TORQUE Resource Manager 4.2.0 Early Access Release Notes

November 2012

The release notes file contains the following sections:

Overview

TORQUE 4.2.0 provides Intel Xeon Phi (MIC architecture) card support, introduces the ability to run a single job in two domains in a Cray system, supports starting and stopping services on a SLES system, and enhances preexisting features.

The New Features section provides more information about what TORQUE 4.2.0 has to offer.

New Features

Intel Xeon Phi (MIC architecture) card supported as new accelerator option

TORQUE can auto-detect the presence of MIC architecture cards when configured to do so. It can report metrics from them and allocate them to workloads. This feature requires the use of the Moab Workload Manager scheduler. See Scheduling accelerator hardware in the TORQUE Administrator Guide for more information.

Ability to run a single job in two domains added

TORQUE now supports multiple heterogeneous (multi-req) resource requests within a job for Cray systems. A job can request compute nodes both inside the Cray and outside of it. TORQUE manages the job on the Cray and non-Cray compute nodes.

max_user_queuable is now global

The server parameter max_user_queuable is now a system-wide parameter. Any configured value applies to all queues collectively. For example, if you set max_user_queuable to 5 previously, TORQUE would allow users to submit up to 5 jobs to each queue. If you set it to 5 now, users would be allowed to submit up to 5 jobs total across all queues.

SLES 11 (SP1/SP2) service management

You can now stop and start TORQUE services on a SLES system.

Known Issues

TORQUE 4.2.0 Early Access still occasionally experiences deadlock conditions. In most cases, this happens when users make extensive use of routing queues, job arrays, and/or job dependencies. Please report instances of deadlock to Technical Support if you encounter such.

- Deadlock occasionally occurs on queues (TRQ-1435).
- You may lose jobs if your server is stuck in deadlock (TRQ-1314).
- If multi-req jobs in a Cray system specify a hostlist, the ALPS reservation could fail. To avoid this problem, do not specify a hostlist for multi-req jobs (TRQ-1431).
- TORQUE may not clear jobs from the nodeboard if NUMA is enabled. Restart pbs_server when jobs are not cleared (TRQ-1426).
- If you restart with slot limits on TORQUE job arrays, slot limit holds may not reset properly (TRQ-1424).
- Moab Workload Manager occasionally receives "End of File" messages from TORQUE (TRQ-1399).
- Multi-node jobs may report resources incorrectly (TRQ-1222).
- Your system may crash if you have a high system load while using TORQUE job arrays (TRQ-1401).
- The momctl command may receive "End of File" errors. When this occurs, TORQUE tries to rerun momctl but may fail again. Manually run momctl again to solve this problem (TRQ-1432).
- If bad job array files exit at startup, pbs_server may segfault. If you encounter this behavior, move the offending .JB and .AR files out of the $TORQUE_HOME/server_priv/jobs and $TORQUE_HOME/server_priv/arrays directories, respectively (TRQ-1427).
- In rare cases, mother superior may not abort a job when a sister node goes down (TRQ-1396).
- Jobs that do not exist on the server may appear on the MOM in a running state (TRQ-1364).
- Jobs may not clean up correctly when you launch mpich2 job with OSC mpiexec (TRQ-1232).
- An incomplete environment variable could cause qsub to segfault. Prevent this by always submitting environment variables with a <name>=<value> pair. Avoid submitting <name>= or <name> only (TRQ-1125).
- At an exceptionally high load and while running many short jobs (under 30-second execution time), jobs may become stuck in a running state (TRQ-696).
- Client commands and API calls can take up to 5 times the pbs_timeout to expire if the destination times out each time (TRQ-1425).
- Deadlock can occur if no jobs can copy their output files back to pbs_server and there is a large number of jobs finishing rapidly. Verify that you have your system configured such that output files are delivered to their proper locations (TRQ-1447).
- In cases of system failures, such as the file system or network hanging, MOMs can become unresponsive. If this happens, restart TORQUE (TRQ-1433).
- Running qsub --version causes TORQUE to hang. Run qstat --version instead to avoid this problem.

System Requirements

The following software is required to run TORQUE 4.2.0:

- libxml2-devel package
- openssl-devel package
- ANSI C compiler (The native C compiler is recommended if it is ANSI; otherwise use gcc.)
- A fully POSIX make. If you are unable to "make" PBS with your make, we suggest using gmake from GNU.
- Tcl/Tk version 8 or higher if you plan to build the GUI portion of TORQUE or use a Tcl-based scheduler.
- If you use cpusets, libhwloc 1.1 or later is required (for TORQUE 4.0 and later)

Installation Information

The directions to install and configure TORQUE are in chapter 1 of the [TORQUE 4.2.0 Administrator Guide](#). Also note additional instructions in the PBS Administrators Guide and README.building_40.

Note that you may need to install libssl-dev in order for the source to make successfully. Specifically, the build system is looking for libssl.so and libcrypto.so. For non-RPM setups, you may need to make a symbolic link from the ssl and crypto libraries to the respective .so names.

Upgrading to TORQUE 4.2.0 and Backward Compatibility

TORQUE 4.2.0 is not backward compatible with versions of TORQUE prior to 4.0. When you upgrade to TORQUE 4.2.1, all MOM and server daemons must be upgraded at the same time.

The job format is compatible between 4.2.0 and previous versions of TORQUE. Any queued jobs will upgrade to the new version with the exception of job arrays in TORQUE 2.4 and earlier. It is not recommended to upgrade TORQUE while jobs are in a running state.

Because TORQUE 4.2.0 has removed all use of UDP/IP and moved all communication to use TCP/IP, previous versions of TORQUE will not be able to communicate with the components of TORQUE 4.2.0. However, all files in the /var/spool/torque ($TORQUE_HOME) directory and all subdirectories are forwardly compatible.

Documentation

The online help for TORQUE 4.2.0 is available in [HTML](#) and [PDF](#) format.

Changelog

Legend
c - crash
b - bug fix
e - enhancement
f - new feature
n - note

- b - Fix a security loophole that potentially allowed an interactive job to run as root due to not resetting a value when $attempt_to_make_dir and $tmpdir are set. TRQ-1078.
- b - Fix down_on_error for the server. TRQ-1074.
- b - Prevent pbs_server from spinning in select due to sockets in CLOSE_WAIT. TRQ-1161.
- e - Have pbs_server save the queues each time before exiting so that legacy formats are converted to xml after upgrading. TRQ-1120.
- b - Fix phantom jobs being left on the pbs_moms and blocking jobs for Cray hardware. TRQ-1162. (Thanks Matt Ezell)
- b - Fix a race condition on free'd memory when check for orphaned alps reservations. TRQ-1181. (Thanks Matt Ezell)
- b - If interrupted when reading the terminal type for an interactive job continue trying to read instead of giving up. TRQ-1091.
- b - Fix displaying elapsed time for a job. TRQ-1133.
- b - Make offlining nodes persistent after shutting down. TRQ-1087.
- b - Fixed a memory leak when calling net_move. net_move allocates memory for args and starts a thread on send_job. However, args were not getting released in send_job. TRQ-1199
- b - Changed pbs_connect to check for a server name. If it is passed in only that server name is tried for a connection. If no server name is given then the default list is used. The previous behavior was to try the name passed in and the default server list. This would lead to confusion in utilities like qstat when querying for a specific server. If the server specified was no available information from the remaining list would still be returned. TRQ-1143.
- e - Make issue_Drequest wait for the reply and have functions continue processing immediately after instead of the added overhead of using the threadpool.
- c - tm_adopt() calls caused pbs_mom to crash. Fix this. TRQ-1210.

- b - Array element 0 wasn't showing up in qstat -t output. TRQ-1155.
- b - Cores with multiple processing units were being incorrectly assigned in cpusets. Additionally, multi-node jobs were getting the cpu list from each node in each cpuset, also causing problems. TRQ-1202.
- b - Removed some ambiguity in the for loop of send_job_work around svr_connect and svr_disconnect. We were checking the handle for positive values but never setting it negative after calling svr_disconnect. Potential race condition to inadvertently close this file in multi-threaded environment.
- b - Finding subjobs (for heterogeneous jobs) wasn't compatible with hostnames that have dashes. TRQ-1229.
- b - Removed the call to wait_request the main_loop on pbs_server. All of our communication is handled directly and there is no longer a need to wait for an out of band reply from a client. TRQ-1161.
- e - Modfied output for qstat -r. Expanded Req'd Time to include seconds and centered Elap Time over its column.
- b - Fixed a bug found at Univ. of Michigan where a corrupt .JB file would cause pbs_server to seg-fault and restart.
- b - Don't leave quotes on any arguments passed to the resource list. TRQ-1209.
- b - Fix a race condition that causes deadlock when two threads are routing the same job.
- b - Fixed a bug with qsub where environment variables were not getting populated with the -v option. TRQ-1228.
- b - This time for sure. TRQ-1228. When max_queuable or max_user_queuable were set it was still possible to go over the limit. This was because a job is qualified in the call to req_quejob but does not get inserted into the queue until svr_enquejob is called in req_commit, four network requests later. In a multi-threaded environment this allowed several jobs to be qualified and put in the pipeline before they were actually commited to a queue.
- b - If max_user_queuable or max_queuable were set on a queue TORQUE would not honor the limit when filling those queues from a routing queue. This has now been fixed. TRQ-1088.
- b - Fixed seg-fault when running jobs asynchronously. TRQ-1252.
- b - Job dependencies didn't work with display_server_suffix=false. Fixed. TRQ-1255.
- b - Don't report alps reservation ids if a node is in interactive mode. TRQ-1251.
- b - Only attempt to cancel an orphaned alps reservation a maximum of one time per iteration. TRQ-1251.
- b - Fixed a bug with SIGHUP to pbs_server. The signal handler (change_logs()) does file I/O which is not allowed for signal interruption. This caused pbs_server to be up but unresponsive to any commands. TRQ-1250 and TRQ-1224
- b - Fix a deadlock when recording an alps reservation on the server side. Cray only. TRQ-1272.
- c - Fix mismanagement of the ji_globid. TRQ-1262.
- b - Fixed a problem in the job rerouting thread where two threads could be running at the same time while rerouting jobs from a routing queue and causing jobs to abort. The result of this behavior made it so pbs_server could not be shut down with a SIGTERM or SIGHUP. TRQ-1224
- c - Setting display_job_server_suffix=false crashed with job arrays. Fixed. bugzilla #216
- b - Restore the asynchronous functionality. TRQ-1284.
- e - Made it so pbs_server will come up even if a job cannot recover because of a missing job dependency. TRQ-1287
- b - Fixed a segfault in the path from do_tcp to tm_request to tm_eof. In this path we freed the tcp channel three times. the call to DIS_tcp_cleanup was removed from tm_eof and tm_request. TRQ-1232.
- b - Fixed a deadlock which occurs when there is a job with a dependency that is being moved from a routing queue to an execution queue. TRQ-1294
- b - Fix a deadlock in logging when the machine is out of disk space. TRQ-1302.
- e - Retry cleanup with the MOM every 20 seconds for jobs that are stuck in an exiting state. TRQ-1299.
- b - Enabled qsub filters to be accessed from a non-default location. TRQ-1127
- b - Put the ability to write the resources_used data to the accounting logs. This was in 4.1.1 and 4.1.2 but failed to make it into 4.2.0. TRQ-1329
- b - Moved record_job_as_exiting from req_jobobit to on_job_exit_task so the job has a chance to move through its exiting routines before the "cleanup stuck exiting jobs thread" tries to remove them. This prevents a deadlock when on_job_exit and the cleanup thread try to run at the same time. I also changed the time comparision in check_exiting_jobs to use like units for the time comparison. TRQ-1306
- b - Fixed a deadlock caused by queue not getting released when jobs are aborted when moving jobs from a routing queue to an execution queue. TRQ-1344.
- c - Fix a double free if the same chan is stored on two tasks for a job. TRQ-1299.
- b - Changed pbs_original_connect to retry a failed connect attempt MAX_RETRIES (5) times before returning failure. This will reduce the number of client commands that fail due to a connection failure. TRQ-1355

- b - Fix the proliferation of "Non-digit found where a digit was expected" messages, due to an off-by-one error. TRQ-1230.