

# Moab HPC Suite – Enterprise Edition 8.1.0 Release Notes

The release notes file contains the following sections:

- [New Features on page 1](#)
- [Differences on page 5](#)
- [Installation and Upgrade Information on page 10](#)
- [Known Issues on page 12](#)
- [Resolved Issues on page 13](#)
- [Key Documentation Changes on page 15](#)

## New Features

The following is a summary of key new features in Moab HPC Suite – Enterprise Edition 8.1.0.

- [Moab Workload Manager on page 1](#)
- [Moab Web Services on page 3](#)
- [Moab Accounting Manager on page 3](#)
- [Moab Viewpoint on page 4](#)
- [Moab Insight on page 4](#)
- [TORQUE Resource Manager on page 4](#)

### Moab Workload Manager

*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 5](#) 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 (i.e., "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 Web Services**

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

MWS now supports multi-line (textarea) configuration parameters in plugins. See Configuration Constraints in Plugins\Plugin Developer's Guide in *Moab Web Services Administrator Guide* for more information.

### *MWS-to-Moab Insight Database Queries*

A new MWS resource has been created that allows privileged and user queries to the Moab Insight database. MWS queries the Insight database view (or table) and returns the result in JSON. Users permitted to run the privileged query receive a list of all rows that match the query's parameters, and users who run the user query receive all requested rows that also have a `user_name` column value that matches their username. This new resource allows database queries using a MongoDB-like query syntax that gets converted to SQL. It supports SQL features such as where clauses, group by fields, aggregate functions, and having clauses. Results can be sorted and paged. See Insight Database Configuration using `mws-config.groovy` in Setup\Configuring Moab Web Services and see Insight Database in Resources in *Moab Web Services Administrator Guide* for 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.

## **Moab Accounting Manager**

### *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 Viewpoint

Viewpoint version 8.1 is has been rewritten from the ground up to focus on HPC use cases. This reconfiguration enables better integration with Moab for managing jobs, resources, and utilization via a customizable portal. Settings are also provided to limit the amount of data that is displayed. See Threshold Settings in the *Moab Viewpoint Administrator Guide* for more information.

Viewpoint can be installed on any head node. However it is recommended that you install Viewpoint on the same server as Moab Web Services (if possible).

**i** The reconfiguration for version 8.1 is not compatible with prior versions of Viewpoint. See Compatibility Requirements in [Installation and Upgrade Information](#) on page 10 for more information.

**i** Moab Viewpoint is only available with a Moab HPC Suite - Enterprise Edition license.

## Moab Insight

The Moab HPC Suite - Enterprise Edition version 8.1 introduces the Insight database as a new component. Insight receives relevant job information from Moab and enables Moab Viewpoint to display this data for the purpose of empowering HPC administrators with information utilization, workload, and cluster performance. Currently Insight will support 50K weekly job submissions and 400 nodes. Job submission rates or cluster sizes beyond these numbers are not supported at this time.

Install Insight if you want to do either of the following:

- Use Moab Viewpoint 8.1
- Run reports and analyze events within the cluster using standard relational database tools such as Crystal Reports.

Insight must be installed on a different server than Moab.

**i** Insight is not backwards-compatible with other suite versions.

**i** Moab Insight is only available with a Moab HPC Suite - Enterprise Edition license.

## TORQUE Resource Manager

*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.

- [Moab Workload Manager on page 5](#)
- [Moab Web Services on page 8](#)
- [Moab Accounting Manager on page 8](#)
- [TORQUE Resource Manager on page 10](#)

### Moab Workload Manager

#### *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[%s] 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 Appendix L in the *Moab Workload Manager Administrator Guide* for more information on configuring power management and tracking energy consumption.

## **Moab Web Services**

### *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).

## **Moab Accounting Manager**

### *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.

## TORQUE Resource Manager

### *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 (Appendix E: Security) in the *Moab Workload Manager Administrator Guide* for more information.

- [Compatibility Requirements](#) on page 11
- [Installing Moab HPC Suite - Enterprise Edition 8.1.0](#) on page 11

- [Upgrading to Moab HPC Suite – Enterprise Edition 8.1.0](#) on page 11

## Compatibility Requirements

This section provides information on compatibility between the difference components of the suite.

### *Moab Workload Manager and TORQUE Resource Manager*

Although the recommended configuration is Moab version 8.1 and TORQUE version 5.1, Moab version 8.1 supports TORQUE version 4.2.9, 5.0.x and 5.1. TORQUE 5.1 requires Moab 8.1 or 8.0.x.

### *Moab Accounting Manager*

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

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 at the same time that you upgrade to Moab Workload Manager 8.1.

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.

### *Viewpoint*

Viewpoint version 8.1 has a new, HPC-based configuration. See Viewpoint in [New Features on page 1](#) for more information.

Because prior versions of Viewpoint are not compatible with version 8.1, you will need to completely uninstall the older version before installing Viewpoint 8.1.

**i** Customized data from a prior version will be lost. If you want to retain customized data, you will need to back it up prior to uninstalling.

## Installing Moab HPC Suite - Enterprise Edition 8.1.0

See the *Moab HPC Suite Installation and Configuration Guide* for requirements and instructions.

## Upgrading to Moab HPC Suite – Enterprise Edition 8.1.0

See the *Moab HPC Suite Installation and Configuration Guide* for requirements and instructions.

# Known Issues

The following are known issues in Moab HPC Suite - Enterprise Edition 8.1.0. Following each issue description is an associated issue number in parentheses.

- 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 (Appendix E: Security) in the *Moab Workload Manager Administrator Guide* for more information. (CVE-2014-5376, MOAB-7525, MOAB-7480)
- If the Moab **JOBNODEMATCHPOLICY** is set to **EXACTNODE**, and if `requirements.tasksPerNode` is used in a job submission to MWS, then Moab will double the resources requested. To avoid this problem, use `requirements.resourcesPerTask.processors.dedicated` instead. (MOAB-7424)
- 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)
- 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 8.1.0. Following each issue description is an associated issue number in parentheses.

- **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.** The issues have been remedied.
- **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 bug 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 gold to shutdown the MAM server.** The 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.** The following changes were made to reduce delays: (TRQ-3026)
  - Speed up process data collection to help diminish time to rerun a job.
  - Correctly sum job memory and cpu time resources for processes created under setsid() calls within a job.
  - Avoid an infinite loop in pbs\_mom when running under a Linux Container (lxc).

# Key Documentation Changes

This section lists some key documentation changes made for Moab HPC Suite – Enterprise Edition 8.1.0.

Features new to this release have been documented but are not listed on this page. For information about those new features and links to their respective documentation, see [New Features on page 1](#).

- [Moab Suite on page 15](#)
- [TORQUE Resource Manager on page 15](#)

## Moab Suite

### *Moab Workload Manager*

- New chapter for Elastic Computing
- New appendix L for Cray-Specific Power Management and Energy-Consumption-by-Job Accounting

### *Moab Accounting Manager*

- New chapters, one for each accounting mode (strict-allocation, fast-allocation, notional-charging, usage-tracking)
- Removed Appendix B Accounting Modes as the information has been updated and reorganized into the new accounting mode chapters

## TORQUE Resource Manager

No significant documentation changes to report.