SGE Basic commands

If you share a cluster with other users, a batch schedule allows for optimal sharing among users. Grid Engine is a robust batch scheduler that can handle large workloads across entire organizations.

Basic Commands

At a basic level, Sun Grid Engine (SGE) is very easy to use. The following sections will describe the commands you need to submit simple jobs to the Grid Engine. The command that will be most useful to you are as follows

- qsub - submit a job to the batch scheduler
- qstat - examine the job queue
- qdel - delete a job from the queue

A more convenient queue status package called userstat combines qstat, qhost, and qdel into a simple easy to use "top" like interface. Each will be described below. Additional information on these commands is available by using `man` *`command-name`*

Submitting a job to the queue: qsub

Qsub is used to submit a job to SGE. The qsub command has the following syntax:

```
qsub [ options ] [ scriptfile | -- [ script args ]]
```

Binary files may not be submitted directly to SGE. For example, if we wanted to submit the "date" command to SGE we would need a script that looks like:

```
#!/bin/bash
bin/date
```

If the script were called **sge-date**, then we could simply run the following:

```
$ qsub sge-date
```

SGE will then run the program, and place two files in your current directory:

```
sge-date.e#
sge-date.o#
```

where # is the `job number` assigned by SGE. The sge-date.e# file contains the output from standard error and the sge-date.o# file contains the output form standard out.

The following basic options may be used to submit the job using **qsub**.

```
-A    [account name] -- Specify the account under which to run the job
-N    [name] -- The name of the job
-l    h rt=hr:min:sec -- Maximum walltime for this job
-r    [y,n] -- Should this job be re-runnable (default y)
-pe   [type] [num] -- Request [num] amount of [type] nodes.
-cwd  -- Place the output files (.e,.o) in the current working directory.
      The default is to place them in the users home directory.
-S    [shell path] -- Specify the shell to use when running the job script
```

Although it is possible to use command line options and script wrappers to submit jobs, it is usually more convenient to use just a single script to include all options for the job. The next section describes how this is done.

Job Scripts

The most convenient method to submit a job to SGE is to use a "job script". The job script allows all options and the program file to placed in a single file. The following script will report the node on which it is running, sleep for 60 seconds, then exit. It also reports the start/end date and time as well as sending an email to user when the jobs starts and when the job finishes. Other SGE options are set as well. The example script can be found here as well.

```
#!/bin/sh
#
# Usage: sleeper.sh [time]]
#        default for time is 60 seconds

# -- our name ---
#$ -N Sleeper1
#$ -S /bin/sh
# Make sure that the .e and .o file arrive in the
# working directory
```

```
#$ -cwd
#Merge the standard out and standard error to one file
#$ -j y
/bin/echo Here I am: `hostname`. Sleeping now at: `date`
/bin/echo Running on host: `hostname`.
/bin/echo In directory: `pwd`
/bin/echo Starting on: `date`
# Send mail at submission and completion of script
#$ -m be
#$ -M deadline@kronos
time=60
if [ $# -ge 1 ]; then
   time=$1
fi
sleep $time

echo Now it is: `date`
```

The "#$" is used in the script to indicate an SGE option. If we name the script `sleeper1.sh` and then submit it to SGE as follows:

```
qsub sleeper1.sh
```

The output will be in the file `Sleeper1.o#`, where # is the job number assigned by SGE. Here is an example output file for the `sleeper1.sh` script. When submitting MPI or PVM jobs, we will need additional information in the job script. See below.

Preserving Your Environment

If you want to make sure your current environment variables are used on you SGE jobs, include the following in your submit script:

```
#$ -V
```

Queue Status: qstat

Queue status for *your jobs* can be found by issuing a qstat command.

More detail can be found by using the `-f` option. An example `qstat -f` issued by user *deadline* is shown below.

To look at jobs for all users, you must issue the following:

```
qstat -u "*"
```

For queue details, you may add the `-f` option as shown above. If you prefer to always see all user jobs, you can use the *alias* command to make this the default behavior. For bash users add the following to your.*bashrc* file.

```
alias qstat='qstat -u "*"'
```

For c shell users, the following can be added to your *.cshrc* file:

```
alias qstat 'qstat -u "*"'
```

Even more data information can be obtained by using the -F option (see the qstat man page for more information. For parallel jobs, the output is not very easy to understand. See userstat for a better display of the data. In the above listing, the stat is either qw (queue waiting), t (transferring), and r (running).

Why Won't My Jobs Run ?

There are several reasons why a job will not run. The first reason is due to the job resource requirements. It is possible that the cluster is full and you have to wait for available resources (processors etc.)
It is also possible the job may have experienced and error in the run script. In which case the status would be "Eqw". You can query a job's status by entering the following:

```
 qstat -explain c -j _Job-ID_
```

where `_Job-ID_` is the Grid Engine job number.

Deleting a Job: qdel

Jobs may be deleted by using the qdel command as follows:

```
$ qdel job-id
```

The *job-id* job number is the number assigned by SGE when you submit the job using qsub. You can only delete you jobs.