID> : prevent a job scheduled for execution in the queue from being considered.  
qrls <job ID> : release a previously defined job hold.

### **Distributed Computing vs. Explicit Parallelism**

The default setup for the cluster is as a distributed-computing environment ("embarrassingly parallel" parallelism), where individual jobs are accorded a dedicated CPU for execution (note that since each node has two CPUs, programs written to effect multi-threaded parallelism will take advantage of both CPUs). Explicit parallelism (i.e., a single job executing over many nodes) is possible, but this requires implementation of a parallel-programming API, like MPICH or LAM-MPI. The cluster has the MPICH environment, but in order to take advantage of it, users must

either use a mpi-aware application (like MPIBlast), or write their own mpi-aware code.

## **NFS Mounts**

The following head-node directories are shared with the compute nodes:

/common

/Library/Perl --> Note: This mount is seen by the compute nodes as /RemotePerl

/Volumes/BioCluster1/Users --> Note: This mount is seen by the compute nodes as /Users

Users, by default, only have write access to their ~ in the /Users directory (or to /scratch on the nodes—see next section).

## **NFS Caveat (IMPORTANT!!!)**

The Network File System (NFS) is the weak point of the cluster foundation. It can tolerate large bursts of activity (i.e., acute usage), but not consistent activity (i.e., chronic usage). Whenever possible, avoid having persistent, open file handles across the NFS: too much continuous NFS activity between the head node and the compute nodes can bring the NFS down, which necessitates a cluster reboot. This is bad. If your jobs are I/O heavy, the best thing to do is to have your jobs run locally: at the start of your scripts, copy the data you need to /scratch (each compute node has a /scratch for this purpose), write any temp. files to /scratch, and then at the end copy what you need back to your ~ (in /Users). Please do not forget to delete your data or temp. files in /scratch afterward!!!

## **Memory-Intensive Jobs**

Jobs that *require* a 64-bit memory space (up to 8 GB) can be run on the "g5hM" queue using the '-q' flag with 'qsub', like so: 'qsub -q g5hM.q ./testScript.sh'

## **The Big Book of SGE**

<http://dalmo.biol.mcgill.ca/bipod/doc/sge/SGE53Ref.pdf>