

# Shared Computing Cluster Usage Cheat Sheet

## Useful Links

RCS Website: rcs.bu.edu  
email: help@scc.bu.edu  
Software list: rcs.bu.edu/software  
Examples: rcs.bu.edu/examples

## qsh/qysh (submit an interactive job)

qsh Submit an interactive X-windows session  
qysh Submit an interactive rsh session

## qsub (submit a batch job)

-P *project-name* Project name  
-N *job-name* Job name  
-l *h\_rt=hh:mm:ss* Hard time limit  
-m *e* Send an email when the job ends  
-m *ea* Send an email when the job ends or is aborted  
-M *my.email@gmail.com* Use non-BU email address  
-j *y* Merge error and output files into a single file  
-l *mem\_per\_core=8G* At least 8G of memory per core (6G, 8G, 9G, 16G, 18G)  
-pe *omp N* Request multiple slots (cores), i.e. 4, 8, 12, 16, 28, 36  
-pe *mpi\_16\_tasks\_per\_node N* Submit MPI job ( 16 or 28 tasks per node )  
-t 1-10 Submit 10 tasks  
-l *gpus=1* Request 1 GPU  
-l *gpu\_c=3.5* Request GPU capability at least 3.5 (6.0 for P100)  
-l *gpu\_type=K40m* Specify GPU type (M2070, K40, P100)  
-hold\_jid *joblist* Setup job dependency list  
-b *y* Submit binary program  
-l *buyin* Force the job to run only on a buyin node  
-q *queue-name* Force the job to run only in a specific queue  
-verify Instead of submitting a job, prints info about the job

## qstat (get information about current jobs)

qstat List of all current jobs  
qstat -u *user-id* All current jobs submitted by the user *user-id*  
qstat -s *r* List of running jobs  
qstat -s *p* List of pending jobs (hw, hqw, Eqw...)  
qstat -u *user-id* -r Display the resources requested by the user for his jobs  
qstat -u *user-id* -ext Extended info about the user's jobs  
qstat -u *user-id* -s *r* -t Display info about sub-tasks of parallel jobs  
qstat -j *job-id* Display job status  
qstat -g *c* Display the list of queues and load information  
qstat -q *queue* Display jobs running on a particular queue

## qdel (delete job from the queue)

qdel *job-id* Delete job *job-id*  
qdel *job-id* -t 5-7 Delete tasks 5 through 7 for job *job-id*  
qdel -u *user-id* Delete all the jobs submitted by the user

## module (software environment)

module avail List available packages  
module avail python List all available versions of python  
module load *python/2.7.13* Load python module version 2.7.13  
module unload *python/2.7.13* Unload the module  
module show *python/2.7.13* Show the content (env. variables) of module *python/2.7.13*  
module help *python/2.7.13* View information/help for specific module  
module list List all loaded modules  
module purge Unload all loaded modules  
module keyword *statistics* list all modules with specific keyword  
module help List options for the module command  
module avail -t 2>\& 1 | less Pipe module list to *less* command  
moduleavail | grep -i bowtie Fast search

## acctool (account information)

acctool -b *y* SU balance summary of all the projects I belong to  
acctool -u *user-id* -b *y* SU balance summary of all the projects *user-id* belongs to  
acctool 06/18/15 Number of jobs and wallclock report for the day  
acctool -d 2 06/18/15 Number of jobs and wallclock detailed report for the day  
acctool -d 2 -t 5 06/18/15 Display detailed report for the top 5 jobs for the day  
acctool -d 4 06/18/15 Most detailed report for all the jobs that finished on particular day  
acctool -j *job-id* 06/18/15 Report for job with given job ID.

## qacct (past job information)

qacct -j *job-id* Detailed report about job *job-id*  
qacct -d 3 -o *user-id* -j Detailed report about all the jobs user ran in the past 3 days  
qacct -d 3 -o *user-id* -q *queue* -j Detailed report about all the jobs user ran using queue *queue*  
qacct -P *project-id* Summary report for the project (current year usage)

## quota (home directory space usage)

Home directory quota: 10GB  
quota Display my Home directory usage  
quota -s Display my Home directory usage in human-readable format  
quota *user-id* Display Home directory usage for *user-id*

## pquota (Project Disk Space usage)

Project directories quota: Up to 1TB with 200GB limit for /project partition  
pquota Project Disk quota and usage for all the projects I belong to  
pquota -u *project* Project Disk quota and usage for *project*

## User Guidelines

15 minutes CPU time on login nodes  
12 hours Default wall clock time for a job  
720 hours Wall clock time limit for a single-node job  
120 hours Wall clock time limit for mpi job running on multiple nodes  
48 hours Wall clock time limit for gpu jobs

## Connecting to the Shared Computing Cluster

<code>scc1, scc2, scc3(geo), scc4</code>	SCC login nodes
<code>ssh username@scc2.bu.edu</code>	Windows (in mobaXterm)
<code>ssh -Y username@scc2.bu.edu</code>	Mac
<code>ssh -X username@scc2.bu.edu</code>	Linux

## Working with the Project Disc Space

<code>groups</code>	List all projects which I belong to
<code>cd /project/myproject</code>	Change directory to the /project directory
<code>cd /projectnb/myproject</code>	Change directory to the /projectnb directory
<code>cd /restricted/project/myproject</code>	Change directory to the /restricted/project directory (from scc4 only)
<code>cd /restricted/projectnb/myproject</code>	Change directory to the /restricted/projectnb directory (from scc4 only)

## Available editors

<code>emacs</code>	Text editor ("the extensible, customizable, self-documenting, real-time display editor")
<code>vi, vim, gvim</code>	Another popular text editor
<code>gedit</code>	GNOME notepad-like text editor
<code>nano</code>	GNU text editor with command-line interface

## Commands to transfer files and Popular FTP clients

Note: The following `scp` commands should be executed on the local machine.

<code>scp filename username@scc1.bu.edu:~</code>	Upload file from your local machine to your home directory on the SCC
<code>scp filename username@scc4.bu.edu:/project/myproject</code>	Upload file from your local machine to your specified project directory on the SCC
<code>scp username@scc4.bu.edu:/project/myproject/filename .</code>	Download file from your project directory on the SCC to the current directory on your local machine
<code>rsync filename username@scc1.bu.edu:~</code>	sync a file from your local machine with the file in your home directory on the SCC

<code>wget http://www.site.org/file</code>	Download a file from a website
--	--------------------------------

<code>Cyberduck</code>	Windows and MAC FTP client
<code>FileZilla</code>	Windows and MAC FTP client
<code>WinSCP</code>	Windows FTP client

<code>dos2unix filename</code>	Convert file with DOS/MAC characters to UNIX/Linux format (execute on SCC)
--------------------------------	--

## Snapshots

<code>.snapshots/\textit{YYMMDD}</code>	Snapshots directory structure
<code>ls .snapshots/161205</code>	View the snapshot of the directory created on December 5th, 2016

<code>180 days</code>	Snapshots are stored for /project and home directories
<code>30 days</code>	Snapshots are stored for /projectnb directories

## VNC (scc2, scc3(geo) and scc4 only)

<code>[scc2 ] vncpasswd</code>	Set VNC password
<code>[scc2 ] vncstart</code>	Start VNC server
<code>[scc2 ] vncstart -geometry 1900x1200</code>	Start VNC server with specific screen resolution
<code>[local ] ssh koleinik@scc2.bu.edu -L 7777:localhost:5906</code>	Configure tunnel (must be executed in the local terminal window)
<code>localhost:7777</code>	Connect with VNC client