

# **TSCC Quick Reference**

## **Login**

#### ssh login@tscc-login.sdsc.edu

Use ssh from a terminal window on Linux or Mac systems; PuTTY and WinSCP are common ssh clients for Windows users. Users with ids in the UCSD Active Directory database can use their UCSD AD password to login; others will need to send a public ssh key to the TSCC administrators to allow access. SSH keys may be generated using tools such as PuTTYgen and ssh-keygen. Cut and paste the public key into the body of an email or send as a separate file.

Login systems are intended for editing, compiling, and job submission—analysis applications should only be run on compute nodes. (See *Accessing Compute Nodes* below.)

#### **Software**

In addition to the CentOS v6.3 software, the TSCC provides many compilation, communication, and analysis programs, found in /opt. Access to these programs is managed through the Unix **module** command, which configures your environment variables as necessary. For example, **module load pgi** sets the environment variables needed to access the pgi compilers. **module avail** displays a list of all available modules. See the module man page for more options. Some modules (e.g., fftw) are compiler- and/or mpi-dependent; to load these, you need to load a compiler and/or an mpi module first.

Compilers: gnu v4.4.6 (/usr/bin) and v4.8.1 (/opt); intel v2013.1.117; pgi v13.2; java 1.7.0\_21 MPI: mvapich2 for infiniband v1.9a2; openmpi for infiniband/ethernet v1.6.3

#### Accessing Compute Nodes

**qsub -I** gives interactive access to one of the TSCC compute nodes via the hotel queue. To submit a batch script, replace the -I in this command with the path to the script. Other common qsub options:

-A account indicates which fund to charge the job to. Each user has a default fund (shown by glsuser \$USER), so this option is only useful for users with access to multiple funds.
-I nodes=count[:ppn=count][:properties] specifies the number and type of nodes and cores to allocate to the job. Supported properties include: mem32/mem64/mem128/mem256 (node memory size in GB); ib/noib (nodes with/without an Infiniband connection); gpu (nodes with four GPUs).
-I walltime=[[hours:]minutes:]seconds indicates the amount of time to reserve for the job. After completion, jobs are charged only for the amount of time used, not the amount reserved.
-q queue indicates which set of nodes to use for the job. Queues also differ in the resource limits placed on jobs running through them.

-q hotel run on the hotel nodes (40 compute w/IB and 64GB memory); 72 hour time limit.

-q gpu-hotel run on the hotel GPU nodes (3 compute w/4 GeForce 680); 72 hour time limit.

-q pdafm run on the pdafm nodes (4 compute w/512GB memory); 72 hour time limit.

-q home (condo users only) run on purchased nodes; no time limit.

-q home-group (group members only) run on nodes purchased by a particular group; useful only to users with access to multiple groups' home nodes; no time limit.

-q condo (condo users only) run on all condo nodes; 8 hour time limit.

-q gpu-condo (condo users only) run on all condo GPU nodes; 8 hour time limit.

-q glean (condo users only) run on all condo nodes; no time limit and not charged against SU balance, but glean jobs may be killed in favor of jobs from home and condo queues. The system performs no checkpointing of killed jobs.

See the qsub and pbs\_resources man pages for a complete list of options. Other useful commands:

**checkjob** *jobid* displays the status of a submission—especially useful for determining why a submitted job isn't running

**gbalance -u \$USER** displays the number of SUs (core-hours) you have available for jobs **qstat** or **showq** displays the status of all submitted jobs **qdel** *jobid* cancels a submitted job

## **Storage**

Each user has a home directory (/home/login) with a 100GB quota. Groups may obtain an additional, shared 100GB project directory (/projects/group). Daily snapshots of these directories allow you to restore files that have been mistakenly deleted; home snapshots can be accessed via **cd** ~/.**zfs/snapshot**. Each user also has a directory (/oasis/tscc/scratch/login) on the Lustre shared file server. This directory is designed to provide running jobs with fast, parallel access to files, rather than for long-term storage. Files on oasis are not backed up and are subject to deletion after 90 days; longer expiration times for individual data sets can be arranged.

Each running job receives an allocation of storage on the local file system, the path to which is stored in the TMPDIR environment variable. This storage is purged when the run completes.

Additional storage can be purchased from SDSC Project Storage project for mounting on the TSCC (see http://www.sdsc.edu/services/StorageBackup.html), and lab NFS servers can also be mounted.

# <u>File Transfer</u>

**scp** is your best choice for small files. Very large files should be transferred via an allocated compute node and **bbftp**—either the client or server can be run on the TSCC. See the TSCC web pages for a description of installing and using bbftp. Portable disk drives can be mounted temporarily on the TSCC by special arrangement.

# **Getting Help**

See rci.ucsd.edu/computing for more information on system characteristics and usage. mail tscc-l@mailman.ucsd.edu—questions/information/help requests for the TSCC user base. mail tscc-support@ucsd.edu—questions/help requests for TSCC User Services.