Torque
Why are my GPU nodes not being scheduled correctly?

Issue: Jobs requesting GPUs willl not run due to lack of GPU resources, yet the GPUs are available.

Cause: Most likely, Torque was compiled without the "--enable-nvidia-gpus" flag, or installed from the Adaptive-provided RPMs (running pbs_server --about will show if that's the case). The end result can be missing GPU information on the Torque server. To see if this is the case, look on the Torque server, in the <TORQUE HOME>/server_priv/node_usage/ directory, and examine the files for nodes with GPUs. You should see a section that looks something like this:

```
"numanode" :
{
     "allocations" : null,
     "cores" : "0-3",
     "gpus" : "0-1",
     "mem" : "8010632",
     "os_index" : 0,
     "threads" : ""
}
```

In the above example, the node has two GPUs. Make sure the file represents the correct number of GPUs. If it does not, then this article should provide the solution. If all of the "gpus" entries look correct, then look elsewhere (like the nvml libraries, for instance) as this solution will not fix anything.

Solution:

The best solution is to re-build Torque with the "--enable-nvidia-gpus" flag (if the nvml libraries were install in a non-standard location, the flags --with-nvml-include and --with-nvml-lib may also be required). After rebuilding and reinstalling the Torque server, and the MOMs if required, and before starting up Torque, remove ALL of the node files from the node_usage directory (Torque will recreate any of them that are missing).

As a workaround, those node_usage files can be manually edited, adding (or fixing) the "gpus" lines as needed.
 Unique solution ID: #1214
 Author: Rob Greenbank
 Last update: 2018-08-09 17:51