Grid Engine Troubleshooting

- There are two core problem types
  - Job Level
    - Cluster seems OK, example scripts work fine
    - Some user jobs/apps fail
  - Cluster Level
    - Problems running all jobs
    - Problems submitting to certain PE/queue/Project
    - Problems with jobs on certain nodes
Grid Engine Troubleshooting

- Dealing with Cluster Level problems
  -STDOUT/STDERR from user jobs still the best initial debug resource
  -SGE messages and logs are usually very helpful
    - $SGE_ROOT/default/spool/qmaster/messages
    - $SGE_ROOT/default/spool/qmaster/schedd/messages
  -Execd spool logs often hold job specific error data
    - Remember that local spooling may be used (!)
    - $SGE_ROOT/default/spool/<node>/messages
  -SGE panic location
    - Will log to /tmp on any node when $SGE_ROOT not found or not writable
Cluster Level Troubleshooting

- Can’t run SGE commands
  - Command not found
  - System not responding
  - Remote operation permission denied
- Try:
  - `qhost` and `qstat -f`
- Likely root cause
  - SGE `settings.sh` script not run/initialized
  - One or more key SGE daemons not running
Cluster Level Troubleshooting

Queue Error States
- When "qstat -f" shows one or more queues in state “E"
- This is bad, usually means a job failed in a spectacular manner
  - SGE invokes E state to prevent “black hole” effect
- Root cause is often system/OS, file system related
  - User does not exist on node, NFS glitch, uid/gid mismatch, etc.
- Solution
  - Once it is determined the problem is not systemic or persistent one can clear E states via “qmod -c”
Cluster Level Troubleshooting

- If you think a node has issues
  - Disable the queue on the node
    - Will NOT affect any running jobs on that node
    - WILL block any new work from landing there
    - Disabled state “d” will persist until cleared
  - Command:
    - `qmod -d <queue name>`
    - To re-enable:
      - `qmod -e <queue name>`
Job Level Troubleshooting

- Job dies instantly
  - First pass
    - Check the .o and .e files in the job directory
    - Check .po and .pe files for parallel MPI jobs
    - Best resource, usually clear error messages found:
      - Permission problem, no license available, path problem, syntax error in app, etc.
  
- Second pass
  - Check qmaster spool messages and node execd messages
Job Level Troubleshooting

- Job dies instantly …
- Third pass
  - `qsub -w v <full job request>`
    - This will tell you if the job can run assuming:
      - All slots on all queues were empty
      - All load values were ignored
    - Good source of info on ‘why can’t my job be scheduled’ problems
Job Level Troubleshooting

- Job pending forever
  - First Pass:
  - `qstat -j <job_id>`
  - This will tell you why the job is pending and if there are any reasons why queues cannot accept the job
- Possible root causes
  - Impossible resource requested, license not available
  - Scheduling oddness
Job Level Troubleshooting

- Job pending forever
  - Second Pass:
    - $SGE_ROOT/default/spool/qmaster/schedd/messages
  - Just to see if anything weird is going on with the scheduler
Job Level Troubleshooting

- Job runs from command line on front end node, but not under Grid Engine
- Most common root cause:
  - Difference in environment variables
  - Difference in shell execution environment
General Troubleshooting

- Many times the problems are not SGE related
  - Permission, path or ENV problems
- Best thing to do is watch STDERR and STDOUT
  - Use the qsub ‘-e’ and ‘-o’ switches to send output to a file that you can read
  - Use qsub ‘-eo’ to send STDOUT and STDERR to the same file (useful for debugging)
General Troubleshooting (cont.)

- To get email listing why a job aborted
  - Use: `qsub -m a user@host [rest of command]`

*SGE training, consulting and special projects - BioTeam Inc. - http://www.bioteam.net*
General Troubleshooting (cont.)

- Checking exit status and seeing if jobs ran to completion without error
  - Use: ‘qacct -j <job_id>’ to query the accounting data
  - Will also tell you if the job had to be requeued onto a different queue or exechost
Basic Debug Process

- Verify for yourself that cluster and SGE is happy before you do anything else
  - ‘qstat -f’, ‘qrsh hostname’, ‘qhost’, etc.
    - This will quickly identify systemic or cluster wide issues
- Then move on to dealing with the specific issue
Basic Debug Process

- If problems persist, verify that the application actually runs OUTSIDE of Grid Engine
  - Easier to catch app/user/system issues
  - Good way to catch the super subtle stuff
  - This is especially useful for MPI parallel programs
Recommendation

- Build a personal portfolio of simple testing scripts
  - qrsh hostname
  - $SGE_ROOT/examples/jobs/simple.sh
  - $SGE_ROOT/examples/jobs/sleeper.sh

- Get your users to supply you with example or dummy scripts that use real portfolio apps
Other Troubleshooting
Trigger a scheduler trace

- Not well documented feature
- As ‘root’ run the command:
  - ‘qconf -tsm’
- Will create a 1-time scheduler trace file
  - $SGE_ROOT/default/common/schedd_runlog
- Useful for dealing with scheduling issues and fine tuning policy configurations
Increase log verbosity

- A SGE configuration parameter
  - “qconf -sconf”
- Default value:
  - log_info
- Options
  - log_info | log_warning | log_err
**qping**

- Generally a developer tool
- Being extended for admin troubleshooting
  - Likely to become a more important utility
- Focused on communication debugging
- Not many online guides yet
  - “qping -help” may be best bet at this time
Enable job spool dir retention

- Each job gets its own unique spooling directory
- Contains useful information about job and the environment
- This is normally deleted automatically as job exits
Job spool dir retention

- Set "KEEP_ACTIVE=TRUE" for execd_params in the global or host-specific configuration
- This disables the job spool dir deletion
- Now directory contents can be analyzed
Job spool dir retention

- Where are the spool dir files?
- Inside “active_jobs/” in the
  $SGE_ROOT/default/spool/<node>/> directory
- Within active_jobs/ directories are named
  by JobID and array task number
  - active_jobs/1.[taskID]
  - active_jobs/2.[taskID]
  - ...

SGE training, consulting and special projects - BioTeam Inc. - http://www.bioteam.net
Job spool dir file contents

- **Config**
  - Job configuration parameters
- **environment**
  - All env variables to be set up for the job
- **error**
  - Notices of severe errors during job startup
- **exit_status**
  - Unix exit status for the job
- **job_pid**
  - Job process id (shepherd child process)
- **pid**
  - Process ID of the shepherd
- **trace**
  - Detailed debug info about job execution
Job spool dir retention

- Remember to turn this feature off!
- Puts extra load on SGE
- Takes up lots of space
Trigger SGE Debug Output

- Special ENV var(s) trigger debug output
  - Some useful for admins, most useful only for Grid Engine Developers

- General process:
  1. Source the “dl.sh” or “dl.csh” script found in $SGE_ROOT/util/
  2. Run command “dl <level>”
  3. Levels 1,3,5 are most useful

- Try for yourself:
  - $ source /$SGE_ROOT/util/dl.sh
  - $ dl 1
  - $ qrsh hostname

SGE training, consulting and special projects - BioTeam Inc. - http://www.bioteam.net
DanT on Debug Output

- Daniel Templeton
  - SGE developer, one of the DRMAA evangelists, frequent Sun blogger on SGE topics
    - Site: http://blogs.sun.com/templedf/

- Permalink to post on debug output