Yeti - Technical Information

- Architecture
- Fair share
- Job Queues
- Internet access
- Software
- Support
- Statistics
- Security

Architecture

The original Yeti cluster contained two submit nodes, a storage server, and 101 execute servers with a total of 1616 compute cores. These execute servers have dual 1.8 GHz Intel E5-2650L Xeon CPU with 8 cores each for a total of 16 cores per server.

Yeti Original Servers (E5-2650L CPU)

<table>
<thead>
<tr>
<th>Total Servers</th>
<th>Memory</th>
<th>Other Features</th>
</tr>
</thead>
<tbody>
<tr>
<td>38</td>
<td>64 GB</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>128 GB</td>
<td></td>
</tr>
<tr>
<td>35</td>
<td>256 GB</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>64 GB</td>
<td>Infiniband FDR network</td>
</tr>
<tr>
<td>4</td>
<td>64 GB</td>
<td>2 Nvidia K20 GPU modules</td>
</tr>
</tbody>
</table>

In February, 2015, 66 additional servers were added to the original Yeti cluster listed above, bringing the total server count to 167 execute servers with a total of 2762 compute cores. The CPU on all expansion machines is dual 2.6 GHz Intel E5-2650v2 Xeon.

Yeti Expansion Servers (E5-2650v2 CPU)

<table>
<thead>
<tr>
<th>Total Servers</th>
<th>Memory</th>
<th>Other Features</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>64 GB</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>256 GB</td>
<td></td>
</tr>
<tr>
<td>48*</td>
<td>64 GB</td>
<td>Infiniband FDR network.</td>
</tr>
<tr>
<td>5</td>
<td>64 GB</td>
<td>2 Nvidia K40 GPU modules</td>
</tr>
</tbody>
</table>

*Note: One of the 48 IB servers listed above has 256 GB RAM installed.

Fair share

Resource allocation on Yeti is based on each group’s contribution to computing cores. The Moab scheduler uses fair share targets and historical resource utilization to determine when jobs are scheduled to run. Within-group priority is also based on historical usage such that heavier users will have a lower priority than light users. Moab uses all of a job’s attributes - such as wall time, resource constraints, and group membership - to determine the order in which jobs are run. More and better information provided by users will help to increase the efficiency of the scheduler. Yeti
has a backfill policy that allow jobs to run out of order. Using job data such as walltime and resources requested, the scheduler can start other, lower-priority jobs so long as they do not delay the highest priority jobs. Because it works by essentially filling in holes in node space, backfill tends to favor smaller and shorter running jobs more than larger and longer running ones.

There is no preemption in the current system; a job in the queue will never interrupt or stop a job in run state.

The number of jobs that a user may have in the queue at one time depends on the requested resources. It varies between 4000 jobs for modest requests to only 100 to those specifying large memory and longer walltime.

**Job Queues**

On May 26 2015 the queue configuration was updated at the request of Yeti's Faculty Operating Committee. Understanding the new queue structure may have a significant effect on your productivity.

The following table shows the queues in Yeti and the limits associated with each of them. Queues batch0 to batch4 are assigned automatically by the scheduler, as is the interactive queue. The gpu and infiniband queues must be requested by the user.

<table>
<thead>
<tr>
<th>Queue</th>
<th>Max Walltime/Job</th>
<th>Max Memory/Job</th>
<th>Max Running Jobs/User</th>
<th>Max Queued Jobs/User</th>
</tr>
</thead>
<tbody>
<tr>
<td>batch0</td>
<td>2 hours</td>
<td>8 GB</td>
<td>512</td>
<td>4000</td>
</tr>
<tr>
<td>batch1</td>
<td>12 hours</td>
<td>8 GB</td>
<td>512</td>
<td>4000</td>
</tr>
<tr>
<td>batch2</td>
<td>12 hours</td>
<td>16 GB</td>
<td>128</td>
<td>400</td>
</tr>
<tr>
<td>batch3</td>
<td>5 days</td>
<td>16 GB</td>
<td>64</td>
<td>2000</td>
</tr>
<tr>
<td>batch4</td>
<td>3 days</td>
<td>no limit</td>
<td>8</td>
<td>100</td>
</tr>
<tr>
<td>interactive</td>
<td>4 hours</td>
<td>no limit</td>
<td>4</td>
<td>16</td>
</tr>
<tr>
<td>gpu</td>
<td>3 days</td>
<td>no limit</td>
<td>4</td>
<td>100</td>
</tr>
<tr>
<td>infiniband</td>
<td>48 hours</td>
<td>no limit</td>
<td>10</td>
<td>100</td>
</tr>
</tbody>
</table>

**Job Types**

User jobs on Yeti can be grouped into four broad categories: general batch, interactive, infiniband, and GPU. The rules for each vary and will be addressed separately.

**General Batch**

This is the catch-all category and might best be defined as "those jobs not belonging to any of the other three categories."

No special rules apply, but be aware that the systems available to run your job will increase if your walltime is lower.

Jobs with a walltime of 2 hours or less will be allowed to run on any node on the system. Walltimes of higher than 2 hours will result in your job being ineligible to run on GPU or Infiniband nodes in order to keep them available for jobs requiring specialized hardware.

Jobs with a walltime of greater than 12 hours will only be eligible for two types of nodes: those purchased by the user's own group and those not purchased by any other group. There are 52 such "public" nodes on the cluster, all first generation.

There are 5 batch queues (see table above). Your job will be automatically assigned to one of them. In general, the shorter and smaller (measured by memory) your job is, the more you can have running at one time.

**Interactive**

Specifying -I on your qsub command line will result in your job being automatically assigned to the interactive queue. No other special rules apply.

**Infiniband**
There is no requirement or expectation that all parallel jobs will request Infiniband nodes. Those that do not can be considered "General Batch" jobs, covered above, for the purposes of understanding how they are affected by the queue configuration.

The following three changes should be made to your submit script to request Infiniband nodes.

1. Add "#PBS -q infiniband" to request the infiniband queue.
2. Remove the core count, "ppn", from your "#PBS -l" directive. The infiniband queue requires that a job use an entire 16-core node. The inclusion of a ppn directive will probably prevent your job from starting.
3. Remove the memory requirement, "mem", from your "#PBS -l" directive. This is not a mandatory change but since entire nodes are being used there is usually little need to specify it. All Infiniband nodes have at least 64 GB of memory.

The following example requests 4 nodes, implying a total of $4 \times 16 = 64$ cores.

```bash
#PBS -q infiniband
#PBS -l nodes=4,walltime=6:00:00
```

Naturally, all jobs in the infiniband queue will only run on Infiniband nodes. To request that your job only run on the 2nd generation nodes with faster CPU, add the "ib2" flag to your request:

```bash
#PBS -q infiniband
#PBS -l nodes=4:ib2,walltime=6:00:00,mem=4000mb
```

The 'infiniband' queue is actually only a routing queue, and the jobs which specify it will actually end up in one of the following Yeti batch queues: ib2, ib12, or ib48. The number in the name of these queues indicates the maximum number of hours that the job is allowed to run. The maximum running jobs per user and the maximum queued jobs per user, as specified in the "Job Queues" section above, is the same for all 3 of these. The breakdown into subqueues was necessary to streamline the access and usage of Infiniband and GPU nodes according to the rules determined by the Yeti Operations Committee.

**GPU**

Jobs using GPU's should request the GPU queue. The following line should be added to submit scripts.

```bash
#PBS -q gpu
```

The 'gpu' queue is actually only a routing queue, and the jobs which specify it will actually end up in one of the following Yeti batch queues: gpu2, gpu12, or gpu72. This breakdown into subqueues is completely analogous to the 'infiniband' queue as described in the last paragraph of the previous section.

**Internet access**

By default, Yeti compute nodes have no outbound Internet access. However, there is a workaround which allows the use of certain protocols, namely HTTP, HTTPS, and FTP after a special module is loaded, like this:

```bash
module load proxy
```

For other protocols, like scp and rsync, one needs to run from the head node. However, please keep in mind that the head node is used by all other users and so it should not be abused with anything bigger than just occasional data transfers, typically for the purpose of installation and testing.

**Software**

- Operating system: Red Hat Enterprise Linux 6.8 (Santiago)
• Scheduler/Resource Manager: Torque/Moab
• Software applications including: Stata, Matlab, Stat/Transfer, R, and other software.

Support

HPC users who need support using the cluster should read the documentation provided on this site. If problems persist, please e-mail hpc-support@columbia.edu with your UNI, a full description of the issue including your submit script, any errors you may have received, and job ID numbers if applicable.

Statistics

CUIT expects to install a cluster monitoring tool called Ganglia. A link will be made available when it is ready.

Security

No security measures have been undertaken while setting up the Yeti cluster and so no support of sensitive data should be assumed. Any data that must not be seen by others will need to be analyzed in a more secure environment.