


Running Jobs, Submission Scripts, Modules

Matt Gitzendanner: magitz@ufl.edu

8/14/13

UF Information Technology www.it.ufl.edu

UNIVERSITY OF FLORIDA | High-Performance Computing




HiPerGator

The University of Florida Supercomputer for Research

UF Information Technology www.it.ufl.edu

UF Data Center

- ▶ UF Data Center on Eastside Campus
 - 10,000 sq.ft and 1.75 MW total
 - 5,000 sq. ft. space for Research Computing




UF Information Technology www.it.ufl.edu

HiPerGator

The University of Florida Supercomputer for Research

- 16,384 cores
- Infiniband interconnect
- 2.1PB fast, high-availability, storage
- Current cluster will be integrated—Fall 2013



UF Information Technology www.it.ufl.edu

UF Research Computing




How do I get my jobs started?

UF Information Technology www.it.ufl.edu

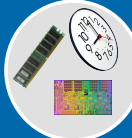
Cluster basics

User interaction




Login node (Head node)

Scheduler



Tell the scheduler what you want to do

Compute resources

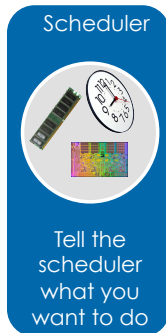


Your job runs on the cluster

UF Information Technology www.it.ufl.edu

Scheduling a job

- ▶ Need to tell scheduler what you want to do
 - **How many CPUs** you want and how you want them grouped
 - **How much RAM** your job will use
 - **How long** your job will run
 - The commands that will be run



UF Research Computing

- ▶ Ordinary Shell Script

```
#!/bin/bash
date
module load test_app
test_app -i file.txt
```

Read the manual for your application

Commands typed on the command line can be put in a script.

UF Research Computing

- ▶ Submission Script

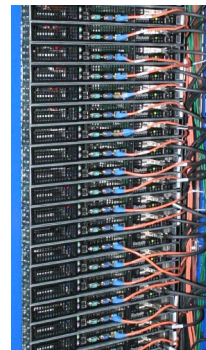
```
#!/bin/bash
#PBS -N My_Job_Name
#PBS -M Joe_Shmoe@ufl.edu
#PBS -m abe
#PBS -o My_Job.log
#PBS -e My_Job.err
#PBS -l nodes=1:ppn=1
#PBS -l walltime=00:05:00
#PBS -l pmem=900mb
```

```
cd $PBS_O_WORKDIR
date
module load test_app
test_app -i file.txt
```



Nodes and processors

```
#PBS -l nodes=1:ppn=4
#PBS -l nodes=2:ppn=32
```



HiPerGator

The University of Florida Supercomputer for Research

- **64 cores per node**
 - If RAM allows, MPI jobs under ~32 cores, should use nodes=1:ppn=##
- Current cluster will be integrated—Fall 2013
 - 4–8 cores per node on most, some 12 and 16



MPI

- ▶ Up to 32 cores: nodes=1:ppn=##
- ▶ Use the --bind-to-core flag
 - mpiexec --bind-to-core my_app ...
- ▶ If you use MPI a lot, talk to us
 - Several optimization flags

Most important question for any parallel application:
Does it scale?

RAM

#PBS -l pmem=900mb

- ▶ Lots to consider, but do your best at estimating RAM needed for job
- ▶ Over about 4GB of RAM, "costs" toward CPU allocation

Wasted RAM leads to idle CPUs and low job throughput



End-of-job emails: #PBS -M Joe_Shmoe@ufl.edu

#PBS -m abe

```
PBS Job Id: 358634.moab.ufhpc
Job Name: NR.25.nex
Exec host: c7a-s1/60
Execution terminated
Exit_status=0
resources_used.cput=07:16:09
resources_used.mem=251348kb
resources_used.vmem=318916kb
resources_used.walltime=07:16:52
```

RAM- bigmem queue

- ▶ For jobs asking for over 16GB per core (pmem)
- ▶ #PBS -q bigmem
- ▶ 1TB node



Processor Equivalents

- ▶ Accounts for large RAM requests
- ▶ Average 3-4GB RAM/core

1 core, 10GB RAM: ~3 PEs
1 core, 60GB RAM: ~18 PEs

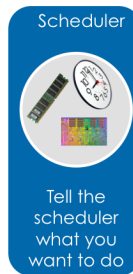
- ▶ Non-investor's limit: 8 PEs
- ▶ Investor limits are based on PEs



Walltime

#PBS -l walltime=00:50:00

- Fairly straight forward
- As with all resource requests, accuracy helps ensure **your** jobs and all other jobs will run sooner



	Maximum	Short	Long
Investor	31 days	<12 hrs	7 days
Other	7 days	<12 hrs	3 days

UF Research Computing

- ▶ Job Management
 - qsub <file_name>: job submission
 - qstat -u <user>: check queue status
 - showq -r -u <user>: shows job efficiency
 - qdel <JOB_ID>: job deletion
 - checkjob -v <job number> (shows PE value)

Lots of jobs

- ▶ What about running lots of jobs?



UF Information Technology

www.it.ufl.edu

Lots of jobs

- ▶ You can script your job submission, **BUT**:
 - How long will each job run?
 - Many short (<20 minutes) jobs are inefficient
 - Scheduling overhead



Pipettes only \$1.99 each! *

* Plus \$1.50 shipping per order

Would you order one at a time
or
place one order for 100?

UF Information Technology

www.it.ufl.edu

Lots of jobs

```
#!/bin/bash
#PBS -l nodes=1:ppn=1
#PBS -l walltime=00:05:00
#PBS -l pmem=900mb
```

```
cd $PBS_O_WORKDIR
date
module load test_app
test_app -i file1.txt
test_app -i file2.txt
test_app -i file3.txt
...
test_app -i fileN.txt
```

You can very easily
run multiple tasks in
a single job script

UF Information Technology

www.it.ufl.edu

Lots of jobs

- ▶ If you do submit lots of individual jobs:
 - **2,000-3,000** jobs maximum at a time
 - Add a ½ second pause between each job:


```
for i in $LIST
do
  qsub job.pbs -v INPUT=$i
  usleep 500
done
```
 - Consider how many jobs will run at once: what is your group's PE limit?
 - Will your group like you when you submit 2,000 jobs?

There is probably a better way...ask for help

UF Information Technology

www.it.ufl.edu

Some helpful environment variables

- ▶ `$PBS_O_WORKDIR` : the directory where you typed qsub
- ▶ `$PBS_JOBID` : the unique job id: e.g. 24461774.torx.ufhpc
- ▶ `$TMPDIR` : temporary directory for each job on compute node's local disk, good for jobs with lots of I/O
- ▶ `$PBS_NP` : Number of processors for single node job, use this when starting a threaded application to tell it how many processors to use. Prevents needing to change in multiple places. E.g. `nodes=1:ppn=4`, `bastn -num_threads $PBS_NP`
- ▶ `$PBS_JOBNAME` : Name you gave your job with #PBS -N

UF Information Technology

www.it.ufl.edu

So what is this "module" thing?

- ▶ **lmod**—Implementation of Environment Modules developed at TACC
- ▶ Allows easy management of user's environment



Lmod: Environmental Modules System

UF Information Technology

www.it.ufl.edu

The standard way

```
PATH=$PATH:/some/long/path/to/application
export $PATH
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/long/path/to/
place/I/probably/can't/find
export $LD_LIBRARY_PATH
```

- ▶ Need to track down paths to applications, libraries, etc.
- ▶ Multiple compilers, and MPI implementations
- ▶ Manage dependencies
- ▶ Multiple versions of apps



The module way

▶ module load trinity

▶ Automatically:

- Sets, \$HPC_TRINITY_DIR
 - To run Inchworm, simply type


```
inchworm --reads reads.fa --run_inchworm [opts]
```
- Loads Bowtie and Allpaths, two Trinity dependencies
 - You don't need to hunt those down, or worry if they are in your path or not

Module discovery

- ▶ `module spider`
 - List everything
- ▶ `module spider cl`
 - List applications that have cl in name
- ▶ `module spider amber/12`
 - List details about this version of AMBER
- ▶ `module key molecular`
 - Keyword search for applications

Multiple versions

```
[magitz@submit1 ~]$ module spider gaussian
Rebuilding cache file, please wait ... done
```

```
-----
gaussian:
-----
```

```
Description:
```

```
A software for electronic structure modeling
```

```
Versions:
```

```
gaussian/e01
gaussian/g03
gaussian/g09
-----
```

```
To find detailed information about gaussian please enter the full name.
For example:
```

```
$ module spider gaussian/g09
-----
```

Multiple variants of a version

```
[magitz@submit1 ~]$ module spider mrbayes/3.2.1
Rebuilding cache file, please wait ... Done
```

```
-----
mrbayes: mrbayes/3.2.1
-----
```

```
Description:
```

```
Bayesian inference of phylogeny
```

```
This module can be loaded directly: module load
mrbayes/3.2.1
```

```
Additional variants of this module can also be loaded
after the loading the following modules:
```

```
intel/2012, openmpi/1.6
```

Module loading

- ▶ `module load raxml`
- ▶ `module load intel raxml`
- ▶ `module load intel openmpi raxml`
- ▶ `module load intel/12 openmpi/1.6 raxml/3.2`
- ▶ `module unload raxml`

Module swapping

- ▶ `module load intel openmpi abyss`
- ▶ `module list`
Currently Loaded Modules:
1) intel/2012 2) openmpi/1.6 3) abyss/default
- ▶ `module swap openmpi/1.6 openmpi/1.5.5`
Due to MODULEPATH changes the following modules have been reloaded:
1) abyss

Basic commands

- ▶ `module spider`
- ▶ `module spider gaussian`
- ▶ `module avail`
- ▶ `module list`
- ▶ `module load clustalw`
- ▶ `module load python/2.6.5`
- ▶ `module add intel openmpi`
- ▶ `module load intel/12 openmpi/1.6 mrbayes`
- ▶ `module del/rm/unload clustalw`

Let's look at some examples

- ▶ Examples of job scripts in:
`/project/bio/training/2013-08-14/wordcloud/`
- ▶ Job scripts can have many commands
- ▶ `qsub` can pass variables into script with `-v` flag
`qsub my_script.pbs -v FILE=f1.txt,OUT=outdir/out1.txt,SIZE=5`

UF Research Computing

- ▶ Help and Support
 - Help Request Tickets
 - <https://support.hpc.ufl.edu>
 - For any kind of question or help requests
 - Searchable database of solutions
 - We are here to help!
 - support@hpc.ufl.edu



UF Research Computing

- ▶ Help and Support (Continued)
 - <http://wiki.hpc.ufl.edu>
 - Documents on hardware and software resources
 - Various user guides
 - Many sample submission scripts
 - <http://hpc.ufl.edu/support>
 - Frequently Asked Questions
 - Account set up and maintenance

