

Moab HPC Suite – Enterprise Edition 8.1.2 Release Notes

The release notes file contains the following sections:

- [New Features on page 2](#)
- [Differences on page 7](#)
- [Installation and Upgrade Information on page 19](#)
- [Known Issues on page 21](#)
- [Resolved Issues on page 23](#)

New Features

The following is a summary of key new features in Moab HPC Suite – Enterprise Edition.

- [General Suite on page 2](#)
- [Moab Workload Manager on page 2](#)
- [Moab Accounting Manager on page 5](#)
- [Moab Web Services on page 5](#)
- [TORQUE Resource Manager on page 6](#)

General Suite

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

[8.1.1/5.1.1](#)

None.

[8.1.1/5.1.1](#)

Support for more OSs

These additional OSs are now supported:

- CentOS 7.x
- RHEL 7.x
- Scientific Linux 7.x
- SUSE Linux Enterprise Server 12

i Support for Red Hat 7-based and SUSE 12-based systems requires the 8.1.1.2/5.1.1.2 maintenance release and after.

[8.0.0/5.0.0](#)

None.

Moab Workload Manager

[8.1.2](#)

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.

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.

Node Features Can Be Shared by the Same Class

NODEACCESSPOLICY now supports the SINGLECLASS attribute.

8.1.1

No new features.

8.1.0

Elastic Computing Feature - Ability to Request Dynamic Nodes From an External Service

A new Elastic Computing feature is available to allow the Moab scheduler to utilize systems that can temporarily provide additional nodes (for example, to create new virtual machines or borrow physical nodes from another system) to fulfill increased workload demand so that job backlog is completed in a reasonable time frame. When this feature is enabled and configured, Moab accesses the dynamic nodes, also known as bursting, to handle the increased workload. Accessed nodes are then released once the demand is filled.

i Elastic Computing is only available with a Moab HPC Suite - Enterprise Edition license that has Elastic Computing enabled. Please contact your account manager at Adaptive Computing for further details and requirements for this feature.

Credential Synchronization between MAM and Moab Available for Native AM Interface

The AMCFG CREATECRED parameter can now be used with the native accounting manager interface (whereas, it was formerly restricted to use with the MAM accounting manager interface).

Accounting Manager and Identity Manager Permit a Finer Granularity Refresh Period

See the corresponding topic in [Differences on page 7](#) for more information.

Accounting Mode

The accounting mode (specified via the AMCFG[] MODE parameter) modifies the way in which accounting-relevant job and reservation stages (e.g. create, start, end, etc.) are processed. The accounting mode can be one of usage-tracking, notional-charging, fast-allocation, or strict-allocation.

- If usage-tracking is specified, charges will simply result in the creation of usage records with no charge value. No charge will be calculated and allocations will not be debited.
- If notional-charging is specified, a charge will be calculated and recorded with the usage record, but allocations are not debited.
- If fast-allocation is specified, usage records will be updated with charge amounts and allocations will be debited, but liens will not be used to protect the allocation from simultaneous use. The elimination of liens and quotes makes this a higher throughput option than strict-allocation.
- If strict-allocation is specified, usage records will be updated with charge amounts and allocations will be debited, and liens will be used to protect the allocation from simultaneous use.

Ability to Differentiate between Failure Action for Periodic Charging and after Resuming a Suspended Job

A new AMCFG[] CONTINUEFAILUREACTION parameter has been introduced to specify the action taken when allocated funds are insufficient for a job to continue when periodic charging is enabled (via AMCFG[] FLUSHINTERVAL). The failure action will, in general, be different from the AMCFG[] UPDATEFAILUREACTION which is used to specify the action taken when a suspended job is resumed with insufficient funds.

Moab's Scheduling Cycle No Longer Needs to Wait on the Cluster Query

A new "threadedqueries" resource manager flag is available. When this flag is set for an individual RM (for example, "RMCFG[torque] TYPE=PBS FLAGS=threadedqueries"), the queries that Moab performs to get information from the RM are done in a separate thread from the main Moab process. This allows Moab to remain responsive during the query, and ultimately reduces the time spent in a scheduling cycle. If multiple RMs are being used, the effect can be more significant because all RMs will be queried in parallel.

Ability to Specify Whether Periodic Liens are Combined or Partial

When periodic charging is enabled via `AMCFG[] FLUSHINTERVAL`, a new `AMCFG[] LIENGRANULARITY` parameter controls whether a lien is sought up front for the entire duration of the job or reservation (Combined) or whether partial incremental liens are obtained for each periodic charge interval (Partial).

Moab Accounting Manager

8.1.2

No new features.

8.1.1

No new features.

8.1.0

New Command to Query Configuration

A new `glsconfig` command was added to display enabled configuration parameter values.

Incremental Balance Now Tracked in Transaction Table

Each transaction that affects the allocation balance (e.g. a charge, deposit, refund, transfer, etc.) now records the resulting allocation balance in the transaction table. This provides a ledger of intermediate balances that can be displayed from within commands such as `glstrans` and `gstatement`.

Moab Web Services

8.1.2

No new features.

8.1.1

No new features.

8.1.0

Support for Multi-Line (textarea) Configuration Parameters in Plugins

MWS now supports multi-line (textarea) configuration parameters in plugins. See **Configuration Constraints** in the *Moab Web Services Reference Guide* for more information.

Trigger Object includes New Type Field

The Trigger object in MWS has a new field called type. The type of a trigger can be either generic or elastic.

TORQUE Resource Manager

5.1.2

Added Feature to Disable Editing pbsnodes File

The qmgr command "dont_write_nodes_file=true" disables all commands which edit nodes. If set to true, this setting locks all access to the nodes file, thereby ensuring the nodes file is never overwritten unintentionally.

Able to Exclude Compute Nodes from allow_node_submit

A new TORQUE server parameter "node_submit_exceptions" is available to specify nodes that are not allowed to submit jobs. This parameter is set in conjunction with "allow_node_submit".

Ability to Trust Certain Users or Groups from Hosts without Allowing All Users from those Hosts to Submit Jobs

Two new TORQUE server parameters "acl_user_hosts" and "acl_group_hosts" are available to support this new validation.

5.1.1

See [General Suite on page 2](#).

5.1.0

Ability to Provide Condensed qstat Output

A 'qstat -C' option, which specifies that TORQUE will provide only a condensed output (job name, resources used, queue, state, and job owner) for jobs that have not changed recently (as per the job_full_report_time parameter), has been added. Jobs that have recently changed will continue to send a full output.

Performance Enhancements to MOM Clean-up Time

Some minor performance enhancements were made to improve MOM clean-up time.

Differences

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

- [General Suite on page 7](#)
- [Moab Workload Manager on page 7](#)
- [Moab Accounting Manager on page 13](#)
- [Moab Web Services on page 15](#)
- [TORQUE Resource Manager on page 16](#)

General Suite

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

[8.1.2/5.1.2](#)

None.

[8.1.1/5.1.1](#)

Daemon Restart

If your configuration uses systemd to start or stop daemons, also use systemd to restart daemons instead of using the direct restart options.

For example, if you use `systemctl start moab.service`, use `systemctl restart moab.service` instead of the `mschedctl -R` option.

For example, if you use

- `systemctl start moab.service`, use `systemctl restart moab.service` instead of the `mschedctl -R` option.
- `systemctl start gold.service`, use `systemctl restart gold.service` instead of the `gold -r` option.



Support for systemd (Red Hat 7-based and SUSE 12-based systems) requires the 8.1.1.2/5.1.1.2 maintenance release and after.

[8.1.0/5.1.0](#)

None.

Moab Workload Manager

8.1.2

Enhanced showq Sorting Priority for Blocked Jobs

The showq command now sorts blocked jobs in descending priority order.

Condensed Queries Between Moab and Torque Disabled

"RMCFG[] FLAGS=NoCondensedQuery" is now the default setting. Previously, faulty job reservations were applied on nodes not allocated for specific job. These false reservations would occasionally corrupt walltimes and kill jobs prematurely. This change resolves those concerns.

Added 'JSTAT' to 'JOBCFG' Object

To apply this change, enable "JOBCFG[X] JSTAT=X" for collecting template statistics. This allows for ease-of-use in reporting and viewing statistics for the different application types. Statistics from different templates can be shown on the same JSTAT which can be seen by using the command "showstats -j".

Enabled "failure" Scheduler Trigger for Recovering in HA Scenarios

Added an HA trigger mechanism used for handling Moab failures. When Moab fails over, a system trigger is fired on the new active server allowing the user to run scripts and perform services such as changing floating IP addresses, stopping and starting MWS, postgres, and other services.

Added Configuration Option to Attach Environment Variables to All Jobs

To enable Moab job environment variables on all jobs, use the "SCHEDCFG[] FLAGS=EnableMoabJobEnv" setting.

Backfill Disabled Within Reservation When Owner Job is Present

Disable Moab backfill with the "SRCFG[] FLAGS=OwnerExclusiveBF" setting. This setting allows for full utilization of clusters by avoiding backfill-induced starvation. While this setting is not advised for all scenarios (backfill is desired in certain cases), this feature provides more options for users wishing to disable backfill.

Enhanced Command Support With UIMANAGEMENTPOLICY FORK

User-initialized UIMANAGEMENTPOLICY FORK now support the "mdiag -a|-c|-g|-q|-r|-S|-s|-t|-u", "showstate", and "showstats" commands.

Init.d Script Updated for Enhanced Start Up Reliability

"OS/EL/etc/init.d/moab stop()" function modified to use killproc -t60.

Set NODEAVAILABILITYPOLICY on a Per-NODECFG Basis

You can now use NODEAVAILABILITYPOLICY with NODECFG at the local level.

i The syntax has a different format when using it locally (with NODECFG) instead of setting it globally (using the NODEAVAILABILITYPOLICY server parameter).

Modify a Job's Maximum Memory Limit

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

8.1.1

Set Job Flags in Identity Manager

Enabled the JOBFLAGS parameter on accounts through moab.cfg and identity managers.

New PREEMPTIONALGORITHM Parameter

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

- PREEMPTORCENTRIC specifies 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 specified 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.

i Preemption now works with JOBMATCHPOLICY EXACTNODE.

Change to ALWAYSSEVALUATEALLJOBS Configuration Parameter

The configuration parameter ALWAYSSEVALUATEALLJOBS 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.

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*.

FSSCALINGFACTOR Pre-Partition Setting

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

msub MOAB_SUBMITDIR Environment Variable

MOAB_SUBMITDIR is populated with the submission directory for jobs submitted with "msub -E".

msub -P

Added '-P' option to msub command to match functionality with 'qsub -P'. This option can only be used by users in the ADMINCFG[1] security level.

Clarify Jobs Submitted with -n

Added log entry when job submitted with '-n' (node_exclusive). 'n' sets the node's access policy as SINGLEJOB.

New Argument for checknode -h

"ALL" is now a valid argument in 'checknode -h'.

Mandatory Queuetimes on Reservation Owners

Added new parameter "SRCFG[] OWNERPREEMPTQT=XX:XX" to configure mandatory queuetimes on reservation owners.

checkjob Change for Node Availability Information on Large Clusters

checkjob will no longer report node availability information on large (>1000 node) clusters unless a '-v' flag is included.

mrsvctl -m Disallows the Modification of a Standing Reservation

mrsvctl -m now disallows the modification of a standing reservation and returns an error to the user if this is attempted.

8.1.0

NAMI Scripts and Interface Changes

A new AMCFG QUERYURL parameter has been introduced to facilitate queries with the accounting manager. This is used in conjunction with the AMCFG CREATECRED parameter to synchronize MAM account and user information in Moab.

Default AMCFG URLs Do Not Need to be Specified When Using the Native Accounting Manager Type

When using the Native accounting manager type, Moab now defaults to using a set of stock scripts that no longer need to be explicitly configured in the server configuration file. If you are using Moab Accounting Manager with the native

interface (TYPE=Native), remove all entries in moab.cfg with the form (AMCFG[*] *URL=exec://*) except for those that you have customized.

Hybrid Accounting Manager Endpoints Permitted

When using the MAM accounting manager type, by default Moab will communicate directly with Moab Accounting Manager via the SSS wire protocol. However, it is possible to enable a hybrid model and override individual accounting actions by specifying the exec protocol and the path of a custom script to the appropriate AMCFG[*] *URL parameters.

Separation of Resume Accounting Stage Into Resume and Continue Stages

Formerly, the accounting "Resume" stage fulfilled the dual purpose of checking whether a job should be continued after being suspended, or to see whether it should continue running after a periodic "charge". However, types of failure actions needed for these two different stages are not, in general, compatible. A new "Continue" stage was introduced to check for authorization whether a job should continue after a periodic accounting update and is associated with the new AMCFG[*] parameters CONTINUEURL, CONTINUEFAILUREACTION, and CONTINUEISBLOCKING. The former "Resume" stage is now reduced to checking whether a job has authorization (e.g. funds) to resume after having been suspended (associated with the AMCFG[*] parameters RESUMEURL, RESUMEFAILUREACTION, and RESUMEISBLOCKING).

AMCFG DISABLEDACTIONS Parameter Is Deprecated

The AMCFG[*] DISABLEDACTIONS parameter is deprecated. It may be removed in a future release. Specify an empty value or a protocol of 'null:' for the corresponding AMCFG[*] CONTINUEURL, CREATEURL, DELETEURL, ENDURL, PAUSEURL, RESUMEURL, STARTURL, and UPDATEURL parameters instead.

AMCFG BLOCKINGACTIONS Parameter Is Deprecated

The AMCFG[*] BLOCKINGACTIONS parameter is deprecated. It may be removed in a future release. Specify the corresponding new AMCFG[*] parameters CONTINUEISBLOCKING, CREATEISBLOCKING, DELETEISBLOCKING, ENDISBLOCKING, PAUSEISBLOCKING, RESUMEISBLOCKING, and STARTISBLOCKING parameters instead.

AMCFG JOBFAILUREACTION Parameter Has Been Removed

The previously deprecated AMCFG[*] JOBFAILUREACTION parameter has been removed. Use AMCFG[*] STARTFAILUREACTION instead.

Accounting Job Charges Are Now Non-Blocking By Default

AMCFG[*] THREADPOOLSIZE now defaults to 2 and AMCFG[*] ENDISBLOCKING defaults to FALSE. Thus if accounting is turned on, job charges will be non-blocking by default.

Prevent Creation of Multiple Accounting Usage Records For a Job

Under certain conditions, multiple accounting usage records could be created for a single job. A new mechanism of storing the usage record id in the job's variable space helps to ensure that only one usage record is created during the lifetime of a job.

Accounting Manager and Identity Manager Permit a Finer Granularity Refresh Period

The AMCFG[] and IDCFG[] REFRESHPERIOD parameters have changed to accept a period of the form [[DD:]HH:]MM:]SS (or INFINITY). The calendar period form of MINUTE, HOUR, DAY, WEEK, or MONTH is now deprecated. The accounting manager or identity manager is now refreshed on the specified period relative to the scheduler start time rather than relative to the beginning of the month.

Periodic Accounting Updates Now Happen Relative to the Start of the Job or Reservation

The AMCFG[] FLUSHINTERVAL parameter has changed to accept a period of the form [[DD:]HH:]MM:]SS (or INFINITY). The calendar period form of MINUTE, HOUR, DAY, WEEK, or MONTH is now deprecated. Moab will update the accounting manager (e.g. make an incremental charge) on the specified period relative to the start of the individual job or reservation rather than being relative to the beginning of the month.

Accounting Manager ChargePolicy Defaults to DebitAllWC

The AMCFG[] CHARGEPOLICY now defaults to DEBITALLWC rather than DEBITSUCCESSFULWC. This change accommodates the majority of sites who want accounting for all jobs (and permits periodic charging to work).

Reservation Consumption Rate Passed Via Attribute

The reservation consumption rate (i.e. the ratio of idle processor seconds to total processors seconds) is now passed via a consumptionRate attribute on the Processors property instead of being sent as a standalone property called ConsumptionRate. This is now done in a similar fashion to jobs, since when periodic charging is enabled, the consumption rate will vary on a per-charge basis. It is no longer necessary to define a ConsumptionRate usage record property and a ConsumptionRate charge rate in Moab Accounting Manager, since the consumption rate will now be factored into the charge amount automatically.

Set Default Accounts on a Per Partition Basis

Added the ability to define default accounts per partition. Also available in fairshare trees.

Node Collection in the Moab Database in MongoDB

The node collection in the Moab database in MongoDB has an index on the attributes field. This field can grow too large to index.

- For existing installations, the following commands on the MongoDB server will fix the problem:

```
$ mongo moab -u moab user -p secret2
> db.node.dropIndex({"attributes":1})
```

The username and password for your database are most likely different from the above example. Check with your database administrator.

- For new installations using this and future releases, the index is no longer created and does not need to be dropped.

IDCFG[] Defaults to TRUE

The default value for IDCFG[] CREATECRED has been changed to TRUE. Moab will now create all credentials that it finds in the identity manager.

Energy-Consumption-by-Job Accounting

The Moab HPC Suite - Enterprise Edition has the ability to report, record, and charge a cost for the electrical energy consumed by a job. Cray-oriented power management reference scripts are enabled to handle a Moab/TORQUE/ALPS architecture where Moab and the TORQUE pbs_server are running on an x86 server inside (internal) or outside (external) the Cray network. Currently, Moab supports this capability only for Cray XC systems running CLE 5.2 or later. See **Cray-Specific Power Management and Energy-Consumption-by-Job Accounting** in the *Moab Workload Manager Administrator Guide* for more information on configuring power management and tracking energy consumption.

Moab Accounting Manager

8.1.2

No known differences.

8.1.1

No known differences.

8.1.0

New Command to Query Configuration

A new glsconfig command has been added to display enabled configuration parameter values.

Incremental Balance Now Tracked in Transaction Table

Each transaction that affects the allocation balance (e.g. a charge, deposit, refund, transfer, etc.) now records the resulting allocation balance in the transaction table. This provides a ledger of intermediate balances that can be displayed from within commands such as `glstrans` and `gstatement`.

Accounting Mode Can Be Demoted By Moab

Moab can override the `accounting-mode` setting to use a less strict value for individual charge, lien, and quote requests based on a new `AMCFG[mam] MODE` parameter in Moab by passing in a new `AccountingMode` request option.

New Transaction Balance and Remaining Attributes

The Transaction table has added two new attributes (`Balance` and `Remaining`) to record the resulting active balance and remaining allocation amount after every action that modifies the amount or activeness of the allocation.

- `Balance` records the effective active balance of the allocation (that takes into account whether the allocation is active or not).
- `Remaining` records the actual allocation amount (whether expired or active).

glstrans Can Display Balance and Remaining

`glstrans` was modified to be able to display the incremental available balance and remaining allocation amount (via the new `Transaction Balance` and `Remaining` attributes).

gstatement Now Displays the Incremental Balance For Itemized Reports

`gstatement` has been modified to display the incremental available balance in the debit and credit detail sections when an itemized report is being generated.

glsalloc, gbalance, and glsfund Have Changed the Names of the Balance and Amount Fields

For consistency with the meaning of the new `Transaction Balance` attribute, `glsalloc`, `gbalance`, and `glsfund` have changed the meaning of some of their displayable fields.

- The former `Balance` field has been renamed to `Effective`, meaning the effective balance (`Remaining - Reserved`).
- The `Amount` field has been removed as its meaning can be ambiguous.
- The `Remaining` field has been changed to mean the actual amount remaining in the allocation (independent of whether it is active).
- The `Balance` field is used in the same sense as it is in the transaction query, meaning the active allocation balance (if the allocation is active, it is the remaining allocation amount; if the allocation is inactive, it is zero).

Thus in glsfund and gbalance, Amount was replaced with Balance. In glsalloc, Amount was replaced with Remaining. These changes help to apply a more consistent meaning for these terms across the commands and objects.

Default Fields Have Changed For Some Commands

- The default fields for mybalance have changed to Name, Available.
- The default fields for gbalance have changed to Id, Name, Balance, Reserved, Effective, CreditLimit, Available.
- The default fields for glsfund have changed to Id, Name, Constraints, Allocated, Balance, DefaultDeposit, Description.
- The default fields for glstrans have changed to Id, Object, Action, Actor, Key, Child, Instance, Count, Amount, Delta, Balance, User, Account, Machine, Fund, Allocation, Usage.

If you would like to customize the default fields that these commands display for your site, uncomment and edit the appropriate gold.conf *.show parameters.

Initialization Scripts Have Been Synchronized With the Accounting Modes

The Moab Accounting Manager initialization scripts have been enhanced to set up sample environments that correspond to each of the four accounting modes.

- The former hpc-allocation-enforcement.sh script was renamed to hpc-strict-allocation.sh.
- The former hpc-notional-charging.sh script was enhanced to set the accounting.mode to notional-charging.
- Two new scripts were created for the other two modes (hpc-fast-allocation.sh and hpc-usage-tracking.sh).

Running these scripts is similar in effect to performing the actions in the respective Setup Guide chapters in the *Moab Accounting Manager Administrator Guide*.

Moab Web Services

8.1.2

No known differences.

8.1.1

No known differences.

8.1.0

Reservation Statistics Value Changes

MWS reservation statistics (CIPS, CAPS, TAPS, and TIPS) values have been changed from floating decimal points (double) to long integers. This supports reservations now passing the consumption rate as an attribute (instead of an element).

TORQUE Resource Manager

5.1.2

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.

Abnormal Job Exit States Combined Into Single Exit Code

Added parameter "exit_code_canceled_job" to force all canceled jobs to have the same exit code, regardless of exit state. Previously, exit codes would be inconsistent depending on the type of cancellation or abnormal exiting of each job. The "exit_code_canceled_job" gets rid of extraneous exit codes and allows users to set their own exit code. For example, if a user sets "exit_code_canceled_job = 300" in qmgr, the exit code for any job that exits with a non-zero status becomes 300.

Qmgr Support Added for "loglevel" Attribute

The "loglevel" attribute name was not recognized by qmgr, while it was recognized in momctl; the string "loglevel" is now equivalent to "log_level" in qmgr. The user can now use either of these attribute names to check the current value of the loglevel.

Node Health Checks Now Run on Sister Nodes

Previously, node health checks would exclusively run on the mother superior. Node health checks now run on sister nodes as well.

Submission Syntax Added to Prevent Mixing NCPUs and Nodes

When mixing NCPUs and nodes, the checkjob command would show available procs as a negative; this mixing is now prohibited.

5.1.1

\$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 than 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.

Disable the Automatic Requeuing of Jobs

Added the ability to disable the automatic requeuing of jobs due to transient failures.

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`.

Log Milliseconds

Added milliseconds in TORQUE's log files.

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_job_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.

5.1.0

CLIENTRETRY Configuration Option Support for qdel

The `qdel` parameter now includes the `-b` option and `CLIENTRETRY` configuration option support. This feature functions similar to `-b` option for `qsub`. Specifically:

- `-b <num>` command line argument support
- `CLIENTRETRY` configuration option support
- `PBS_CLIENTRETRY` environment variable support


Jobs Deleted When the Dependency Can No Longer Be Satisfied

When a job is deleted because its dependency can no longer be satisfied, that job will follow the `keep_completed` parameter set (if any) for `pbs_server`. Previously, jobs were purged immediately.

pbs_server -t No Longer Supports hot|warm|cold Options

The `pbs_server -t` option no longer supports the `hot|warm|cold` options. Other options are still supported.

Installation and Upgrade Information

 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.

- [Compatibility Requirements on page 19](#)
- [Installing Moab HPC Suite 8.1.x on page 19](#)
- [Upgrading to Moab HPC Suite 8.1.x on page 19](#)

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 8.1.x and TORQUE version 5.1.x, Moab version 8.1.x supports TORQUE version 4.2.9, 4.2.10, 5.0.x, and 5.1.x.

TORQUE 5.1.x requires Moab 8.1.x or 8.0.x.

Moab Accounting Manager

Moab Accounting Manager version 8.1.x is compatible *only* with Moab Workload Manager version 8.1.x.

If you are using Moab Accounting Manager with your current Moab solution, you will need to upgrade to the new Moab Accounting Manager 8.1.x at the same time that you upgrade to Moab Workload Manager 8.1.x.

Also in this release, the mam user became the new owner of MAM files and directories. If you have any custom scripts (including Moab Native scripts for MAM), these may need to be changed to be owned by the mam user when upgrading.

Installing Moab HPC Suite 8.1.x

Please see **Requirements** and also see **Preparing for Installation** and **Installing Suite RPM** for manual or RPM-based installation instructions, respectively, in the *Moab HPC Suite Installation and Configuration Guide*.

Upgrading to Moab HPC Suite 8.1.x

Please see **Preparing for Upgrade** and **Upgrading from 7.2 or Upgrading from 8.0** for manual or RPM-based installation instructions, respectively, in the *Moab HPC Suite Installation and Configuration Guide*.

Known Issues

The following are known issues in Moab HPC Suite – Enterprise Edition 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 first occurred or were reported.

8.1.2/5.1.2

No known issues.

8.1.1/5.1.1

- 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)
- Job user priority lost after running 'mjobctl -m userprio'. (MOAB-8094)
- Jobs do not exit the routing queue if the job is submitted with a hold (e.g. qsub -h). (TRQ-2788)

8.1.0/5.1.0

- 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)
- BACKFILLPOLICY BESTFIT does not support multi-req jobs. Only FIRSTFIT supports multi-req jobs. (MOAB-6824)
- ~~DNS caches are not purged of removed nodes when Elastic Computing is enabled. SLES 11 SP1 has an issue with giving the old IP address to TORQUE even after updating /etc/hosts. Do not use SLES SP1 if you are~~

~~using this method to manage the IP addresses for pbs_server. The mom's OSs are irrelevant. (TRQ-2765)~~ This issue is resolved in 5.1.1.

- Some limitations exist in the way that pbsdsh can be used. Please note the following situations are not currently supported:
 - Running multiple instances of pbsdsh concurrently within a single job. (TRQ-2851)
 - Using the -o and -s options concurrently; although requesting these options together is permitted, only the output from the first node is displayed rather than output from every node in the chain. (TRQ-2690)

Resolved Issues

The following is a list of some key bugs fixed in Moab HPC Suite – Enterprise Edition. Following each issue description is an associated issue number in parentheses.

Resolved issues are aggregated and grouped by the release version in which they were resolved.

8.1.2/5.1.2

- **Rounding of ConsumptionRate variable caused noticeable charge discrepancies.** MAM now receives an increased precision of the processor consumption rate value. (MOAB-7940)
- **When sending periodic reset calls, Moab got stuck and shut down incorrectly.** Added a 60 second shutdown alarm to avoid getting hung up during shutdown. (MOAB-8011)
- **OMAX* parameters were not recognized in the identity manager.** Enables setting OMAX* parameters in the identity manager. (MOAB-7567)
- **Wrong 'queuestatus' was shown for a blocked job.** Previously, if a job's individual walltime violated the set walltime, the checkjob xml would incorrectly display the job's queuestatus as eligible. Client commands now agree regarding queuestatus. (MOAB-8266)
- **Some job dependencies did not run before a job started.** "DEPEND:before" jobs no longer blocked when "mjobctl -m" parameter used before canceling job. (MOAB-8082)
- **Multiple instances of standing reservation were created when some nodes go down.** This issue is fixed. (MOAB-8042)
- **Under certain conditions a false "cancelfailed" email was sent to the user.** Canceled jobs that are not yet removed from the queue no longer block other jobs. (MOAB-7964)
- **Memory was divided incorrectly per task in array jobs.** Moab would commit too many jobs to a node when initializing an array job, and user specified memory limits would be exceeded. Moab no longer commits excessive memory with array jobs. (MOAB-7959)
- **Jobs were placed in incorrect partitions when one or more partition is down.** This issue is fixed. (MOAB-7103)
- **Moab redundantly pinned on ROLLBACKOFFSET reservations.** This issue is fixed. (MOAB-7942)
- **Moab would silently shut down with a valid license during a network outage.** This issue is fixed. (MOAB-7640)

- **If periodic charging was enabled on the grid head, extraneous charges would occur.** Under certain circumstances, if a job were created with multiple charges, the grid head would notice inconsistencies and show the job as migrated. This issue is fixed. (MOAB-7842)
- **Authentication failures occurred when using unicode characters with the gauth security promotion method.** Fixed authentication failures when using unicode characters with gauth *and* improved printing of unicode characters in client output. (MAM-315)
- **Server timed out to MOM when running a lot of small jobs.** Added load balancing to login nodes when login nodes begin to get busy. (TRQ-3367)
- **Numbered directories in server_priv/jobs (and arrays) were missing when the server attribute "use_jobs_subdirs" was set to TRUE.** This issue is fixed. (TRQ-3185)
- **Random group names displayed for users that 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)
- **Read timeouts were retried indefinitely by pbs_server.** This issue is fixed. (TRQ-3306)
- **End-of-job times were not consistent.** End-of-job times are now recorded closer to when nodes are free for use. (TRQ-2840)
- **Torque init scripts were not LSB compatible.** The pbs_mom service was not returning exit code 3 when exiting, which is an LSB requirement. As a result, the builder was failing to properly initialize pbs_mom, causing nose tests to fail. Init script return codes are now altered to correct values for LSB compatibility. (TRQ-3254)
- **Reporter MOM was unable to handle ALPS UNKNOWN role.** This issue is fixed. (TRQ-3245)
- **Jobs were stuck in EXITING state on mother superior.** This issue is fixed. (TRQ-3165)
- **Jobs with square brackets "[]" in their name were aborted on restart if they were not array subjobs.** Torque now allows all jobs that have "[]" in their name. (TRQ-3214)
- **Improperly escaped xml caused job files and dependencies to fail upon reload attempt.** This issue is fixed. (TRQ-3284)
- **Jobs were not created properly if the path had one or more spaces in the name.** Added multiple spacing options for job file path name. (TRQ-3235)
- **Completed jobs were not cleaned up properly.** Fixed various issues relating to restarting dependency jobs. (TRQ-3175)

- **Job files were never deleted when "\$thread_unlink_calls" value was set to TRUE.** Fixed issue where the threadpool in pbs_mom was not started properly causing job files to be left behind. (TRQ-3232)
- **Some threads would cause Torque to hang and become completely unresponsive.** Count_proc() now returns 0 on a NULL argument. (TRQ-3196)
- **Jobs were run in the wrong order and canceled prematurely.** Running jobs are no longer deleted due to dependencies. (TRQ-3189)
- **Server entered deadlock around completed jobs map.** This issue is fixed. (TRQ-3226)
- **Flooding server with client commands disabled some functionality.** Resolved issue where server gets stuck due to large volume of client requests. (TRQ-3296)
- **Kill_delay sent multiple signals.** Signals sent to job pids are now tracked to eliminate redundant sending. (TRQ-3239)
- **Qsub -W stage-in displayed incorrect user grouping.** The log was showing users belonging to groups that the users were not a part of. This issue is fixed. (TRQ-3312)
- **pbs_submit() method changed for stable job submissions.** The qsub command currently utilizes the pbs_submit_hash() method to submit jobs and is stable. Submitting jobs via the pbs_submit() method would occasionally render pbs_server unresponsive. Now pbs_submit() is functioning properly. (TRQ-3314)
- **Moab crashes in Torque API.** This issue is fixed. (TRQ-3368)

8.1.1/5.1.1

- **glscnfig was installed with incorrect permissions.** Corrected installed file permissions for glscnfig. (MAM-310)

 The glscnfig fix requires the 8.1.1.1 maintenance release or later.

- **Moab failed to register GRES update via qalter.** Fix applied to update changes made to job gres via qalter. (MOAB-7559)
- **showhist.moab.pl was missing output file and working directory in the output.** Added ability to display the output file and working directory in the output of showhist.moab.pl. (MOAB-5609)
- **mnodectl -m features with regex only updates one node.** Enabled mnodectl -m <features> x:<node_regex> for node features. (MOAB-7843)

- **Moab is calling salloc/srun with wrong options.** Changed call for Slurm's srun command from "-n1 -N1" to "-n<num_tasks>". (MOAB-6770)
- **mjobctl 'command-line arg info not available' is logged.** Added "starttime" to the event logged when either the 'showstart' or 'mjobctl -q starttime' command is run. (MOAB-7230)
- **Group missing from showq -b.** Added GROUP to output of "showq -b" and "showq -b -v". (MOAB-6762)
- **Unable to disable a RM that is down.** Added ability to set state of down RM to disabled. (MOAB-7481)
- **Unable to clear failures for ID in mdiag -R output.** Added an ID option to mrmctl -f to clear the ID failure messages in mdiag -R. (MOAB-7259)
- **Moab was reading in old job information and using it.** Added log warning if job id from RM is already found in the completed job table. (MOAB-7196)
- **mysql error code was missing in the list of ignored SQL error codes.** This issue is fixed. (MOAB-7758)
- **runjob ignores policies.** Added the enforcing of partition limits to runjob (mjobctl -x). (MOAB-6132)
- **Job submission rejected for insufficient resources yet still created template jobs.** Cleaned up job workflows that could not be submitted. (MOAB-7919)
- **Moab peer to peer grid with LOCALWORKLOADEXPORT results in livelock.** Enabled "RMCFG[] TYPE=MOAB FLAGS=asynccdelete" for more responsive grids. (MOAB-7415)
- **GPU usage did not show up in workload trace.** This issue is fixed. (MOAB-7498)
- **Potential race condition when node idle purge time exceeded.** Enabled automatic node draining when NODEIDLEPURGETIME is reached. (MOAB-7663)
- **Moab not escaping properly when sending XML to MAM.** Enabled sanitizing of credential names. (MOAB-7419)
- **Erroneous message reported about classes when nodes are down.** This issue is fixed. (MOAB-7770)
- **Moab log is incorrect.** Fixed log message about nodes located for job. (MOAB-7788)

- **"could not get user info" message displays for job triggers with flag "user"**. Set job trigger exec user to default user of job if not explicitly set. (MOAB-7907)
 - **Checkjob reporting misleading error**. Suppressed non-essential error output when unable to connect to the CLIENTUIPORT. (MOAB-7539)
 - **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**. This issue is fixed. (MOAB-7748)
 - **Moab was scheduling jobs before setting up the rsv event table**. This issue is fixed. (MOAB-7953)
 - **MPBSWorkloadQuery failure caused problems with jobs**. Enabled RMCFG[] FLAGS=NoCondensedQueries to disable Torque's condensed qstat queries. (MOAB-7958)
 - **Problems reported with credential REST queries**. Changed max_idle_jobs, max_jobs, max_nodes, max_processors, and max_processor_seconds from integer to string. (WS-2388)
 - **Liens can be left around when a non-running job is removed and if threaded accounting has been enabled for job starts**. This issue is fixed. (MOAB-7746)
 - **Charges were not attempted when Moab is started but the accounting manager is not yet available**. Modified MAM state behavior so that Moab will always attempt charges even if the accounting manager is unavailable at startup.
 - **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. This issue is fixed. (MOAB-7389)
 - **Using MySQL database, the MAM migration process could create Currency or Float data types as floating point rather than double precision**. This resulted in a potential loss of precision for very large values. (MAM-304)
- i** If your site uses the MySQL database for MAM and has used a migration script to upgrade, contact support to see if you are affected and for assistance to correct the problem. PostgreSQL is the preferred DBMS for MAM. See **Migrating the MAM Database from MySQL to PostgreSQL** in the *Moab HPC Suite Installation and Configuration Guide*.
- **Queue default resources were being modified**. Corrected an issue where the all memory values were converted to bytes after a restart. (TRQ-3139)

- **Some completed jobs were taking quite a while to clear out.** Hardened the removal of completed jobs from pbs_server. (TRQ-3044)
- **Ability to have the procct values displayed on the queues where it was set was lost.** This issue is fixed. (TRQ-3135)
- **max_user_queueable issues.** Fixed some bugs causing incorrect max_user_queueable values. (TRQ-2841)
- **PBS_NUM_NODES on multi-req job wrong when using specific hostnames.** This issue is fixed. (TRQ-1949)
- **Dependency jobs fail to start during HA fail over.** Dependencies now display in the format of depend=type:jobid instead of depend=type:jobid@server. (TRQ-2332)

```
# Was
depend=afterok:12.napali@napali

# Now
depend=afterok:12.napali
```

- **Bug reported that caused jobs to not start when ALPS incorrectly returns a permanent confirmation failure.** This issue is fixed. (TRQ-3023)
- **Problem reported that caused MOM restarts to intermittently fail.** This issue is fixed. (TRQ-2307)
- **TORQUE will not compile with --enable-debug on configure.** This issue is fixed. (TRQ-2969)
- **pbs_server would count completed jobs against queue limits when pbs_server was restarted.** This issue is fixed. (TRQ-3087)
- **Compute node offline state doesn't persist across server restarts.** This issue is fixed. (TRQ-2790)
- **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.** This issue is fixed. (TRQ-2993)
- **qmgr refuses numerical node ids.** This issue is fixed. (TRQ-2946)
- **X11 very slow with qsub -I -X.** The port forwarding buffer size has been increased to improve performance when enabling X11 forwarding from qsub (-X switch). (TRQ-2740)
- **Problem reported with interactive jobs not staying on the node from which they were submitted.** This issue is fixed. (TRQ-3122)
- **Jobs were getting stuck in a running state when an asynchronous run failed.** This issue is fixed. (TRQ-3114)
- **Array slot limits were not getting decremented when a job is preempted or rerun.** This issue is fixed. (TRQ-3110)

8.1.0/5.1.0

- **mrmctl -f {messages|stats} failed to work for AM.** Fixed bug so that messages and statistics can be cleared for AM as described in the documentation.
- **Several mrmctl options ignored the option argument or did not honor them properly.** These issues are fixed.
- **A job could charge the wrong allocation if it had no lien and was out of funds but another lien was present.** This issue is fixed. (GOLD-267)
- **Jobs were getting final charges made twice.** Added a fix to prevent Moab from making final charge twice even if resource manager reports multiple completions. (MOAB-7565)
- **Extra zero charges on some jobs.** Added a fix to prevent an extraneous zero charge for a canceled idle job after it has been requeued. (MOAB-7637)
- **Role permission updates were incorrectly applied to user permissions.** This issue is fixed. (WS-2340)
- **multireq jobs take hours to start.** Fix bug where multi-req jobs were slow to start in certain cases. (MOAB-6824)
- **Unauthorized error does not tell you what permission you are lacking.** Improved error messages related to accessing resources without sufficient permissions. (WS-2301)
- **Submitting remote job scripts (that is, including the script as part of the JSON payload) is not currently supported.** Job scripts can now be included in the POST body when submitting jobs via MWS. The new field is called commandScript. (WS-2112)
- **MWS could send phony node name to Moab.** MWS is now more robust when parsing the nodesRequested field during a job PUT. (WS-2352)
- **Reservation trigger parser is broken.** The rest/reservations resource now correctly returns trigger IDs (if any) on reservations. (WS-2342)
- **Jobs purged within Moab could leave a disconnected reserve usage record in MAM.** Fixed issue to prevent the extraneous lien delete ensuring a combined usage record. (MOAB-7353)
- **HOST_NAME_SUFFIX was no longer adding suffix to job names.** This issue is fixed. (TRQ-2956)
- **Migration scripts could have a PATH problem when using promotion method gauth.** This issue is fixed.

- **Incorrect proportions could be charged to expired allocations (but the total amount charged would be correct).** This issue is fixed. (MAM-262)
- **Some systems could get "Insecure \$ENV{ENV} while running with -T switch" when using the gauth security promotion mechanism.** This issue is fixed. (MAM-276)
- **Split charges could occur, or liens could be denied, because charges, liens, and quotes were factoring in inactive liens.** This issue is fixed. (MAM-274)
- **Auto-generated objects that had default values for fields (such as the Active field for accounts) were not being set to their default values when auto-generated via a scheduling action (e.g. a job charge).** This issue is fixed.
- **The column sort up arrow in the MAM GUI list view was broken/missing.** This issue is fixed.
- **The MAM GUI lookup buttons did not work in Internet Explorer.** This issue is fixed.
- **Issues reported related to the incorrect modification or removal of the mam pid file that resulted in the inability to use goldd to shutdown the MAM server.** These issues are fixed. (MAM-287)
- **pbs_mom filling up the logs in a HA environment.** Reduced verbosity in error logging in HA environments. (TRQ-2863)
- **Delays reported with Preemption.** Speed up process data collection to help diminish time to rerun a job, including correctly sum job memory and cpu time resources for processes created under setsid() calls within a job, and avoid an infinite loop in pbs_mom when running under a Linux Container (lxc). (TRQ-3026)