Moab HPC Suite – Basic 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.

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.

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 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.

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

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).

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

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 Web Services

9.0.0

No known differences.

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.

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 Web Services

9.0.0

No known issues.

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)
- 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)
- 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)
- mnodectl -m features with regex only updates one node. Enabled mnodectl -m <*features*> x:<*node_regex*> for node features. (MOAB-7843)
- 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 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)

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)