Moab HPC Suite – Enterprise Edition 9.0 Release Notes

Revised December 2015

The release notes file contains the following sections:

New Features

This section contains a summary of key new features.

In this section:

General Suite

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

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

9.0.0

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 as 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.

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.

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>]].

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*.

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.

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.

Additional CLASSCFG [] Attributes for Class Remapping

CLASSCFG now includes a MAX.TPN attribute. In addition, both MAX.TPN and MIN.TPN are enabled for class remapping.

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.

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.

Manually Write Out the Checkpoint File

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

New PREEMPTIONALGORITHM Parameter

The PREEMPTIONALGORITHM is added to designate how Moab handles preemption scheduling policies. Valid values are PREEMPTORCENTRIC or PREEMPTEECENTRIC. PREEMPTEECENTRIC 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.
- PREEMPTEECENTRIC – 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.

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>.

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.

> 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.

Diagnose License Information

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

Ignore Hostlist Requirements on Jobs

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

Query Details for Jobs that Have Already Terminated

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

Rebuild Standing Reservations While Moab is Running

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

High-Availabity Trigger

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

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.

Change the Requested Tasks Per Node for a Job

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

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.

Add Environment Variables to Jobs

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

Node Features Can Be Shared by the Same Class.

NODEACCESSPOLICY now supports the SINGLECLASS attribute.

Moab Accounting Manager

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 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

9.0.0

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

9.0.0

Moab Insight is the centralized, relational database available with Moab HPC Suite – Enterprise Edition. 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

9.0.0

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 to 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 be see that application template used to submit the job, but not edit the application template.

Torque Resource Manager

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

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

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

9.0.0

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.

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*.

Enhancement to "make install"

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

Ability to Specify Where to Save the Log File

The mschedctl -L command now includes a log file variable. This enables you to specify the save location for the log file. If no log file is given, Moab continues logging to Moab's default log file.

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.

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*.

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

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

Change to ALWAYSEVALUATEALLJOBS Configuration Parameter

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

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.

FSSCALINGFACTOR Pre-Partition Setting

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

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.

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.

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

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.

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.

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

9.0.0

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.

> 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}
- 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-{accounts,allocations,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 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

9.0.0

No known differences.

Moab Insight

9.0.0

N/A

Moab Viewpoint

9.0.0

N/A

Torque Resource Manager

6.0.0

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`.

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.

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.

$prologalarm is Always Honored

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

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

This section provides information on compatibility 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 4.29, 5.0.x and 5.1.x.

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

Moab Web Services and Moab Workload Manager

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

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.

Installing Moab HPC Suite 9.0

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

Upgrading to Moab HPC Suite 9.0

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

Known Issues

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

In this section:

Moab Workload Manager

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)
- 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

9.0.0

No known issues.

Moab Web Services

9.0.0

No known issues.

Moab Insight

9.0.0

No known issues.

Moab Viewpoint

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.

Torque Resource Manager

6.0.0

Running multiple instances of pbsdsh concurrently within a single job is not supported.

pbsdsh will fail to return under certain conditions (not-passing high-stress tests). *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.

Resolved Issues

This section lists 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

9.0.0.1

- 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. 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. The Makefile now informs the user to run "make install" when make is called without a target. (MOAB-8285)

9.0.0

- Reservation end time is not adjusted if a reservation is created where the start time is earlier than the present time. (MOAB-6412)
- 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)
- Moab failed to register GRES update via qalter. (MOAB-7559)
- OMAX* parameters were not recognized in the identity manager. (MOAB-7567)
- Bug reported with the setting of tasks while modifying the hostlist using "mjobctl -m hostlist=". (MOAB-7681)
- Liens could be left around when a non-running job is removed if threaded accounting has been enabled for job starts. (MOAB-7746)

- 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)
- Reservations could charge for cycles already charged to a job when DEBIT*BLOCKED charge policy is specified. (MOAB-7840)
- mnodectl -m features with regex only updates one node. Enabled mnodectl -m <*features*> x:<*node_regex*> for node features. (MOAB-7843)
- Moab sent low precision processor consumption rate to MAM causing charging discrepancies when using alternative charge policies. (MOAB-7940)
- Moab was scheduling jobs before setting up the rsv event table. (MOAB-7953)
- Standing rsvs were selecting new nodes for the rsv if one node was in a "draining" state. (MOAB-7990)
- Could not submit a job from a directory that contains a space in the name. (MOAB-8056)
- Wrong queuestatus was being shown for a blocked job. (MOAB-8203)

Moab Accounting Manager

9.0.0.1

None.

9.0.0

- 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)
- MAM init script is not LSB compatible and caused issues with Moab high-availablity functions. (MAM-297)
- 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-read-configuration (formerly glsconfig) was installed with incorrect permissions. (MAM-310)
- Authentication failures reported when using unicode characters with mamauth. (MAM-315)

Moab Web Services

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

9.0.0.1

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

9.0.0

N/A

Moab Viewpoint

9.0.0.1

None.

9.0.0

N/A

Torque Resource Manager

6.0.0.1

- A hang in pbsdsh occurred if the pbs_mom daemon was started with a -q or -r option. (TRQ-3308)
- Array templates were being reported as jobs. (TRQ-3405)
- Typo found in the error message reported when the swap memory limit could not be set.

6.0.0

- With kill_delay and $exec_with_exec set, a job would be set to a completed state after running qrerun instead of getting set back to queued. (TRQ-2993)
- Array slot limits were not getting decremented when a job is preempted or rerun. (TRQ-3110)
- Jobs were getting stuck in a running state when an asynchronous run failed. (TRQ-3114)
- Interactive jobs not staying on the node from which they were submitted. (TRQ-3122)
- 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)
- 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; threadpool in pbs_mom was not being started. (TRQ-3232)
- Reporter mom did not correctly handle UNKNOWN role. (TRQ-3245)
- Read timeouts were being retried indefinitely by pbs_server. (TRQ-3306)