

Moab HPC Suite 9.0.3 Release Notes

The release notes file contains the following sections:

- New Features
- Differences
- Installation and Upgrade Information
- Known Issues
- Resolved Issues

New Features

This section contains a summary of key new features in Moab HPC Suite.

In this section:

- General Suite
- Moab Workload Manager
- Moab Accounting Manager
- Moab Web Services
- Moab Insight
- Moab Viewpoint
- Torque Resource Manager

General Suite

This section contains information applicable to more than one of the components in the Moab HPC Suite.

9.0.3

No new features.

9.0.2

No new features.

9.0.1

Client Authentication Now Supports SHA-512

Moab Workload Manager and Moab Web Services now support SHA-512 for message digests and message authentication codes. This support is in addition to the existing support for SHA-1.

- Moab Workload Manager has a new server checksum algorithm parameter named SERVERCSALGO. Valid values:
 - HMAC64: the default (SHA-1)
 - HMACSHA2: more secure (SHA-512)
- Moab Web Services has a new configuration parameter named moab.messageDigestAlgorithm. Valid values:
 - SHA-1: the default
 - SHA-512: more secure

 If SERVERCSALGO is set to HMAC64, then moab.messageDigestAlgorithm must be set to SHA-1. Likewise, if SERVERCSALGO is set to HMACSHA2, then moab.messageDigestAlgorithm must be set to SHA-512.

9.0.0

Offline Install Support

Instructions on how to prepare for an offline install are now available for configurations where clusters are not connected to the Internet.

Moab Workload Manager

9.0.3

Ability to Requeue a Job Asynchronously

Jobs can be requeued asynchronously with `mjobctl -R --flags=force`.

Added Node Count Per User Option

`showq -u <username>` now displays a node count for the user.

Pass Job ID to Elastic Trigger Script

Elastic scripts have access to a `$JOBID` variable to obtain the job ID from Moab.

Dynamic Assignment of Classes to Nodes (MOAB-8708)

Node class assignments formerly persisted until Moab was restarted. Class assignments can now be removed from nodes using `mnodectl -m cfgclass=-<CLASSES>`.

Improved GPU Support With Slurm Integration

Changes to `msub` better support assigning GPUs to submitted jobs.

Reservation Profiles Enabled for Standing Reservations

A reservation profile for a standing reservation can be specified using `SRCFG[] PROFILE=NAME`.

Support for More Commands With UIMANAGEMENTPOLICY FORK

To reduce processing time, when UIMANAGEMENTPOLICY is set with FORK, Moab creates new processes to handle requests for specific commands. In addition to the previously supported commands, Moab now supports `showstate`, `showstats`, and `mdiag -a|-c|-g|-q|-r|-S|-s|-t|-u`.

Can Turn Off Backfill for Nodes With a Suspended Job

New Features

Setting the SUSPENDEDJOBNOBFINELIGIBLE flag turns off backfill for nodes with a job suspended due to preemption. No other jobs are scheduled, allowing the preempted job to resume after the preemptor job is completed.

mdiag -j -v Reports Total Job Memory

mdiag -j reports memory per task or process. mdiag -j -v additionally reports total job memory.

9.0.2

New msub -n Option

msub -n is added to support node exclusivity.

createdDate Added to Block Messages

A createdDate has been added to the job.blocks array for MWS and Insight.

fullEnvironmentVariableList Job Attribute

Moab now provides a fullEnvironmentVariableList job attribute to MWS and Insight that includes environment variables not included in the environmentVariables field.

Commands to Display the Reason Jobs Are Blocked Or Deferred

showq -b -v and *showq -b -v --blocking* display the reason jobs are blocked or deferred.

Messages Showing a Resource Manager is Down Are Only Displayed to Admins

When a resource manager is down, the *showq* command only reports that the RM is down to admin users.

9.0.1

Allow for Specifying USEMOABJOBID on a Per Job Template Basis

Enabled new job flag "USEMOABJOBID" for job submission and job templates.

Administrator Policy for Max Preemption Count

The JOBMAXPREEMPTCOUNT policy can be set to specify the number of times a single job can be preempted.

showq Filter for Feature

Enable "showq -w nodefeature=" to filter on node feature strings.

Ability to Require One of Several in a Reservation

Enable REQUIREDUSERLIST and REQUIREDACCTLIST for SRCFG.

Command to Refresh LDAP Credentials Cache

`mcredctl -r uid user:<UID>` refreshes a user's UID and GID.

9.0.0

Basic Docker Support

Customers can now run their serial workloads inside docker containers. Docker containers provide an isolated environment with the correct linux distribution version of all the libraries the user needs to run the workload. System administrators can preset the containers or users can create their own images and have the administrator upload their images to a central repository so the users can create containers from them. You can also configure job templates to force workloads and/or users to run inside Docker containers, as well as running preemptable or interactive jobs in containerized environments.

Moab and NUMA-Aware Scheduling

Moab now works with Torque to support "NUMA-aware" scheduling and task placement. Moab schedules and Torque places tasks in such a manner that they have exclusive access to their requested resources so they can execute as fast as possible in a NUMA-based hardware environment, which Torque enforces by pinning resources to a task using a Linux control group ("cgroup").

NUMA-awareness allows Moab to schedule tasks (e.g. MPI "ranks", individual processes, etc), regardless whether homogeneous or heterogeneous, to hardware resources based on their characteristics, and permits Torque to place the tasks on the node hardware in such a manner as to promote their fastest possible computation speed (fast local memory accesses for the core's socket/NUMA node versus slow remote memory accesses outside a core's socket/NUMA node).

When a Torque pbs_mom starts on a compute node, it now uses the "hwloc" library to query for information about the node's internal hardware architecture; (i.e., the number of sockets within the node, NUMA nodes within a socket, cores within a socket or NUMA node, threads within a core, memory regions, and PCIe-based accelerators). It also determines which memory regions and accelerators are attached to which sockets or NUMA nodes. Torque retains this information for later task placement and passes most of the information to Moab so it becomes aware of each node's internal architecture for scheduling tasks.

i The NUMA-aware system functionality is considered in Beta and works for generic x86 systems. It has not been designed to work with Cray ALPS systems that use the "aprun" command. Moab and Torque will not rewrite the "aprun" command line syntax within a job script to match a Moab/Torque NUMA-aware job submission.

Retry Failed Job Charges

The AMCFG[] RETRYFAILEDCHARGES parameter can be enabled to automatically retry job charges that fail due to a connection failure between Moab and the accounting manager (this is the default). This policy can be configured to retry every failed connection charges up to a limit of AMCFG[] CHARGERETRYCOUNT times. A script (mam-charge-retry.pl) has been provided to modify and

retry charges that have failed for non-connection reasons. See Accounting, Charging and Allocation Management in the *Moab Workload Manager Administrator Guide*.

Separate Handling for Accounting Connection Failure Actions

The accounting failure actions now allow the specification of a separate action to be used if there is a connection failure between Moab Workload Manager and Moab Accounting Manager. The AMCFG[] **CONTINUEFAILUREACTION**, **CREATEFAILUREACTION**, **RESUMEFAILUREACTION**, and **STARTFAILUREACTION** parameters now permit values of the form <GeneralFailureAction>[,<FundsFailureAction>[,<ConnectionFailureAction>]].

Ability to Specify a Minimum Size Before the Job is Eligible For Priority Reservation

A new MINPRIORITYJOBRSVSIZE server parameter is available to define the minimum total job size (in processors) for jobs that can get a priority reservation. Jobs smaller than the specified value will still be started during normal and backfill scheduling, but will not be eligible for priority reservations. Default is 0.

Change the Requested Tasks per Node for a Job

Enabled "mjobctl -m tpn=X" for modifying tasks per node.

Display Job Dependencies Even After the Job Began Running

A new "showcompleteddependencies" SCHEDCFG flag is available to show dependencies on a job even after the dependencies have been satisfied.

High-Availability Trigger

Enabled "failure" scheduler trigger for recovering in HA scenarios.

Rebuild Standing Reservations While Moab is Running

mrsvctl now includes an option (-B <SRVID>) to enable you to rebuild/refresh standing reservations while Moab is running.

Query Details for Jobs that Have Already Terminated

Enabled checkjob ALL --flags=COMPLETE to obtain checkjob information for every job, including completed jobs.

Ignore Hostlist Requirements on Jobs

Added CLASSCFG[] IGNHOSTLIST=TRUE to ignore hostlist requirements on jobs.

Diagnose License Information

A new "-l" argument is added to mdiag to let you diagnose license information contained in the moab.lic file.

Support for "CLASS:<name> REQUIREDUSERLIST=<user>" in the Identity Manager

The REQUIREDUSERLIST parameter was added to let you dynamically set the user list with a class for jobs.

i Removing a user from the REQUIREDUSERLIST will not affect the user's running jobs. However, the user's idle jobs will become blocked because the user no longer has access to the class requested.

Cancel a Job Workflow

Added "mjobctl -c flags=follow-dependency <job_id>" to let you cancel all jobs that the specified <job_id> depends on. This is different from the CANCELFAILEDDEPENDENCYJOBS scheduler flag which automatically cancels jobs that depend on the <job_id>.

Node Features Can Be Shared by the Same Class

NODEACCESSPOLICY now supports the SINGLECLASS attribute.

Manually Write Out the Checkpoint File

mschedctl has a new "-W" flag to manually write out the checkpoint file.

Ability to Modify the Node Access Policy for a Queued Job

mjobctl -m has a new "nodeaccess=" attribute to let you modify the node access policy for a *queued* job.

Specify a From Address for Emails Sent from Moab

The "MAILFROMADDR" Moab parameter is added to let you set the from address for all emails sent from Moab. This is used in conjunction with the MAILPROGRAM Moab parameter.

Additional CLASSCFG[] Attributes for Class Remapping

Enable CLASSCFG[] MAX.TPN and MIN.TPN for class remapping.

Limit Which "mdiag" Options the User Can Run

Enabled "USERCFG[] PRIVILEGES=<SCHED|RM|NODE>:diagnose" to let you specify which mdiag options the user may run.

New COMPLETIONCODE Variable for Trigger Scripts

When used in conjunction with the config parameter "JOBCFG" and a trigger event type of "end", COMPLETIONCODE provides the trigger with the return code of the job.

New PREEMPTIONALGORITHM Parameter

New Features

The `PREEMPTIONALGORITHM` is added to designate how Moab handles preemption scheduling policies. Valid values are `PREEMPTORCENTRIC` or `PREEMPTTEECENTRIC`. `PREEMPTTEECENTRIC` is the default.

- `PREEMPTORCENTRIC` – Moab uses the normal scheduling policy and obeys all configured policies (such as `JOBNODEMATCHPOLICY`, `NODEALLOCATIONPOLICY`, `NODEACCESSPOLICY`). Previously, Moab did not support those policies for preemption.
- `PREEMPTTEECENTRIC` – Moab uses the custom scheduling policy that ignores many policies to ensure the fewest and least important (by priority) preemptees are disturbed by the preemptor.

Add Environment Variables to Jobs

A new template extension attribute (`ENV`) is available to add specified job environment variables to the job.

Moab Accounting Manager

9.0.3

No new features.

9.0.2

No new features.

9.0.1

No new features.

9.0.0

Gold References Removed from MAM

References to Gold have been replaced with references to MAM throughout the MAM distribution. This created changes to client and server command names, configuration files, log files, etc. See [Differences](#) for more details about these changes.

MAM can be configured via a configure option (`--with-legacy-links`) to create symbolic links to the old server and client command names. When running a command under its old name, the command will issue a deprecation warning. This warning can be disabled by setting `client.deprecationwarning = false` in the `mam-client.conf` file.

Summary Reporting

Aggregate values are now displayed for certain client commands to support summary reporting. The `--show` parameter has been augmented to support Sum, Average, Count, Unique, Min, Max, and

New Features

GroupBy operators for the mam-list-usagerecords, mam-list-transactions, mam-list-allocations, mam-list-funds, mam-list-liens, mam-list-quotes, and mam-balance client commands.

Moab Web Services

9.0.3

No new features.

9.0.2

No new features.

9.0.1

#PBS Requirements Passed to Job

#PBS directives included inline with job scripts are processed server-side. This means that #PBS requirements will be passed to a job, provided they do not conflict with other defined job requirements in the web service call.

Health Check for the Insight Server

MWS has a new health check for the Insight Server. The affected calls are /rest/diag/health/summary and /rest/diag/health/detail.

New Resource to Retrieve License Information

MWS contains a new resource, /rest/diag/licenses, that retrieves license information from Moab Workload Manager.

New RLM License Plugin

The RLM plugin for MWS is new in version 9.0.1. This plugin polls a Reprise License Manager (RLM) for purchased and available licenses for a given independent software vendor (ISV) and product. It reports this license information to Moab Workload Manager as a resource on the GLOBAL node.

9.0.0

Additional Job Parameters

MWS now lets you get job priority information (priority-analysis) and information about the job's eligibility (job-analysis) to run on the nodes managed by Moab.

Moab Insight

9.0.3

No new features.

9.0.2

Archived Node Reliability and Retention

Insight now stores archived nodes in the Insight MongoDB database. By default, nodes will remain in the MongoDB database two years past the last time they have been reported by Moab Workload Manager. However, a new parameter (`archiver.nodes.expireAfterSeconds`) is available that lets you specify when the nodes will be deleted from the MongoDB database (to save disk space).

Archived Reservation Reliability and Retention

Insight now stores archived reservations in the Insight MongoDB database. By default, reservations will remain in the MongoDB database two years past their end date. However, a new parameter (`archiver.reservations.expireAfterSeconds`) is available that lets you specify when the ended reservations will be deleted from the MongoDB database (to save disk space).

Performance Improvement in Job Sample Processing

Job samples are now generated considerably faster.

Insight Provides Diagnostics Call

Insight provides a diagnostics call that MWS can access from its Health page.

9.0.1

Improved Archiver and Failure Handling

Insight more easily recovers from downtime. The data archiving is more efficient and a process is in place to better recover from failures. A new clean-up script is also available to clean orphan jobs or nodes.

Archived Job Reliability and Retention

Insight now stores archived jobs in the Insight MongoDB database. By default, jobs will remain in the MongoDB database two years past their completion date. However, two new parameters (`archiver.jobs.expireAfterSeconds` and `archiver.job_state_journal.expireAfterSeconds`) are available that let you specify when the completed jobs will be deleted from the MongoDB database (to save disk space).

9.0.0

Moab Insight is the centralized, relational database available with Moab HPC Suite. Insight keeps track of historical Moab data for reporting purposes (Crystal Reports or Moab Viewpoint). For Moab Viewpoint use, Insight also interfaces with a Mongo cache database for sub-second targeted queries.

Insight must be installed on its own host (Insight Server Host).

Moab Viewpoint

9.0.3

Simplified Role Management

Viewpoint's role management page has been greatly simplified. Admins no longer have to specify domain permissions and can now use a tree navigator to more easily manage their user's permissions.

9.0.2

Colors Show Status of Nodes in Node List

The status of nodes in the node list is indicated by color.

Workload Tables Show Job Status by Color

The queue status of jobs in the workload tables is indicated by color.

Application Templates Can Request Malleable Cores

Application templates allow users to request malleable core jobs (relates to Moab's TRL parameter).

Allocated Node List Sorted and Role Indicated by Color

The allocated node list in job details page has been improved to reflect actual allocation order, as well as adding coloring so users can more easily determine the role of each of the nodes allocated to the user's job.

Job Messages Include Timestamp

Job messages on the Job Details page display a date/time stamp to help users determine if the message information is relevant.

Application Templates May Specify Total Memory or Memory Per Core

Application templates allow users to request either total memory for the job or memory per core.

File Manager Configuration Page Includes a Connection Validation Test

The configuration page for File Manager includes Test button that verifies whether the server parameters the user entered are correct.

Search by Job Name on Workload List Page

On the Workload List page, users can search for jobs by job ID, job name, or submitter name.

Sessions Page Toggles Between List and Grid Views

The Sessions Page allows users to toggle between list and grid views of remote sessions.

Application Templates Allow Users To Select an Operating System at Submission Time

The Advanced Settings section of the Application Template page allows users to request an operating system at job submission time.

9.0.1

Remote Visualization Support

Adaptive Computing now offers a remote visualization solution for your scheduling needs. Viewpoint users can now easily submit and manage remote visualization sessions directly from their web browser and Moab handles the scheduling remotely.

Nitro Support for Viewpoint

Viewpoint has been extended to allow Nitro jobs to be submitted and managed directly from the web portal. Nitro application templates streamline and simplify their user's job submission processes by providing administrators with the tools they need to make job submission easier than ever.

New Application Template Features

Viewpoint Application Templates have significantly been updated. The following is some of the new features available with the 9.0.1 release:

- Version support. Now every time an application template is saved, it creates a new version. The ability to restore or export prior application template versions is also supported.
- Additional application template widgets. Several new widgets are added. Administrators can specify whether custom widgets are required or optional.
- Performance enhancements. Now you can multi-select operations on your application templates such as export, publish/unpublish, copy, and delete.
- Node policies are now exposed in application templates.
- Added application templates types (HPC Application, Job Array, Remote Visualization and Nitro Application). Each type has its own icon associated with to enable it to be easily identified from other application template types.

File Manager Improvements

New Features

The file manager now behaves like a traditional file browser, which allows support for common key shortcuts. You can also preview the header and footer of your files. The file manager now provides features to assist in file organization such as pages instead of a single list of files.

License Support

You can specify the use of the features you care about the most. The provided license manager software allows for easy ways to view additional product functionality and the tools to license these features for use.

9.0.0

Moab Viewpoint is a web application that, interacting with Moab Workload Manager, allows administrators and users to manage jobs and resources without the complexities of maintaining Moab via the command line. Viewpoint also enables the creation of application templates for submitting jobs and connection to a remote file system directly from the browser so that users can easily import and save files (for example, job submission scripts or standard output or error files).

Currently, Viewpoint contains two objective-based portals. The Admin Portal and the User Portal (also referred to as the Job Submission Portal). Viewpoint uses principals and roles to let you customize which pages, tools, and settings certain users or groups are permitted to use, manage, and view. For example, you can grant a user the ability to see that application template used to submit the job, but not edit the application template.

Torque Resource Manager

6.0.3

Torque Setup Script Improvements (TRQ-3866)

The `torque.setup` script now permits specification of a config path (default is the path used when configuring Torque). New options have also been added to specify a prefix path (path to executables is in `PREFIXPATH/bin` and `PREFIXPATH/sbin`) and a new option has been added to log errors to `syslog`.

pbs_mom Switch to Force the Server to Accept a Hardware Update

`pbs_mom -f` forces the server to accept an update of the hardware on the node. You should use this switch the first time you start the MOM after an upgrade.

Added default_gpu_mode Server Parameter

The `default_gpu_mode` server parameter sets the default value GPU mode. This should facilitate using CUDA 8.

Simplified Torque RPM Installation

The RPM installation no longer writes the server name into pbs_mom's config file, eliminating the need to configure the MOM config file in the default case.

Simplified Installation Process

"make install" and the package installer now install service/unit files, set up the path to the Torque libraries, and establish a shell path to binaries for sh and csh shells.

Resource Plug-In

There is now an API for creating resource plug-ins that report custom varattrs, generic resources, generic metrics, and features. Additionally, jobs can report custom resources through the plug-in.

-L Submission Recorded in the Accounting Log

Jobs submitted with the -L request syntax will now have the -L submission recorded in the accounting log.

Change hashname for a Torque Job to Match the Corresponding Moab Job ID

When jobs are submitted using qsub -J, the job's hashname is set to the job ID of the Moab job.

qstat -f Reports Slot Limit For Arrays

A slot limits for an array job is now shown in "qstat -f" output under the job_array_request attribute. A hold type due to a slot limit for an array job is now shown under the Hold_Types attribute ("I" hold type means the hold is due to a slot limit).

MOM Parameter to Overwrite Output Files

The \$force_overwrite MOM parameter, if set to true, forces the output files to be overwritten each time a job is started.

Option to Turn Off trqauthd Logging

Added a -n option to trqauthd that disables trqauthd from logging anything.

6.0.2

Recover Array Subjobs

The new server parameter "ghost_array_recover" is added. pbs_server will now recover array subjobs even when the array (.AR file) couldn't be recovered. This parameter is set to TRUE by default.

Cray-Enabled Torque May Also Be Configured with cgroups

Support is added for Cray-enabled Torque configured with cgroups.

New Features

- On the login node, each job will have all of the CPUs and all of the memory controllers in its cgroup.

Epilogue Script Runs, Even if Output Files Cannot Be Appended

The epilogue script will now run when `spool_as_file` is configured, even if error and output files are not available and cannot be appended.

Job Script Path is an Argument to Prologue and Epilogue Scripts

A new positional parameter contains the full path of a job's job script to the job's prologue and epilogue scripts when the Torque `pbs_mom` "mother superior" launches the scripts. The prologue script is the new 8th positional parameter. The epilogue script is the new 11th positional parameter.

User May Request No Emails be Sent, Even on Failure

The `qsub -m p` option makes it so no emails are sent for the job, even on failure.

Torque Commands Appear in User's Login Shell Path

RPMs created via `build-torque` now create `/etc/profile.d/torque.sh` (and `torque.csh`) files so that Torque commands appear in a user's login shell path.

Option to Email User on All Non-zero Exit Codes

The `qsub -m f` option sends an email if a job has a non-zero exit code. The `qsub -m f` option can be used with the `a`, `b`, and `e` options, but not with `n` or `p`.

Added an Option for When to Send Mail Notifications

Added a `-m f` option to `qsub` to send mail when a job terminates with a non-zero exit code.

Option to Disable Reading of RUR Information

Added `$cray_check_rur` configure option to disable reading of Resource Utilization Reporting (RUR) energy usage for Cray login nodes. If set to `false`, login MOMs will not look at the energy resource information used for each job. Disabling this may improve performance.

New "email_batch_seconds" Server Parameter

The new server parameter "email_batch_seconds" lets you control at what frequency a batch of emails are sent to each user.

6.0.1

User settable kill_delay Through qsub -K Option

Added a user settable, per-job kill delay, called `kill_delay`. It is settable via the *new* `qsub -K` option.

Added gres_modifier Server Parameter

New Features

Gives permission to a list of users to modify the gres resource of their own running jobs.

Untrusted Host Vaidation

Added the ability to trust certain users or groups from hosts without allowing all users from those hosts to submit jobs.

Added ghost_queue Queue Attribute

When pbs_server restarts and recovers a job but cannot find that job's queue, it will attempt to recover the job by creating a ghost_queue for that job.

Added Process Adoption Through pbs-track

Added the ability to adopt running processes into a job with pbs_track.

6.0.0

cgroup Support

Torque is enhanced to create one Linux control group (cgroup) per task based on the new NUMA-aware, task-based job submission option (-L) and to create one cgroup for all tasks of a job on the same compute node for the older job-based option (-l). Torque uses cgroups to manage CPU and memory accounting, enforce memory usage limits, set up CPUset management, and bind cores/threads, memory, and accelerators, such as GPUs and MICs, to jobs.

When binding resources that include an accelerator to a task, Torque will make a best-effort attempt to place a task on the cores/threads and memory of the socket/NUMA node to which the accelerator attaches.

Ability to Prevent Nodes Being Dynamically Edited

A new qmgr parameter is available. When 'dont_write_notes_file' is set to true, the nodes file cannot be overwritten for any reason; qmgr commands to edit nodes will be rejected. The default is FALSE.

Execute the Job Starter Script with Elevated Privileges

The '\$job_starter_run_privileged' MOM configuration parameter is added and lets you specify whether Torque executes the job starter script with elevated privileges. The default is FALSE.

Differences

This section contains differences in previously existing features that require a change in configuration or routine.

In this section:

- General Suite
- Moab Workload Manager
- Moab Accounting Manager
- Moab Web Services
- Moab Insight
- Moab Viewpoint
- Torque Resource Manager

General Suite

This section contains information applicable to more than one of the components in the Moab HPC Suite.

9.0.3

LOGFILEMAXSIZE may be specified in MB or GB

The default unit for the LOGFILEMAXSIZE parameter is bytes, but it can also be specified in mb, MB, gb, or GB.

9.0.2

No known differences.

9.0.1

No known differences.

9.0.0

Multiple-Host Configuration

The installation process now provides better focus and support for multiple-host configurations. This includes changes to RPM installs to reduce dependencies between the different suite components. The installation documentation has also been updated.

Moab Workload Manager

9.0.3

Check for Circular Dependencies No Longer Default

Moab formerly checked job dependencies for circular dependencies. This check is now only performed if the SCHEDCFG CHECKCIRCULARDEPENDENCIES flag is set.

Moab Handles qsub Job Arrays the Same as msub

Moab now understands Torque job arrays the same as its own job arrays. Job arrays submitted to Torque will show up as job arrays in Moab.

9.0.2

EPEL Installation Changes for RHEL 6 and 7

Installation of the EPEL RPM repository for Red Hat Enterprise Linux (RHEL) is no longer required. The RPMs that were being pulled from EPEL are now included in the Moab RPM download. Previously, EPEL was automatically installed by the install-rpm-repos.sh script.

Turn Off Idle Node Purging on a Per Node Basis

For elastic computing, you can turn off the purging of *individual* dynamic nodes output using the varattr attribute in Torque or through the wiki interface.

Unset the MAXLOAD on a Specific Node with nodecfg

Added ability to set MAXLOAD to -1 on a node to unset the default.

9.0.1

Added UseRMCompletedJobStats SCHEDCFG Flag

Use RM reported CPUtime when calculating job efficiency with SCHEDCFG[] FLAGS-S=UseRMCompletedJobStats.

Enable NODEAVAILABILITYPOLICY on Reservations and NODECFG

Added ability to set NODEAVAILABILITYPOLICY on a per-NODECFG basis.

Limit Max Array Sub Job Size

Enable USERCFG[] MAXARRAYSUBJOBS to limit the size of job arrays on a per user basis.

Added mjobctl -m Option to Alter MAXMEM

Differences

Enabled "mjobctl -m MAXMEM=X" to modify a job's maximum memory limit.

Added Low-Latency v2 Commands

For low-latency you can now use the following commands: showq, showres, mdiag, checkjob, checknode, showstats, showbf, showstart, showconfig.

User Can Choose Where to Save File

Added ability to specify path of log level with "mschedctl -L".

9.0.0

AMCFG[] THREADPOOLSIZE Parameter Now Undocumented

The AMCFG[] THREADPOOLSIZE parameter has been removed from the documentation. Previously documented in Accounting, Charging, and Allocation Management in the *Moab Workload Manager Administrator Guide*.

Moab Always Attempt Charges

Moab will always attempt charges even if the accounting manager is marked as Down and will consistently update the accounting manager state after accounting operations.

Moab Installs init and profile Scripts by Default

It is no longer necessary to specify the --with-init and --with-profile configure options. These options are now enabled by default. You must use the --without-init and --without-profile configure options if you do not wish the init and profile scripts to be installed for your distribution.

Default Changed for USERPRIOWEIGHT

The default for the USERPRIOWEIGHT server parameter has changed from 0 to 1. This change for Moab to add a weight by default supports setting a user priority when creating a job through Viewpoint.

Additional Handling Option for Torque Condensed Queries

An additional RM configuration flag is added for handling Torque condensed queries. Use RMCFG[] FLAGS=EnableCondensedQuery to enable the queries. Whereas, you can use the existing RMCFG[] FLAGS=NoCondensedQuery to disable the queries.

i NoCondensedQuery is the default behavior for Moab 9.0 and later.

Reset UID/GID of Users with mcredctl -r uid

Enhanced mcredctl -r to reset the uid/gid of a given user. For example: 'mcredctl -r uid user:john' resets the uid/gid for the user named john.

FSSCALINGFACTOR Pre-Partition Setting

Enabled "PARCFG[] FSSCALINGFACTOR" for partition-specific fairshare usage scaling.

Job Submit Time Sent to Accounting Manager

Moab now sends the job submit time to Moab Accounting Manager; enabling it to be recorded with the usage record for the job.

Change to ALWAYS-EVALUATE-ALL-JOBS Configuration Parameter

The configuration parameter ALWAYS-EVALUATE-ALL-JOBS was changed from a boolean to an enumerated value. The possible values are ALWAYS (formerly TRUE), FIRSTRSV (formerly FALSE), and FULLRSV (an intermediate setting).

i No change is required when upgrading from earlier versions. The TRUE value will map to ALWAYS and the FALSE value will map to FIRSTRSV.

mdiag -j Now Displays the Node Count Instead of the Processor Count

Added DISPLAYFLAGS NODECENTRIC feature to the output of 'mdiag -j'.

RSVSEARCHALGO by Partition

Enabled "PARCFG[] FLAGS=WideRsvSearchAlgo" to allow for per-partition specific scheduling rules. See the RSVSEARCHALGO parameter in the *Moab Workload Manager Administrator Guide*.

MAXJOB and MAXRSVPERNODE Default Increase

The defaults for MAXJOB and MAXRSVPERNODE are increased to accommodate advancements in system performance.

- The default for MAXJOB has changed from 4096 to 51200.
- The default for MAXRSVPERNODE has changed from 48 to 64.

Enhancement to "make install"

Added default .moab.key file to "make install" with randomly generated key.

Moab updates MAM Usage Record with Latest Accounting Stage

Moab passes the Accounting Stage (e.g. Create, Start, Pause, Update, Continue, Resume, End) to Moab Accounting Manager to store with the Usage Record in order to provide better information about the latest registered accounting stage.

Moab Accounting Manager

9.0.3

No known differences.

9.0.2

Differences

No known differences.

9.0.1

Sample Allocation Summary Report Provided

A sample allocation summary report has been added to the contrib directory that summarizes the status of current and expired allocations. By default it will display fund constraints and allocation and usage details for the currently active allocations. If a time-frame is given, both active and expired allocations overlapping the specified time period will be displayed. Filters can be specified to narrow the set of allocations displayed. Specific fields can be specified for display. A totals line is displayed unless using the csv or raw format. Additional fields include the number of jobs and reservations charged, processor-seconds used, as well as percent of idle reservation cycles used.

9.0.0

References to Gold Are Replaced with References to MAM

Command names, configuration files and log files have been renamed to replace references of Gold with MAM.

i MAM can be configured via a configure option (`--with-legacy-links`) to create symbolic links to the old server and client command names. Also see the corresponding item in [New Features](#) for more information about the configuration option and the change in general.

The mam server command (`goldd`) was renamed to `mam-server`. Correspondingly, the server configuration file (`goldd.conf`) has been renamed to `mam-server.conf` and the server log files (`goldd.log*`) have been renamed to `mam-server.log*`. The client configuration file (`gold.conf`) has been renamed to `mam-client.conf` while the client log files (`gold.log*`) have been renamed to `mam-client.log*`. The gui configuration file (`goldg.conf`) has been renamed to `mam-gui.conf` while the gui log files (`goldg.log*`) have been renamed to `mam-gui.log*`. The site configuration file (`site.conf`) has been renamed to `mam-site.conf`. The client module `Gold.pm` has been renamed to `MAM.pm` and the modules now reside in the `lib/MAM` directory and are referred to with the `MAM::` package name. The `hpc.gold` bootstrap script has been renamed to `hpc.mam`. The `--with-gold-libs` configure parameter was renamed to `--with-mam-libs`. The gauth security promotion method has been renamed to `mamauth` while the gauth program has been renamed to `mam-auth`.

The client commands have also been renamed as follows:

- `goldsh` is now `mam-shell`
- `g{balance,refund,statement}` are now `mam-{balance,refund,statement}`
- `g{deposit,transfer,withdraw}` are now `mam-{deposit,transfer,withdraw}`
- `g{charge,reserve,quote}` are now `mam-{charge,reserve,quote}`
- `gmk{account,event,fund,lien,org,quote,rate,role,usage,user}` are now `mam-create-{account,event,fund,lien,organization,quote,chargerate,role,usagerecord,user}`

Differences

- `gch {account,alloc,event,fund,lien,org,quote,rate,role,usage,user}` are now `mam-modify- {account,allocation,event,fund,lien,organization,quote,chargerate,role,usagerecord,user}`
- `gchpasswd` is now `mam-set-password`
- `grm {account,alloc,event,fund,lien,not,org,quote,rate,role,usage,user}` are now `mam-delete- {account,allocation,event,fund,lien,notification,organization,quote,chargerate,roles,usagerecord,user}`
- `gls {account,alloc,event,fund,lien,org,not,quote,rate,role,trans,usage,user}` are now `mam-list- {account-s,al-locations,events,funds,liens,organizations,notifications,quotes,chargerates,roles,transactions,usagerecords,users}`
- `glsconfig` is now `mam-read-configuration`

Zero-Sized Charges Must be Attributed to a Valid Allocation

Zero-sized charges will result in a failure if no valid allocations can be found for the charge. Formerly, zero-sized charges were always permitted even if they could not be attributed to any valid allocation.

Usage Recorded for Failed Charges

Charge failures will now result in proper recording and clean up. When a charge fails, MAM will create a usage record to track the usage, clean up liens, and record the relevant transactions. The failure message will be recorded in the charge transaction's description. Formerly, if a charge failed, the entire transaction would be rolled back, leaving straggling liens and no trace of the job's completion in MAM.

UsageRecord Stage to Store Moab Accounting Stage

The function of the UsageRecord Stage property has been changed to store the latest Moab accounting job phase rather than the last MAM routine called. This enables better information about the current Moab job stage (as it relates to accounting), such as whether the job has started, paused, resumed, performed a periodic accounting update, or ended. The `mam-create-usagerecord`, `mam-modify-usagerecord`, `mam-charge`, `mam-reserve`, and `mam-quote` command-line clients now accept a new corresponding `--stage` option. The `mam-charge -x (State)` option has been removed.

setDatum Method Call is Renamed to addDatum

The `setDatum` method call in the Request and Response objects has been changed to be `addDatum` since the call is appending a new datum to the data list.

Submit Time is Now Tracked with the Usage Record

A `SubmitTime` attribute has been added to the UsageRecord object to enable Moab to pass submit time in with the accounting information. The `SubmitTime` attribute can be displayed in the `mam-list-usagerecords` command line client.

Changes to -s and -e Options for mam-list-transactions (formerly glstrans)

The `-s` option now displays transactions occurring ON OR after the specified start time. The `-e` option now displays transactions occurring strictly before the specified end time. These changes were made to

Differences

give you the transactions for an inclusive start time and exclusive end time to facilitate reporting for adjoining periods.

mam-list-transactions Can Show Usage Record Fields

The mam-list-transactions (formerly glstrans) --show option has been expanded to allow unambiguous usage record properties derived from the stored usage record id to be specified for display with the transaction query (e.g. Group, Organization, Class, QualityOfService, Nodes, Processors, Memory, Disk) as well as the derived fields: NodeHours, NodeSeconds, ProcHours, and ProcSeconds.

Changes to -s and -e Options for mam-list-usagerecords (formerly glsusage)

The -s option now displays usage records that ENDED on or after the specified start time rather than usage that STARTED on or after the specified time as it did previously. The -e option now displays usage records the ended STRICTLY after the specified end time rather than usage that ended ON OR after the specified end time. These changes allow you to more consistently draw a line on which period a specific usage record is reporting against so that reports from adjoining periods do not double count the same usage.

Additionally, mam-list-usagerecords has been expanded to allow the derived fields NodeHours, NodeSeconds, ProcHours, and ProcSeconds.

Support Added for systemd init Service

The configure/install process will now check to see if your system supports systemd and will automatically install the systemd init service script if supported by your operating system.

Change to Zero Default Deposit Amount Behavior

A zero default deposit amount now results in an allocation being created with a zero balance when the fund is reset. Formerly, a zero default deposit amount would result in the ending of a fund's active allocations when the fund was reset. Using a negative default deposit amount is now required to end a fund's active allocations when the fund is reset. This change was made to support sites who intentionally create zero-balance allocations (e.g. when using fallback). The migration script will update zero values for the fund deposit amount to be negative so as to maintain the originally intended behavior.

Optionless Argument Changes for Certain Commands

- mam-list-transaction – An optionless argument is taken as the transaction id.
- mam-{create,delete,list,modify}-chargerate(s) – An optionless argument is taken as the name. This changes the previous behavior of mam-create-chargerate where the optionless argument was taken as the amount.
- mam-create-usagerecord no longer takes instance as an optionless argument.

MAM GUI Only Provides the Viewpoint Skin

When installing the MAM GUI, the Viewpoint skin will always be used. The --with-gui configure option no longer supports a skin argument and the associated gui.style GUI configuration parameter has been removed.

Minor Perl API Method Changes

- A `MAM::Request->delOption` method has been added to delete an option from a request.
- A `MAM::Request->addOption` method has been added to add an option to a request.
- The `MAM::Request->setOption` method has been modified to replace existing options with the same name.
- The `MAM::Datum->delProperty` method has been modified to permit the property name as its only argument; no longer allows (name, value).
- The `MAM::Datum->addProperty` method has been removed since `MAM::Datum->setProperty` is sufficient for all current use cases.

Quotes, Liens, and Charges Are Applied Uniformly

Quotes, liens, and charges are now applied uniformly to the allocations that were active when the usage occurred, even if the allocations are currently expired. Formerly, a quote, lien, or charge would apply only to the currently active allocation, even if the quote, lien, or charge was for a job that completed in the past.

This behavior has changed such that a charge that is issued for usage during a past timeframe will now charge allocations that were active during the usage timeframe rather than the currently active allocation. Furthermore, charges for usage that spans allocation periods will be broken up into their respective allocations. Similarly, a lien or quote that is issued for usage occurring in a future timeframe will be broken up among current and future allocations if such allocations have been predefined. By default, the charge, lien, and charge routines will complain if there are gaps in the allocations covering the usage timeframe. However, by default, if a gap is encountered, it will be filled by the other existing allocations in the usage timeframe. It is important to note that when a charge is issued, in order to avoid double booking against different allocations, charges will first be made against the allocations for which liens exist, and any remainder will then be charged uniformly to the allocations that were active during the usage timeframe. This change coincides with the new ability in Moab to retry failed charges. This ability to charge expired allocations can also be useful for sites that wish to import past usage into MAM.

Refunds Applied to the Originally-Charged Allocations

Refunds are now applied to the allocations that were originally charged. Formerly, refunds would be applied only to currently active allocations. With this change, refunds are applied to the allocations that were originally charged, even if they are currently expired. Refunds can also accept a new allocation option which specifies the allocation to be credited if you wish to have greater control over the allocations to be refunded. In addition, the former fund and filter options have been removed.

Moab Web Services

9.0.3

No known differences.

Differences

9.0.2

No known differences.

9.0.1

New dedicateAllProcessors Field

The job requirements array has a new field: `dedicateAllProcessors`

Within a requirement, if `dedicateAllProcessors` is true, then all processors on the node where the job runs will be dedicated to the job.

Deprecated MWS RM Perl Scripts

The use of MWS as a native resource manager has been deprecated.

```
# Deprecated
RMCFG [mws] TYPE=NATIVE
RMCFG [mws] CLUSTERQUERYURL=exec://$TOOLSDIR/mws/cluster.query.mws.pl
RMCFG [mws] WORKLOADQUERYURL=exec://$TOOLSDIR/mws/workload.query.mws.pl
...
```

Therefore, these scripts have also been deprecated:

<code>cluster.query.mws.pl</code>	<code>image.update.mws.pl</code>	<code>job.resume.mws.pl</code>	<code>node.modify.mws.pl</code>
<code>image.create.mws.pl</code>	<code>job.cancel.mws.pl</code>	<code>job.start.mws.pl</code>	<code>node.power.mws.pl</code>
<code>image.delete.mws.pl</code>	<code>job.modify.mws.pl</code>	<code>job.submit.mws.pl</code>	<code>workload.query.mws.pl</code>
<code>image.list.mws.pl</code>	<code>job.requeue.mws.pl</code>	<code>job.suspend.mws.pl</code>	

If you still need these scripts, they are available in `$TOOLSDIR/tools/mws/deprecated`. They may be removed in a future release.

The new approach is to use the MWS resource manager type.

```
RMCFG [mws] TYPE=MWS
RMCFG [mws] FLAGS=UserSpaceIsSeparate
RMCFG [mws] BASEURL=http://localhost:8080/mws
```

See *Configuring Moab Workload Manager* in the *Moab Web Services Reference Guide* for more information.

9.0.0

No known differences.

Differences

9.0.3

No known differences.

9.0.2

No known differences.

9.0.1

Expose Queue Status to Job Details Page and the Workload Page

Insight now stores a job's queue status in both the relational and Mongo databases.

9.0.0

N/A

Moab Viewpoint

9.0.3

Additional Base Roles

Viewpoint now configures with six default (base) roles. The HPCAdmin and HPCUser are the same as they were in previous Viewpoint versions. The other four roles are:

- NitroAdmin – Administrative user, with permission to create Nitro application templates and manage other user's Nitro jobs.
- NitroUser – Basic user, with permission to create and manage their own Nitro jobs.
- RemoteVizAdmin – Administrative user, with permission to create remote visualization application templates and manage other user's remote visualization jobs.
- RemoteVizUser – Basic user, with permission to create and manage their own remote visualization jobs.

i All six default roles are present regardless of whether your configuration includes Nitro or Remote Visualization.

Two Messages No Longer Generated When QoS Changed

Previously, when changing a QoS for a job, Moab changed the system priority to 0 and Viewpoint countered by resubmitting the appropriate system priority; resulting in two messages stating that the system priority changed. A QoS change now generates only one message.

Differences

9.0.2

No known differences.

9.0.1

Viewpoint Uses Latest Version Django Framework

As part Viewpoint now uses the latest version of the Django framework. This change allowed Adaptive Computing to streamline and simplify some Viewpoint configuration and management steps. For example:

- `/opt/viewpoint/iris/settings.py` is replaced with `/opt/viewpoint/etc/viewpoint.cfg`. The `viewpoint.cfg` file contains the values for the Viewpoint administrator username and password, the Insight database environment variables and the Resource Job Timeline threshold settings.
- The `manage.py` script (`/opt/viewpoint/manage.py`) is not used. To initialize Viewpoint's PostgreSQL database use `/opt/viewpoint/bin/viewpoint migrate` (this selects the correct version of Python).
- WebDav client directory change. was: `/opt/viewpoint/webdav_client`, now: `/opt/viewpoint/lib/viewpoint/webdav_client`

9.0.0

N/A

Torque Resource Manager

6.0.3

`pbs_server` Does Not Shut Down When `trqauthd` is Stopped

The `trqauthd` service is no longer a requirement of `pbs_server`, so `pbs_server` no longer shuts down when `trqauthd` is stopped.

TorqueConfigure Script Does Not Enable the GUI Component

When the `devel-tk` and `devel-tcl` libraries have been installed, the Torque configuration script does not enable the GUI component. To build the GUI component, specify `--enable-gui` when building from source.

`$CUDA_VISIBLE_DEVICES` is not set by default

`$CUDA_VISIBLE_DEVICES` is not set by default if you're using `cgroups`. Also, this makes its contents compatible with varying CUDA versions.

Default Value for `keep_completed` Parameter Changed

The default value for the Torque `keep_completed` parameter has been changed to 300 seconds. The `keep_completed` parameter specifies the number of seconds jobs are kept in the job execution queue after they are completed. Keeping completed jobs in the queue enables Torque to report on the status of the jobs.

6.0.2

NUMA-Aware cgroup Creation by Per Task or Per Job

A new Torque server parameter "`cgroup_per_task`" is available to let you specify whether cgroups are created per task or per job. The default is `FALSE`, meaning jobs submitted with the `-L` syntax will have *one* cgroup created per host; this behavior is similar to the pre-6.0 `cpuset` implementation.

The `qsub/msub -L` syntax is also modified to let you specify whether the cgroup is per task or per job at the job submission time.

 Some MPI implementations are not compatible with using one cgroup per task.

`legacy_vmem` Server Parameter Affects Behavior of the `-l vmem` Option

`legacy_vmem` is a new server parameter that affects the behavior of the `-l vmem` option. When set to true, the `vmem` request will be the amount of memory requested for each node of the job. When it is unset or false, `vmem` will be the amount of memory for the entire job and will be divided accordingly.

Queue Support for Both `resources_default.*` and `req_information_default.*` Settings

When queues have both `resources_default.*` and `req_information_default.*` set then they are applied according to their resource request type. `resources_default.*` settings are applied to jobs that do not explicitly use the `-L` syntax, while `req_information_default.*` settings are applied only to jobs that explicitly use the `-L` resource syntax.

Prohibited Mode Not Allowed for User Jobs

Setting the compute mode of an NVIDIA GPU to prohibited makes it so the GPU cannot be used at all. In previous versions of Torque users were allowed to set a GPU to prohibited mode. But then it could not set the mode to anything else since the GPU was now prohibited. This change went into effect for version 5.1.3, 6.0.2, and later.

Support for Single Job Dependencies and Array Dependencies at the Same Time

Jobs can depend on single job dependencies and array dependencies at the same time.

Reduced the Number of Logging Statements

Differences

Reduced the number of logging statements when a node isn't up and therefore can't receive the mom hierarchy.

Added `tcp_incoming_timeout` Server Parameter

`tcp_incoming_timeout` specifies the number of seconds before incoming connections timeout. `tcp_timeout` now specifies the timeout for outgoing connections or connections initiated by `pbs_server`. `tcp_incoming_timeout` functions exactly the same as `tcp_timeout`, but governs incoming connections while `tcp_timeout` governs only outgoing connections (or connections initiated by `pbs_server`).

6.0.1

Revert `vmem` Calculation Changes

Added the ability to control whether or not `vmem` is seen as per job or per node when `cgroups` are enabled.

Submission Syntax Check Added to Prevent Mixing NCPUs and Nodes

`qsub` guarantees that `ncpus` and `nodes` cannot be mixed.

Added a Way for `allow_node_submit` Exceptions

Added a way to exclude compute nodes from `allow_node_submit`.

Added Capability to Pass Environment Variables to `pbsdsh`

Added capability to pass environment variables to tasks created using `pbsdsh`. Two new options have been added:

`-e list` - Lets user specify list of environment variables separated by commas. If only a variable name is listed or a variable name is given with no value (ex. `name=`), its value will be read from the `pbsdsh` environment if it exists, otherwise it will be empty. If a variable name with a value is specified (ex. `name=value`) then the specified value will be assigned to the variable name in the tasks' environment.

`-E` - Include all environment variables from the `pbsdsh` environment in the tasks' environment.

i When using `-e` and `-E` together, and when common variable names are read (or set in the case of `-e`), if `-e` is specified *first* then `-E` read values will prevail in the tasks' environment. Otherwise, `-e` specified values will prevail.

`qmgr` Support Added for "loglevel" Attribute

Allows for `qmgr` to recognize "loglevel" as an equivalent of "log_level". The user can now type in either as a valid attribute.

`pbs_server` Enhancement for Very Large Number of Jobs

`pbs_server` has been enhanced to better handle a very large number of jobs (several hundred thousand or more) by enabling an alternate way for it to store job-related files in the directories `$PBS_HOME/server_priv/jobs` and `$PBS_HOME/server_priv/arrays`.

A new boolean server attribute, `use_jobs_subdirs`, lets an administrator direct the way `pbs_server` will store its job-related files. When `use_jobs_subdirs` is unset (or set to false), job and job array files will be stored directly under `$PBS_HOME/server_priv/jobs` and `$PBS_HOME/server_priv/arrays`. This is the default behavior and the way the server has stored these files in the past. When `use_jobs_subdirs` is set to true, job and job array files will be distributed over 10 subdirectories under their respective parent directories. This method helps to keep a smaller number of files in a given directory.

If an administrator wishes to change the `use_jobs_subdirs` attribute from its previous value (or when setting it to true when it has not previously been set), it is highly recommended that Torque be drained of all jobs. Failing to take this action may result in the loss of existing jobs.

6.0.0

Sprologalarm is Always Honored

`$sprologalarm` was ignored on the prologue for a job. Also when the epilogue was run the `$sprologalarm` value was ignored if it was more than 300. Now the `$sprologalarm` value is always honored regardless of how large it is for both prologue and epilogue scripts. The default timeout is still 300 seconds.

pbs_mom Now Sets Environment Variable for NVIDIA GPUs

A new mom config parameter, `$cuda_visible_devices`, was added to specify whether `pbs_mom` sets the `CUDA_VISIBLE_DEVICES` environment variable when it starts a job. The default is TRUE.

down_on_error Server Parameter Now Defaults to TRUE

By default, nodes that report an error from their node health check to `pbs_server` will be marked down and unavailable to run jobs.

Default RPM Installation Path Is Changed

The Torque default path for an RPM installation has been changed to match the path used during a tarball (Manual) installation. The default path for both install methods is `/usr/local`.

Installation and Upgrade Information

This section identifies information useful when installing and upgrading.



When installing or upgrading, it is *strongly* recommended that administrators configure Moab with mauth authentication with a complex key value. See Mauth Authentication in the *Moab Workload Manager Administrator Guide* for more information.

In this section:

- [Compatibility Requirements](#)
- [Installing Moab HPC Suite 9.0.3](#)
- [Upgrading to Moab HPC Suite 9.0.3](#)

Compatibility Requirements

This section provides information on compatibility with specific OS distributions and between the different components of the suite.

Moab Workload Manager and Torque Resource Manager

Although the recommended configuration is Moab version 9.0 and Torque version 6.0, Moab version 9.0 also supports Torque version 5.0 and 5.1.

Torque 6.0 requires 8.0 or later; however, some Torque 6.0 functionality requires Moab 9.0.

Moab Web Services

Moab Web Services does not support SUSE 11-based systems.

Since SLES 12 has ceased to provide MongoDB 2.4 in its repositories, fresh 9.0 tarball installs are not supported for MWS on SLES 12. However, upgrades of existing 9.0 MWS tarball installs to 9.0.3 are supported on SLES 12. Customers desiring to install MWS on SLES 12 may install MWS 9.0.3 via RPM (which bundles the missing MongoDB RPMs) or may install MWS 9.1 via tarball or RPM (which uses a newer version of MongoDB).

If you are using Moab Web Services with your current Moab solution, Moab needs to be installed on a MWS-compatible OS.

Moab Accounting Manager

Moab Accounting Manager version 9.0 is compatible *only* with Moab Workload Manager version 9.0.

If you are using Moab Accounting Manager with your current Moab solution, you will need to upgrade to the new Moab Accounting Manager 9.0 at the same time that you upgrade to Moab Workload Manager 9.0.

Moab Viewpoint Version 9.0.2

Installation and Upgrade Information

Moab Viewpoint version 9.0.2 requires Moab Workload Manager and Moab Insight to also be version 9.0.2.

Installing Moab HPC Suite 9.0.3

Please see the *Moab HPC Suite Installation and Configuration Guide* for manual or RPM-based installation instructions.

Upgrading to Moab HPC Suite 9.0.3

Please see the *Moab HPC Suite Installation and Configuration Guide* for manual or RPM-based upgrade instructions.



The RLM v12.1 (build:2) release resolved memory leak and security issues. The RLM package available with Moab HPC Suite 9.0.2, contains the v12.1 (build:2) release. Adaptive Computing *strongly* recommends that your RLM Server is v12.1 (build:2).

Known Issues

This section lists known issues in Moab HPC Suite. Following each issue description is an associated issue number in parentheses. Known issues are aggregated and grouped by the release version for which they were first reported.

In this section:

- [Moab Workload Manager](#)
- [Moab Accounting Manager](#)
- [Moab Web Services](#)
- [Moab Insight](#)
- [Moab Viewpoint](#)
- [Torque Resource Manager](#)

Moab Workload Manager

9.0.3

No known issues.

9.0.2

Certain multi-node/multi-task jobs submitted using the new -L syntax will start correctly but on subsequent iterations the tasks per node will revert to 1. (MOAB-8718)

9.0.1

- Based on task placement and node hardware configuration, it is possible that Moab will schedule a job that has specified 'usecores' in its task definition to a node that is already full. This will cause the job to fail to start and Moab will continue trying to schedule the job to the same invalid node until the job is either canceled by a user or deferred by Moab. (MOAB-8447) *Resolved 9.0.2*

9.0.0

- Jobs submitted with invalid credentials are put in a held state, instead of rejected, until the administrator can respond. The checkjob command gives administrators further information regarding why the job is held. Blindly assuming that all held jobs should in fact be running RIGHT NOW is not only unsafe, but circumvents intentional Moab policies and workflow. An administrator should exercise care when resolving held jobs. (CVE-2014-5375, MOAB-7478, MOAB-7526)

Known Issues

- When installing or upgrading, it is *strongly* recommended that administrators configure Moab with mauth authentication with a complex key value. See Mauth Authentication in the *Moab Workload Manager Administrator Guide* for more information. (CVE-2014-5376, MOAB-7525, MOAB-7480)
- When altering a GRES with 'mjobctl -m' on a job submitted with "-l software=" (instead of with "-l gres="), the change incorrectly reverts after an iteration. As a workaround, use '-l gres=' instead of '-l software='. The 'software' syntax will be deprecated in favor of 'gres'. (MOAB-7631)
- Requesting multiple GRESEs with "-l software=" honors only the first license request. Use "-l gres=" instead. The 'software' syntax will be deprecated in favor of 'gres'. (MOAB-7630)

Moab Accounting Manager

9.0.3

No known issues.

9.0.2

No known issues.

9.0.1

No known issues.

9.0.0

No known issues.

Moab Web Services

9.0.3

No known issues.

9.0.2

No known issues.

9.0.1

No known issues.

Known Issues

9.0.0

No known issues.

Moab Insight

9.0.3

No known issues.

9.0.2

No known issues.

9.0.1

No known issues.

9.0.0

No known issues.

Moab Viewpoint

9.0.3

- If you try to view a Remote Visualization session via Viewpoint, the FastX server may crash. (DOC-3091)
- When migrating jobs to Torque from Viewpoint, Moab will translate the request into the equivalent `qsub` command with the proper `-l procs` syntax. In some situations, Torque's queues may have been configured with a `default_resources.nodes` setting that is incompatible with the job's `-l procs` request. In this situation, the `default_resources.nodes` setting should be removed from the queue or the job should be submitted to a queue that does not have a `default_resources.nodes` setting. (DOC-3031)

9.0.2

- Viewpoint shows a 500 internal error if the connection is lost to Viewpoint's internal database. This is due to the fact that Viewpoint information is stored in the database itself. (IRIS-2537)

9.0.1

Known Issues

No known issues.

9.0.0

- When changing a QoS for a job, Moab changes the system priority to 0 but Viewpoint counters it by resubmitting the appropriate system priority; resulting in two messages stating that the system priority changed. This is normal behavior because changing the QoS also changes the system priority. (IRIS-1641)

Torque Resource Manager

6.0.3

No known issues.

6.0.2

- Torque won't compile when the tk-devel and tcl-devel packages are installed on your host. (TRQ-3723)
As a work around, disable building of the gui component by using --disable-gui when executing configure.
- qsub -X may incorrectly look for xauth in /usr/X11R6/bin/ instead of /usr/bin/. (TRQ-3489)
As a workaround, you can set XAUTHPATH /usr/bin/xauth in TORQUE_HOME/torque.cfg on client machines, and \$xauthpath /usr/bin/xauth in TORQUE_HOME/mom_priv/config on the compute nodes. Alternatively, you may be able to work around the issue by simply creating a sym-link from /usr/X11R6/bin/xauth to /usr/bin/xauth on pbs_mom hosts.

6.0.1

- pbs_mom failed to add job tasks to the devices cgroup on sister nodes of a parallel job. The failure to add a job pid to the devices cgroup results in the job not having restrictions to GPU or MIC devices. All GPU and MIC devices are available to the job. (TRQ-3522) *Resolved 6.0.2*
- Devices subsystem is enabled for cgroups. However for RHEL 6-based systems, the devices subsystem is considered a "Technology Preview". We have tested the devices subsystem and we have it working in our tests. However, any problems with the devices subsystem and Torque may be caused by the early access to this feature.
- When using cgroups, cgroup directories may be left behind for some jobs. Once the jobs are completed, these cgroup directories can be removed using rmdir at the convenience of the sysadmin. *Resolved 6.0.2*

6.0.0

Known Issues

- Running multiple instances of pbsdsh concurrently within a single job is not supported. (TRQ-2851)
- pbsdsh will fail to return under certain conditions (not-passing high-stress tests). (TRQ-3308)
Resolved 6.0.0.1
- Kernel crashes may occur when using cgroups on CentOS or RHEL prior to 6.6. See https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/6.6_Technical_Notes/kernel.html; especially RHEL6.6 fix BZ#1204626. If cgroups are part of your configuration, Adaptive Computing recommends running a more recent version of CentOS or RHEL. (TRQ-2583)

Resolved Issues

Resolved issues are aggregated and grouped by the release version in which they were resolved. When applicable, each resolved issue has the associated issue number in parentheses.

In this section:

- [Moab Workload Manager](#)
- [Moab Accounting Manager](#)
- [Moab Web Services](#)
- [Moab Insight](#)
- [Moab Viewpoint](#)
- [Torque Resource Manager](#)

Moab Workload Manager

9.0.3

- Future (DEPTH>1) liens did not charge the correct allocation. (MAM-390)
- mam-charge-retry.pl script failed for array jobs. (MAM-388)
- NODESET definition in a class overrode reservation affinity. (MOAB-8903)
- Moab was not creating standing reservations on all partitions by default. (MOAB-8888)
- Jobs with an invalid minwclimit were causing Moab to crash. (MOAB-8863)
- Jobs with NODEACCESSPOLICY SINGLEUSER were being put on hold under certain conditions. (MOAB-8821)
- Fixed an issue with -n (node exclusive) being set no matter what was set with NODEACCESSPOLICY. (MOAB-8816)
- Fixed an issue where mstat_converter was not converting data for import into a SQLite database. (MOAB-8803)
- The systemd file generated with the --prefix option was not using the specified path. (MOAB-8727)
- mdiag -S --xml did not display 5-minute statistics, as it did in non-xml mode. (MOAB-6896)

9.0.2

- systemd was terminating the active Moab server instead of the local Moab server. (MOAB-8667)
- Scheduler Analysis Block Reasons section displayed duplicate messages. (MOAB-8657)
- Task count was incorrect when requesting generic resource scheduling (GRES) on the global node. (MOAB-8627)

Resolved Issues

- Moab block messages were not always informative. (MOAB-8612)
- If jobs with a custom name were canceled, subsequent jobs with a similar custom name failed. (MOAB-8661)
- Jobs were rejected when requesting NUMA resources and NODEACCESSPOLICY=Y=SINGLEJOB. (MOAB-8565)
- msub did not allow you to specify a working directory, as with qsub -w <path>. (MOAB-8551)
- The class of GPU jobs that request more than one task was not being mapped correctly. (MOAB-8538)
- DPROC was not being calculated correctly on GPU jobs. (MOAB-8536)
- mcredctl was not reporting variables in credentials. (MOAB-8510)
- Jobs place on hold or batchhold were not being correctly recognized as being on hold. (MOAB-8433)
- Requeuing a job that was restarting caused it to be canceled. (MOAB-8422)
- Jobs with the "AFTER" dependency were not auto-cancelling. (MOAB-8387)
- Job ID collisions were occurring in a grid environment. (MOAB-8385)

9.0.1

- Requesting a node's name as a feature would prevent other jobs from getting that node. Jobs that request node names as features are placed on hold. (MOAB-8420)
- Issues reported groups, users, and accounts with spaces in the name. Enabled SCHEDCFG[] FLAGS=AllowCredentialsWithSpaces to let Moab ignore POSIX standards and allow groups, users, and accounts with spaces in the name. (MOAB-8399)
- Enabled JOBMAXPREEMPTCOUNT to limit number of times a single job can be preempted. (MOAB-8381)
- Machine becomes unresponsive when running "make" instead of "make install". A tarball install traditionally requires three steps: configure, make, and make install. Adaptive's tarball install installs precompiled binaries, therefore, the make step is not required. Running the make target (not make install) causes the make script to get caught in an infinite loop and consumes resources to the point the machine becomes unresponsive. The Makefile now informs the user to run "make install" when make is called without a target. (MOAB-8285)
- Request features on class not working. Enable CLASSCFG[] IGNHOSTLIST=TRUE to ignore hostlist requirements on jobs. (MOAB-8225)
- Jobs were blocked for jobs exceeding walltime incorrectly. Fixed issue with multi-req jobs and "EnforceReservedNodes". (MOAB-8105)
- Moab was not collecting template statistics correctly. Enable "JOBCFG[X] JSTAT=X" for collecting template statistics. (MOAB-8068)

9.0.0.1

N/A

9.0.0

- Wrong queuestatus was being shown for a blocked job. (MOAB-8203)
- Could not submit a job from a directory that contains a space in the name. (MOAB-8056)
- Standing rsvs were selecting new nodes for the rsv if one node was in a "draining" state. (MOAB-7990)
- Moab was scheduling jobs before setting up the rsv event table. (MOAB-7953)
- Moab sent low precision processor consumption rate to MAM causing charging discrepancies when using alternative charge policies. (MOAB-7940)
- Reservations could charge for cycles already charged to a job when DEBIT*BLOCKED charge policy is specified. (MOAB-7840)
- Jobs not taking all procs when "flags=allprocs" is requested on the job and "set queue batch resources_default.ncpus = 1" is set in Torque. (MOAB-7748)
- Liens could be left around when a non-running job is removed if threaded accounting has been enabled for job starts. (MOAB-7746)
- Bug reported with the setting of tasks while modifying the hostlist using "mjobctl -m hostlist=". (MOAB-7681)
- OMAX* parameters were not recognized in the identity manager. (MOAB-7567)
- Moab failed to register GRES update via qalter. (MOAB-7559)
- Start and completion times could be lost for jobs that finished while Moab was down temporarily. This would result in this information being missing from the event files and in zero-sized charges and lingering liens in MAM. (MOAB-7389)
- Reservation end time is not adjusted if a reservation is created where the start time is earlier than the present time. (MOAB-6412)

Moab Accounting Manager

9.0.3

- Allow a query offset to be expressed independently of a query limit. (MAM-392)
- Fixed Red Hat 6 rpm upgrade to use /sbin/service when restarting server. (MAM-381)
- Fixed an issue where unicode characters used in a request could result in mismatched digests due to a change in behavior of the Digest::SHA module. (MAM-379)

9.0.2

- MAM GUI sometimes failed after an RPM upgrade due to changed ownership of the GUI log file. (MAM-373)

Resolved Issues

- Clients using the mamauth promotion method were intermittently returning an empty digest and signature causing the failure message "Incoming digest does not match calculated digest." (MAM-372)
- Multi-chunk queries sometimes caused server to time out and client to hang. (MAM-361)

9.0.1

- The systemd unit file has been corrected to use the configured sbin directory for ExecStart. (MAM-338)
- Event alarm scoping problem that resulted in client "connection reset by peer" failures an in rare occasions resulted in the death of the server. (MAM-340, MAM-322)

9.0.0

- Authentication failures reported when using unicode characters with mamauth. (MAM-315)
- mam-read-configuration (formerly glsconfig) was installed with incorrect permissions. (MAM-310)
- MySQL migration process would create currency values as floating point rather than double precision; resulting in a loss of precision for very large currency amounts. (MAM-304)
- MAM init script is not LSB compatible and caused issues with Moab high-availability functions. (MAM-297)
- If a charge failed, the entire transaction would be rolled back, potentially leaving stale liens and no trace of the job's completion in MAM. Charge failures will now result in proper recording and clean up. For example, if a charge failure prevents the debit from occurring against the proper allocation, liens will still be cleaned up, the usage record will be updated, and the charge attempt will be recorded in the transaction table. (MAM-273)

Moab Web Services

9.0.3

- MWS could not handle a large volume of messages from ZMQ. (WS-2554)

9.0.2

- The MWS health check page did not correctly indicate LDAP DN's errors. (WS-2442)
- Performance when loading large LDAP trees with many attributes needed improvement. (WS-2254)

9.0.1

Resolved Issues

- During a POST to /mws/rest/jobs, MWS now returns a 400 error if tasksPerNode field is defined but taskCount is not defined. (WS-2504)
- CPUtime in accounting usage record was being rendered as a date-time. (WS-2469)
- Added javadoc for job field "submitCommandFile". (WS-2454)
- Exception reported when database "moab_insight" does not exist. MWS is designed to connect to the Insight database. If you are not using Insight, MWS will still work, but it will log errors like this to /opt/mws/log/jdbc.log: 2015-10-28T14:01:58.027-0600 localhost-startStop-1 ERROR org.apache.tomcat.jdbc.pool.ConnectionPool 0 Unable to create initial connections of pool You can ignore these errors if you are not using Insight.

If you are upgrading from a previous version of MWS, you will need to configure log4j to log org.apache.tomcat.jdbc messages to /opt/mws/log/jdbc.log. See rollingFile name: 'jdbc' and warn additivity: false, jdbc: 'org.apache.tomcat.jdbc' in the sample mws-config.groovy file shown in Upgrading Moab Web Services in the *Moab HPC Suite Installation and Configuration Guide*. (WS-2424)

9.0.0.1

- The MWS plugin "ViewpointQueryHelper" consumes MongoDB threads and connections. This resulted in the eventual failure "java.lang.OutOfMemoryError: unable to create new native thread". (WS-2449)

9.0.0

- Problems reported with credential REST queries. Changed max_idle_jobs, max_jobs, max_processors, max_processor_seconds, and max_nodes from integer to string. (WS-2388)

Moab Insight

9.0.3

No resolved issues.

9.0.2

No resolved issues.

9.0.1

- Request to not store as many idle cpu stats. Insight no longer records job samples when the job is not running. (Avoids taking up database space with rows that will be all zeros) (IN-429)

Resolved Issues

- First row in job_samples table for each job always contains 0s. Inserted a "virtual" row in the job_samples table for the moment when the job starts so that the actual job_sample would not always contain zeros. (IN-419)
- Node's initial state was not being recorded. (IN-316)

9.0.0.1

- Creating a reservation overwrites the node state in the Insight database. (IN-446)

9.0.0

No resolved issues.

Moab Viewpoint

9.0.3

- File Manager now considers all groups to which a user belongs when making decisions on access rights. (Previously, it only considered the user's primary group.) (IRIS-3161)

9.0.2.1

- Download operation in Viewpoint's File Manager UI was not working. (IRIS-2655)

9.0.2

No resolved issues.

9.0.1

- File Manager displayed no feedback message after a copy/move operation. (IRIS-2593)
- When uploading files, File Manger overwrote files with the same name. (IRIS-2591)
- File Manager displayed an internal server error when a user without read permission attempted to read a file or its details. (IRIS-2542)
- File Manager did not handle Unicode file and directory names correctly. (IRIS-2511)

9.0.0

No resolved issues.

Torque Resource Manager

6.0.3

- GPU models k40s and k80s were not detected correctly when using GPUs and cgroups. (TRQ-3861)
- pbs_server segfaulted on systems with large number of open file descriptors. (TRQ-3847)
- pbs_server segfaulted due to a change in the pthread library. (TRQ-3841)
- qsub loaded erroneous jobs after a server shutdown, causing pbs_server to crash. (TRQ-3839)
- Deleting a job without permission caused the dependencies to be removed. (TRQ-3833)
- Torque sometimes calculated available disk space incorrectly. (TRQ-3832)
- Queues applied conflicting defaults, such as when a job requests nodes and the queue has a procs default value. (TRQ-3830)
- Torque crashed due to a job with failed dependencies. (TRQ-3810)
- pbs_server crashed when using the mail throttling feature. (TRQ-3801)
- Systemd unit files do not correctly start/stop Torque daemons. A new -F (don't fork) switch has been added to pbs_server, pbs_mom, and trqauthd. (TRQ-3795)
- stdout was not properly delivered on NFS file systems when Torque was configured with --disable-spool.(TRQ-3792)
- Torque was not allocating resources correctly for cgroup jobs. (TRQ-3790)
- pbs_server was not starting when an array dependency was cleared before pbs_server started. (TRQ-3774)
- Torque had communication issues when IPv6 was configured but not used. (TRQ-3773)
- Stdout/stderr files were not handled correctly when there were spaces in the path. (TRQ-3767)
- Torque was not waiting for a compute node to do a copy out process. (TRQ-3762)
- Fixed node locking issues that were causing pbs_server to abort when built with newer versions of the threading library. (TRQ-3755)
- Server build was failing with --disable-spool configure option. (TRQ-3751)
- Requesting nodes with both mem and vmem limits caused cgroup memory limits to be set to an incorrect value. (TRQ-3749)
- Forced requeue command (qrerun -f) was not obeying kill_delay settings. (TRQ-3737)
- Job dependencies were not released when job_suffix_alias was set to something other than the hostname. (TRQ-3735)
- When adding a dynamic node, the server was initially putting the node in free state, but not reporting this state to the server. (TRQ-3733)
- The memory cgroup was not being set correctly for jobs requesting pmem and procs. (TRQ-3728)
- Jobs were not being removed from node_usage files after ending. (TRQ-3727)
- pbs_server aborted under ghost array recovery when job array files were missing. (TRQ-3719)
- pbs_mom crashed when started on the reporter node without \$apbasil_protocol set in the config. (TRQ-3715)

Resolved Issues

- momctl was not displaying the \$varattr script or the correct time and date. (TRQ-3712) (TRQ-3617)
- Error messages logged when jobs failed to copy files were misleading. (TRQ-3711)
- Jobs' environments and stdout paths were invalid when submitted from a deleted working directory. (TRQ-3609)
- qsig was not working correctly when `display_job_server_suffix = false`. (TRQ-3102)
- Allocating GPUs did not correctly set shared mode when requested. (TRQ-3213)
- pbs_mom init.d script sometimes timed out during startup. (TRQ-2911)
- qsub -v was not processing environment variables with no value correctly. (TRQ-2699)
- qstat -f was not displaying `init_work_dir`. (TRQ-2459)
- pbs_demux had an unnecessary library dependency. (TRQ-1996)

6.0.2

- Torque was not allocating enough memory controllers to satisfy memory requests. (TRQ-3681)
- pbs_server was not being properly shut down when in HA mode. (TRQ-3670)
- pbs_server was not detecting and updating `total_threads` when a node's hyperthreading was enabled. (TRQ-3662)
- pbs_server was not properly restarted when running "service pbs_server restart" during installation. (TRQ-3657)
- Memory and swap limits were not set in cgroup. For information on how memory and swap options are used, see [1.1 -L NUMA Resource Request](#). (TRQ-3656)
- Jobs submitted with -l option with exclusive access to the node were not receiving all CPUs and memory controllers in the cgroup. (TRQ-3649)
- Corrected logging to only log that a signal is sent to a process when it is actually issued. (TRQ-3638)
- pmem was not getting set correctly. With cgroups enabled, pmem is the amount of resident memory allocated per process where a process is given by the value of ppn. For example: `qsub -l nodes=1:ppn=2,pmem=250mb` will allocate a total of 500 MB on the node where the job is run, 250 MB per ppn. (TRQ-3628)
- NUMA -L syntax defaulted to override user-specified parameters. (TRQ-3623)
- qstat -x returned nothing (instead of an empty XML document) when there are not jobs queued. (TRQ-3622)
- Jobs in which a task required more than one socket could not be started using NUMA -L syntax. (TRQ-3618)
- \$usecp parameter was ignored when specifying which directories should be staged. (TRQ-3613)
- Server deadlocked when `job_save()` failed. (TRQ-3605)
- Tasks' memory usage was sometimes not reported. (TRQ-3601)
- Crash/infinite loop when loading certain node usage files. (TRQ-3576)

Resolved Issues

- Interactive jobs skipped submit filter directives if the first line was not #PBS. (TRQ-3585)
- Issue reported with alps login nodes. Updated cpusets for alps login nodes so that all of the cpus are in the job's cpuset. (TRQ-3568)
- Torque crashed intermittently when using the -L syntax. (TRQ-3566)
- Torque returned non-specific network failure messages to Moab. (TRQ-3539)
- Completed jobs were still reported in pbsnodes. (TRQ-3525)
- A deadlock occurred when handling job dependencies. (TRQ-3519)
- cgroup directories were not removed when jobs were completed. (TRQ-3515)
- drmaa unable to link with Torque. (TRQ-3511)
- Epilogue not showing up in momctl -d3 output. (TRQ-3495)
- Job dependencies were not being cleared with High Availability server. (TRQ-3477)
- A shell escape in pbs_mom's config file when specifying GRES did not show up in pbsnodes or Moab. (TRQ-3393)
- libtorque.so was not being created. (TRQ-3374)
- qrls gave no response and logged no problem when a failure occurred due to a slot limit restriction. (TRQ-3328)
- Problems building RPMs on Red Hat 6/CentOS 6 systems. (TRQ-3283)
- Jobs started even if mother superior could not resolve the hostname for a sister node. (TRQ-3134)
- Several log messages were unclear. (TRQ-2860)
- Job holds were not updated when the slot limit was changed for a job array. (TRQ-2360)

6.0.1

- Array subjobs did not have a queued entry in the accounting log. (TRQ-3470)
- Segfault in create_alps_subnode with node_note populated. (TRQ-3445)
- Problems with clearing a node note. Removed length restriction on a node note. (TRQ-3439)
- Jobs that never ran were receiving end records. (TRQ-3432)
- Resources_used.walltime changed to seconds from HH:MM:SS in accounting logs. (TRQ-3385)
- pbs_server timed out connection to pbs_mom. Added load balancing to login nodes when they start to get busy. (TRQ-3367)
- pbs_mom would hang when sending status from a child. Added a timeout for node health check scripts so that they cannot make the mom daemon hang. (TRQ-3364)
- pbs_mom hangs on restart with init script. Ensured that necessary services have been brought up before starting the Torque daemons and that the Torque daemons are shutdown before their required services are shutdown. (TRQ-3345)
- Fixed a memory leak when jobs were being started asynchronously. (TRQ-3326)
- qsub -W stage-in was not working. Fixed failures where the group name showed up in the log as the problem but the user did not belong to the group name given in the error. (TRQ-3312)

Resolved Issues

- Multiple moms sent invalid `destroy_alps_reservation/req_delete_reservation`. Only allows one kill orphaned reservation request per reservation at one time. (TRQ-3299)
- Jobs with square brackets in the name were aborted on restart if they weren't array subjobs. An issue was fixed with jobs getting aborted if they are named with "[]" in the name but aren't Torque array jobs. (TRQ-3214)
- Down/offline nodes caused TORQUE to not online elastic nodes. `pbs_server` is now able to bring up new nodes even when there are nodes in the system that are down or offline. (TRQ-3066)
- Array templates were being reported as jobs. (TRQ-3405)
- Memory calculation issues reported when cgroups enabled and `-l vmem|pmem|mem` are used. (TRQ-3499)
- Logs filled with messages about not sending hierarchy to mom. Failures are only logged the first time it can't send the hierarchy to a mom. (TRQ-3156)
- Error condition where the mom's port would be inserted into the `.JB` file name. (TRQ-3090)
- Torque was not able to release holds on job arrays. Running `qrls` on an array subjob allows `pbs_server` to correct slot limit holds for the array it which it belongs. (TRQ-3088)
- Completed jobs were not getting cleaned up. Fixed various issues when restarting dependency jobs, including them not getting removed even after completion. (TRQ-3175)
- Node recovers when behind processing requests. `pbs_server` now detects when a node is failing too frequently and mark it down temporarily if this happens. Once a node is marked down, it will be marked up again if either two consecutive communications from `pbs_server` to the node receive successful replies, or after five minutes of staying offline (whichever comes first). A node is considered to be failing too frequently if it has three failures to reply to a server request without have two consecutive successes in between. (TRQ-2517)

6.0.0.1

- A hang in `pbsdsh` occurred if the `pbs_mom` daemon was started with a `-q` or `-r` option. (TRQ-3308)
- Typo found in the error message reported when the swap memory limit could not be set.

6.0.0

- Threadpool in `pbs_mom` was not being started. When `$thread_unlink_calls` is set to true in `/var/spool/torque/mom_priv/config`, job files were not being deleted at job end in the mom. (TRQ-3232)
- Read timeouts were being retried indefinitely by `pbs_server`. (TRQ-3306)
- Reporter mom did not correctly handle UNKNOWN role. (TRQ-3245)
- Occasionally a random group name would show up for a user who did not belong in the group. A race condition was fixed by changing to thread safe calls to get group and user ids. (TRQ-3190)
- Interactive jobs not staying on the node from which they were submitted. (TRQ-3122)
- Jobs were getting stuck in a running state when an asynchronous run failed. (TRQ-3114)

Resolved Issues

- Array slot limits were not getting decremented when a job is preempted or rerun. (TRQ-3110)
- With `kill_delay` and `$exec_with_exec` set, a job would be set to a completed state after running `qre-run` instead of getting set back to queued. (TRQ-2993)