Getting on to hyrax/rhino

1. rhino (= ssh -X jayoung@rhino.fhcrc.org )

2. grabcpu (or grabnode). Timelimit seems to be 5 days as of Nov 20123.

if I DON'T want to use X-windows, can work directly on that, but if I do:

3. see what node comes up and use another window to ssh to that, e.g.

ssh -X jayoung@hyraxD01.fhcrc.org

4. close the first terminal window used to grabnode

(if I work on rhino nodes, night not need

(should I try ssh -tY instead? got from http://cbsresource.fhcrc.org/~mtmorgan/presentations/FHCRC-HPC-2009/HighPerformanceComputing\_lecture\_1.pdf , which seems useful)

working at home: getting disconnected?

http://phshelpdesk.fhcrc.org/index.php?/Knowledgebase/Article/View/6/12/ssh-i-am-getting-disconnected-while-im-in-the-middle-of-something

from rhino, to see what nodes I have grabbed:

**squeue | grep 'jayoung'**

to cancel an allocation:

**scancel <jobnumber>**

can specify groups of jobs, or "job arrays". e.g.

**15-690** is every job in that range.

**1-10:2** is jobs 1,3,5,7,9

**1,2,3,5-11**  etc

to check on a particular job:

**checkjob <jobnumber>**

might help diagnose why it's not running yet. but that doesn’t work any more – this works, though:

**scontrol show job myJobID**

this command gives worst case scenario on when a job might run:

**showstart <jobnumber>**

it's based on requested walltime of the running jobs, and many of them finish way earlier than that.

**wtb** : a command on rhino that shows that shows the load on various rhinos

queue information on all queues:

**sinfo -a**

to show what's running or waiting to run

**squeue -p queue\_name**

**squeue -u jayoung**

**squeue -A malik\_h**

**squeue -u jayoung -t PD**

-t is status = PD (pending)

to cancel only pending jobs

**scancel -u jayoung -t PD**

Reasons a job might not be running:

Resources = There aren't resources available to run the job

Priority = Higher priority jobs are queued ahead of this job

AssociationResourceLimit = The job would violate a per-user or per-account limit

QOSResourceLimit = The job would exceed a limit for the partition

ReqNodeNotAvail = A requested node configuration is temporarily unavailable

to show info on jobs INCLUDING completed jobs

**sacct -u jayoung**

**sacct -A malik\_h | more**

to show current per user / per account settings:

**sacctmgr show qos public format=name,maxtresperuser,maxtrespa**

a script to show who has what in the queue:

**hitparade**

show my usage recently:

**fhgizmojobs --csv --days-back 30 -u jayoung > ~/clusterUsageJan2014.csv**

timelimit column (walltime requested) is in minutes. default wall time is 2 days (48h = 2880min). duration (time taken) is in seconds.

could use this information to be smarter about how much wall time I'm requesting for each type of job. I would do that by including an option on sbatch, e.g.

**sbatch -t 1-0 bin/bowtie.basic.s**

-t: days - hours (so the above command asks for 1 day)

some kind of summary information on the queues

**sinfo -o "%P %C"**

example output:

PARTITION CPUS(A/I/O/T) #### allocated/idle/other/total

pubbase 1450/62/104/1616

publow 1450/62/104/1616

pubnorm 484/20/20/524

pubintbase 116/28/0/144

pubintlow 116/28/0/144

pubint 116/28/0/144

running in a specific queue

**srun -p queue\_name**

running on a specific node (I think - untested)

**srun --nodelist hyraxD84 --pty bash**

job information

**scontrol show job <job\_id>'**

info on how priorities are calculated:

https://teams.fhcrc.org/sites/citwiki/SciComp/Pages/Gizmo%20Workload%20Scheduling%20-%20Administrators%20Guide.aspx

can submit jobs in groups ("arrays") (I don't understand this yet)

**sbatch --array=1-5 bin/array.sh**

(then you get jobs with suffixes in the ID, e.g. 1050036\_1, 1050036\_2. Can cancel whole array (scancel 1050036) or just individual elements (scancel 1050036\_1)

tools to look at priority and "fairshare" (based on recent usage)

**sshare | head**

**sshare | grep 'jayoung'**

**sprio -u jayoung**

manual version of grabcpu (see email from Michael, July 19, 2012)

**srun -p campus -t 1-0 -c 1 --pty /bin/bash -il**

manual version of grabfullnode for 14 days

**srun -p campus -t 14-0 -c 12 --exclusive --pty /bin/bash -il**

-p is partition (campus versus tukwila)

-t is time: days-hours (there are other ways to specify, with : etc)

-c is number of cpus

-il maybe something to do with making it be an interactive login shell

for more on sbatch, see https://teams.fhcrc.org/sites/citwiki/SciComp/Training%20Materials/Introduction%20to%20Gizmo/gizmo-101.html#slide52

OLD:

From Dirk, April 20th, 2011:

Htop is nice (more user-friendly than top).

You would care about mem and swap. If mem is used up., swap will increase, swap is an extension of main memory on disk. If that rises too much your box will be very slow. Another interesting column is RES (for resident memory) It shows how much memory each of your processes is actually using right now.

> What I'm doing right now (as a test) is I started one Terminal window, ssh'd to rhino, reserved one whole node using grabnode (tells me I have exclusive access to 12 CPUs) - today that got me on to hyraxD02. Then I started a couple of other Terminal windows going and did "ssh -X hyraxD02" and started R (Rnew) jobs from there. I'm playing with R's multicore package.

> 1: is that the intended way to log in to a hyrax node when I want multiple CPUs?

Yes, if you want an interactive session that's the best way. The R multicore package is probably one of the best tools if you do not want to use more than 12 CPUs

> 2: is it greedy to do a "grabnode" and sit on that for a week or however long?

No, if your are using just one machine at the time that's just appropriate. You can also execute grabhalfnode which would give you 6 cpus

> 3: hyrax versus rhino - is there any difference? Should I care which one I first log in to?

Rhino and hyrax nodes are (almost) identical. We are working on making them fully identical. Most people logon to the rhinos and do their day to day development work on them and when they want to execute something more long running they spawn it off using the srun (like: srun Rscript yourScript.R > outputFile.txt & ) command which execute the program on the hyrax cluster. grabnode is really only for those users who want to have interactive control over their program.

> I'll probably start to get an intuitive idea of how much memory various types of data manipulations take, but perhaps I should also be monitoring "top". Whan I look at top's memory stats, is it "Mem" or "Swap" I care about (when I say "care" I mean in terms of making sure I don't do anything that kills my R process).

Hyrax versus mercury 64-bit nodes versus mercury 32-bit nodes

Asked them to switch me to tcsh on hyrax to match mercury nodes. Wait for that to happen before I mess with paths etc.

Can I put traskdata/bin\_linux\_64 in my path? And add path for PERL5LIB? And what is perl version?

I think so, IF I make sure path to R is maintained, OR I install my own R (which I think I might do)

OK: R is built. I will need these vars set up:

R\_HOME=/home/btrask/traskdata/lib\_linux\_64/R

R\_LIBS\_USER=/home/btrask/traskdata/lib\_linux\_64/R/library

Things that work on both (at least on initial test):

blastn etc

Jim Kent tools

Cross\_match, phred, phrap

RepeatMasker (with a warning about deprecation)

Bwa

Samtools

Things that don’t:

My R installation error while loading shared libraries: libgfortran.so.1: cannot open shared object file: No such file or directory

Consed error while loading shared libraries: libstdc++.so.5: cannot open shared object file: No such file or directory

Perl versions:

Notes from Oliver, when I was having some trouble and asked him about swap and memory (May 2011)

Linux tools: system monitor or activity monitor

Disk fragmenting

C-like languages heap biggest free chunk

Swap = it is writing to disk rather than using memory, so it is SLOW (I/O cost).

“dirty swap”. “thrashing the swap”

memory leaks (my trouble cam after I had been doing some pairs plots using smoothScatter and swap didn’t seem to be clearing out after the process finished).

Notes from conversation with Matt Fitzgibbon, May 20th 2011

general

/proc/cpuinfo is a useful file to look at to see how many CPUs, how much memory etc.

physical locations:

Hyrax is in Tukwila (copying can take a long time). Horton is a file server that’s also in Tukwila – might be possible to use that on hyrax nodes if copying or I/O is an issue, but might need sysadmin to set me up.

Rhinos here.

Mercury here.

Monitoring I/O: wait for Carl to complain!

scratch/tmp/local drives:

Hyrax - quite small scratch or tmp disks (e.g. 84Gb) – no equivalent to /loc/no-backup.

Rhinos - /loc/no-backup is big

Mercury - also ~50Gb scratch/tmp disk. If I log in using qsub there is an enviromental variable that tells me where my tmp dir is. It might be bigger.

Generally /tmp is one place to look for local storage but it is likely shared with swap and it would be REALLY BAD to fill it up.

cd /tmp

df –h . (or df –h)

on a mercury node: printenv | grep ‘scratch’ (e.g. on kong053 TMPDIR=/var/scratch/516463.hoppy where 516463 is my job number)

Batch jobs

to run on mercury, I’ve been using qsub, qstat, torque and MOAB.

hyrax/rhino uses slurm. srun (could get me an interactive shell, and is probably what’s behind grabnode), sbatch would run things as batch jobs. Try man sbatch.

Amazon: AWS cloud - data storage and transfer are expensive, but CPUs are cheap.

ISB cluster: 8c / core / hr (April 2013). Charge by CPU time, not walltime, so I/O shouldn't be charged. Limited free CPUs on a server that is often overloaded.