High Performance Computing Facility Specifications, Policies and Usage

Supercomputer Project

Bibliotheca Alexandrina

Topics

- Specifications Overview
- Site Policies
- Intel Compilers
- Intel MPI
- Sun Grid Engine (SGE) Job Scheduler

Specifications Overview

- Hardware
  - 130 Compute Nodes
  - ~12 TFLOPS peak performance
  - 9.1 TFLOPS LINPACK benchmark
  - ~1 TBytes main memory
  - 10 Gbit/s infniband interconnect
  - 36 TBytes shared scratch storage (raw)
  - High-density, highly automated tape storage

Specifications Overview (cont'd)

- Software
  - Operating System
    - RHEL 5.2
  - Compilers
    - GNU Compiler Collection 4.1.2 (gcc, g++, … etc.)
    - Intel Compiler 11.0
    - Intel MPI3.2
    - OpenMPI, MPICH, ...etc.

Specifications Overview (cont'd)

- Software (cont'd)
  - Management
    - Provisioning: ROCKS
    - Monitoring: Ganglia
  - Shared Scratch Storage
    - Lustre File System
  - Backup Software
    - Veritas Netbackup

Site Policies

- Home Directory
  - NFS
  - Hosted on a separate node
  - Shared among all users
  - Very limited space; 100MB/user

Site Policies (cont'd)

- Working Directory
  - Lustre File System
  - Hosted on SunFire X4500 server
  - Shared among all users
  - Quota/user not yet decided

Site Policies (cont'd)

- Backup Policy
  - Archive Working Directory
  - Full Backup
    - Frequency: 1/Week
    - Retention Period: 2 Weeks
  - Differential-Incremental Backup
    - Frequency: 1/Day
    - Retention Period: 2 Weeks

Intel Compilers

- Load Intel Compiler 11
  - $ module load intel/compiler/11.0
- Fortran
  - ifort [options] file1 [file2]
- C/C++
  - icc/icpc [options] file1 [file2 …]

Intel MPI

- Load Intel MPI modules
  - $ module load intel/mpi/3.2
  - This will load the underlying intel/compiler/11.0
- Fortran MPI Wrappers
  - mpif77, mpif90 [options] <files>
- C/C++ MPI Wrappers
  - mpicc/mpic++ [options] <files>

SGE Job Scheduler

- Load sge6.2 module
  - $ module load sge6.2
- Most useful SGE commands
  - qsub / qdel (Submit jobs & delete jobs)
  - qstat & qhost (Status info about queues, hosts and jobs)
  - qacct (Summary info on completed job)

SGE Job Scheduler (cont'd)

- qsub General format:
  - $qsub <qsub options> program <prog options>
- Useful qsub options

Option	Description
-b y[es]|n[o]	Indicate explicitly whether command should be treated as binary or script.
-cwd	Execute the job from the current working directory.
-j y[es]|n[o]	Specifies whether or not the standard error is merged into the standard output.
-l resource=value,...	Launch the job in a queue meeting the given resource request list.
-pe parallel_environment n	Parallel programming environment (PE) to instantiate.
-r y[es]|n[o]	Identifies the ability of a job to be rerun or not, in case the node on which the job is running crashes.

SGE Job Scheduler (cont'd)

- How to provide options to qsub
  - Default request file ($HOME/.sge_request)
  - In job script (preceded by #$)
  - On the command line

SGE Job Scheduler (cont'd)

- qstat General format:
  - $ qstat <qstat options>
- Useful qstat options

Option	Description
-explain a|A|c|E	Displays the reason for the state of a queue instance. 'a' shows the reason for the alarm state. Suspend alarm state reasons will be displayed by 'A'. 'E' displays the reason for a queue instance error state.
-f	Shows a summary of information on all queues to be displayed along with the queued job list.
-j [job_list]	Prints various information for all jobs.
-ne	In combination with -f option suppresses the display of empty queues.

SGE Job Scheduler (cont'd)

- qacct General format:
  - $ qacct <qacct options>
- Useful qacct options

Option	Description
-j [ID|name]	Print the accounting information about a finished job given its names or ID. If neither a name nor an ID is given all jobs are enlisted.
-o [Owner]	The name of the owner of the jobs for which accounting statistics are assembled.
-P [Project]	The name of the project for which usage is summarized.

SGE Job Scheduler (cont'd)

- Debugging your jobs
  - qstat -j <job_id> (problems with pending jobs)
    - This will tell you why the job is pending
  - qsub -w v <full job request> (verify submit)
    - This will tell you if the job can run
  - Watch your STDERR and STDOUT
  - qacct -j <job_id>
    - Check exit status to see if jobs were completed.
  - Ask for help if you get stuck


Thanks
supercomputer@bibalex.org
Bibliotheca Alexandrina