High Performance Computing (HPC)

Boise State Research Computing offers a high performance computing Linux environment to provide university researchers sufficient computing power to do their intensive data analysis.

Overview

The HPC environment consists of:

- **R2** (r2.boisestate.edu)

Use of this environment is provided to those users with significant computational and/or large data manipulation needs.

The HPC is designed to meet the needs of the following:

- Faculty, staff, graduate and professional students requiring high end computing functionality
- Undergraduate students with faculty sponsorship requiring high end computational functionality
- Authorized guests from other research institutions working in collaboration with university employees

Resources

There are two methods in which you can compile software on HPC.

- The first method is compiling within your home directory, making the software accessible by you and you only.
- The second method is to have the HPC staff install in a shared folder that allows other members/groups on HPC to use the software.

The compiler software includes:

- GNU, PGI & Intel compiler suites
- C, C++ & Fortran.

Programming Languages

- Python 2/3, R, Matlab, Mathematica

Most typical HPC software and libraries are already installed on R2, and have modules built.

Modules

On a complex computer system, on which it is necessary to make available a wide choice of software packages in multiple versions, it can be quite difficult to set up the user environment so as to always find the required executables and libraries. (This is particularly true where different implementations or
versions use the same names for files). Environment modules provide a way to selectively activate and deactivate modifications to the user environment which allow particular packages and versions to be found.

The basic command to use is

**module:**

```
module
(no arguments)              print usage instructions
avail or av                 list available software modules
whatis                      as above with brief descriptions
load <modulename>     add a module to your environment
unload <modulename>   remove a module
purge                       remove all modules
```

Enter the command *module avail* to see the entire collection of currently available modules.

Some modules refer to administrative software and are not of interest to users, e.g. cluster-tools is the Bright Cluster Management Suite; also some modules load other modules. It is possible to make use of various versions of Intel compiler and parallel libraries by explicitly loading some of the those modules. By default the login environment loads gcc, and the batch scheduling system (slurm). One can list the modules actually loaded by issuing the command *module list*. Once you know which modules you need _module load_ <modulename>.

**Obtain Access**

- To obtain access to R2 users can request an account via Accounts & Access or send an email to researchcomputing@boisestate.edu. In your email indicate your PI, your project and what software you intend to use so that your account is enabled properly for the environment.
- To use the R2 cluster, direct your secure shell software (ssh) client to:
  - ssh -XC -c arcfour128 *username*@r2.boisestate.edu
- Requests for one-on-one training is strongly encouraged for anyone inexperienced using the HPC cluster. Contact researchcomputing@boisestate.edu for information regarding training.

**Storage**

Users’ home directories have a 50GB limit, with a 30TB scratch space. Scratch space is temporary storage, and while there are no restrictions HPC staff ask that you watch utilization. Scratch space is not backed up and is subject to clean-up if it gets too close to capacity.

**Running Jobs**

The R2 scheduler is SLURM.

**Sample submission scripts**
• To use SLURM (as in PBS), one creates a batch job which is a shell script containing the set of commands to run, plus the resource requirements for the job which are coded as specially formatted shell comments at the top of the script. The batch job script is then submitted to SLURM. A job script can be resubmitted with different parameters (e.g. different sets of data or variables). Please copy and edit the sample submission scripts that can be found under /cm/shared/scripts/slurm-examples. Lines beginning #SBATCH are directives to the batch system. The rest of each directive specifies arguments to the sbatch command. SLURM stops reading directives at the first executable line.

Submitting the job to the queuing system
• The command sbatch is used to submit jobs, e.g.
  ◦ sbatch submission_script
• The command will return a unique job identifier, which is used to query and control the job and to identify output.

Monitoring Jobs
• In SLURM, the command squeue shows what jobs are currently submitted in the queueing system and the command squeue -u <username> shows only those jobs belonging to that user.
• The command scontrol is a more powerful command allowing more detailed queries. E.g. to examine a particular job with id<jobid> in detail: scontrol show job <jobid>.

Deleting Jobs
• to cancel a job either running or still queuing use scancel: scancel <jobid>

Support
If you are a new compute user or if you’re looking for a general consultation, contact Research Computing by clicking Request Help. All other inquiries should be directed to the High Performance Computing support team at researchcomputing@boisestate.edu.