

# Moab

## Installation and Configuration Guide 9.1.1 for SUSE 11- Based Systems

April 2017



© 2017 Adaptive Computing Enterprises, Inc. All rights reserved.

Distribution of this document for commercial purposes in either hard or soft copy form is strictly prohibited without prior written consent from Adaptive Computing Enterprises, Inc.

Adaptive Computing, Cluster Resources, Moab, Moab Workload Manager, Moab Viewpoint, Moab Cluster Manager, Moab Cluster Suite, Moab Grid Scheduler, Moab Grid Suite, Moab Access Portal, and other Adaptive Computing products are either registered trademarks or trademarks of Adaptive Computing Enterprises, Inc. The Adaptive Computing logo and the Cluster Resources logo are trademarks of Adaptive Computing Enterprises, Inc. All other company and product names may be trademarks of their respective companies.

Adaptive Computing Enterprises, Inc.  
1712 S. East Bay Blvd., Suite 300  
Provo, UT 84606  
+1 (801) 717-3700  
[www.adaptivecomputing.com](http://www.adaptivecomputing.com)



*Scan to open online help*

<b>Welcome</b> .....	<b>1</b>
<b>Chapter 1 Planning Your Installation</b> .....	<b>3</b>
Server Hardware Requirements .....	4
Component Requirements .....	13
RLM Licensing Model .....	19
Identify The Installation Methods .....	23
<b>Chapter 2 Manual Installation</b> .....	<b>25</b>
Manual Installation .....	26
Preparing For Manual Installation .....	26
Installing Torque Resource Manager .....	28
Installing Moab Workload Manager .....	35
Installing Moab Accounting Manager .....	43
Installing RLM Server .....	53
Nitro Integration .....	56
Preparing For Nitro Manual Installation .....	56
Installing Nitro .....	57
Additional Configuration .....	63
Opening Ports In A Firewall .....	63
Moab Workload Manager Configuration Options .....	63
Moab Accounting Manager Configuration Options .....	65
Using Multiple RLM Servers .....	66
Running Multiple Coordinators On The Same Node .....	67
Manual Upgrade .....	68
Upgrading Torque Resource Manager .....	68
Upgrading Moab Workload Manager .....	75
Upgrading Moab Accounting Manager .....	77
Upgrading RLM Server .....	82
Upgrading Your Nitro Integration .....	83
Preparing For Nitro Manual Installation .....	83
Upgrading Nitro .....	84
Migrating The MAM Database From MySQL To PostgreSQL .....	86
<b>Chapter 3 Troubleshooting</b> .....	<b>89</b>
General Issues .....	89
Moab Workload Manager Issues .....	95




# Welcome

Revised: 04/26/2017

Welcome to the 9.1.1 Moab HPC Suite Installation and Configuration Guide for SUSE 11-Based Systems.

This guide includes detailed instructions for installing each component of the suite so that you can quickly get up and running.

This guide is intended for system administrators who are responsible for installing the Moab HPC Suite components.

 Depending on your system configuration and license, not all of the HPC Suite components may be available.


The 9.1.1 Moab HPC Suite contains the following components for SUSE 11-based systems:

- Torque Resource Manager 6.1.1.1
- Moab Workload Manager 9.1.1
- Moab Accounting Manager 9.1.1
- Nitro 2.1.1; Nitro Web Services is not available for a SUSE 11-based system.
- Reprise License Manager 12.1.2


Before commencing the installation or upgrade, please see [Chapter 1 Planning your Installation on page 3](#) to verify your system conforms to minimum prerequisites.



## Chapter 1 Planning your Installation

 It is highly recommended that you *first* perform installations and upgrades in a *test environment*. Standard installation and upgrade procedures and use cases are tested prior to release. However, due to the wide range of possible configurations and customizations, it is important to exercise caution when deploying new versions of software into your production environments. This is especially true when the workload has vital bearing on your organization's day-to-day operations. We recommend that you test in an environment that mirrors your production environment's configuration, workflow and load as closely as possible. Please contact your Adaptive Computing account manager for suggestions and options for installing/upgrading to newer versions.

There are many different ways to install and configure the Moab HPC Suite. Each environment has its own set of requirements and preferences. This chapter is intended to help an administrator understand how each of the Moab HPC Suite components interact, basic requirements and configuration information to prepare for the installation.

 Code samples have been provided for convenience. Some code samples provide sample passwords (i.e. "changeme!"). We strongly recommend that you do not use these passwords during installation, as using the documented passwords could introduce unnecessary security vulnerabilities into your system.

In this chapter:

- [Installation Terminology on page 3](#)
- [Where to Start on page 4](#)
- [Server Hardware Requirements on page 4](#)
- [Identify the Installation Methods on page 23](#)
- [Component Requirements on page 13](#)

### Installation Terminology

To aid in documentation clarity, Adaptive Computing uses the following terms in this Installation and Configuration Guide:

- Components – The different "products" included in the Moab. For example, Moab Workload Manager, Moab Web Services.

- Servers – Also known as components, but specifically relating to the actual services. For example, the Moab Workload Manager component is referred to as the Moab Server for non-client services.
- Host – The actual box where an Moab component (server or client) is installed.

**i** Previous documentation typically used Head Node to designate a host or a Server.

## Where to Start

You will need to plan your environment and determine how many hosts you will need and for which you components you will install using the Manual Installation or the RPM Installation method. The following are suggested steps to help you in your planning and installing process.

1. Determine whether you have a small, medium, High-Throughput or large environment; including an example, and required and recommended hardware requirements. See [Server Hardware Requirements on page 4](#).
2. Decide whether you will perform a Manual Installation or an RPM Installation for the various components. See [Identify the Installation Methods on page 23](#).

**i** The Manual Installation and the RPM Installation chapters each have an "Additional Configuration" section that provides additional information and instructions for optional, but recommended configurations (for example, Configuring SSL in Tomcat).

3. Review the software requirements for your components and set up your hosts accordingly. See [Component Requirements on page 13](#).
4. Install the individual components on their respective host(s). See [Preparing for Manual Installation on page 26](#) or [1.1 About RPM Installations and Upgrades](#) as applicable.
5. Refer to [Chapter 3 Troubleshooting on page 89](#) for assistance in addressing common problems during installation and configuration.

## Server Hardware Requirements

The Moab is installed and configured differently for small, medium, or large environment types. This topic provides a general topology of the Moab HPC Suite and the server hardware requirements depending on your environment size.

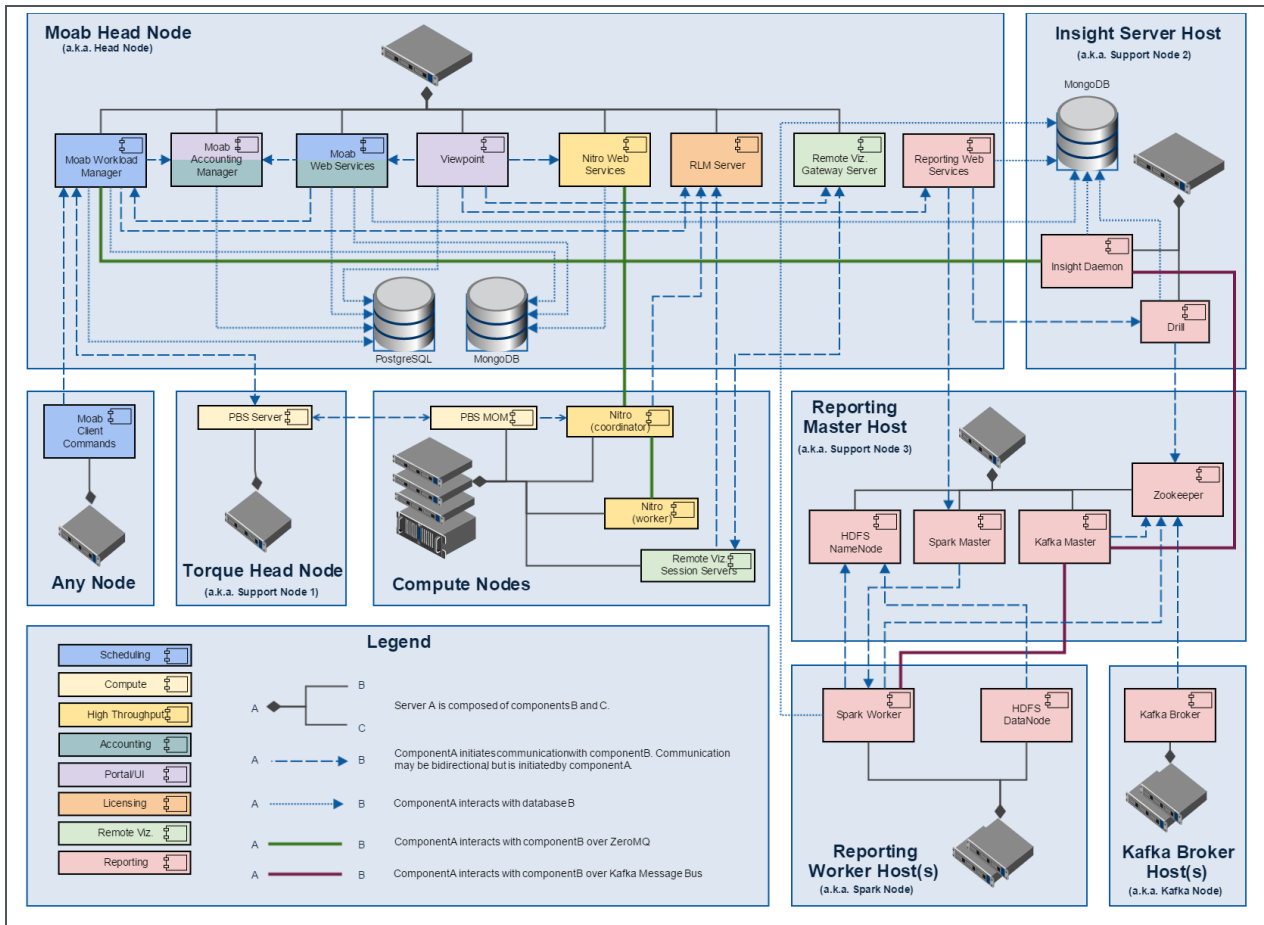
In this topic:



- [Topology on page 5](#)
- [Hardware Requirements on page 8](#)

## Topology

The following diagram provides a general topology of the Moab HPC Suite for a medium (with high throughput) or a large environment.



Please note the following:

- Smaller environments may elect to consolidate the Torque Server with the Moab Server on the same host, including PBS Server in the list of components installed on the same host.
- Although Moab Workload Manager and Moab Accounting Manager may share the same database instance, it is not a requirement. Two database instances may be used, one for each component.
- Larger systems will require more dedicated resources for each component, in which case it may be necessary to move individual components from the Moab Server Host (i.e. databases, Moab Accounting Manager, and/or Viewpoint) to their own respective servers.

Software components that may be included in a Moab HPC Suite installation are described in the table below.

Component	Description
<b>Moab Workload Manager</b>	A scheduling and management system designed for clusters and grids.
<b>Moab Elastic Computing</b>	Manages resource expansion and contraction of bursty workloads utilizing additional resources from private clouds or other data centers.
<b>Torque Resource Manager - PBS Server</b>	A resource manager for Moab. Torque provides the low-level functionality to discover and report cluster resources/features, and to start, hold, cancel, and monitor jobs. Required by Moab Workload Manager.
<b>Torque Resource Manager - PBS MOM</b>	Torque MOMs are agents installed on each compute node that complete tasks assigned to them by the Torque Server. When a multi-node job runs, one of the Torque MOMs is assigned the role of Mother Superior and all other nodes assigned to the job are sister nodes. Mother Superior manages the job across all the sister nodes by communicating with each of them and updating the Torque Server. Required by Torque.
<b>Moab Passthrough</b>	Enables job submission and monitoring with Slurm.
<b>Slurmd</b>	The compute node daemon of Slurm. It monitors all tasks running on the compute node, accepts work (tasks), launches tasks, and kills running tasks upon request. The Automated Installer does not install slurmd at this time. Slurmd is assumed to already be installed.
<b>Moab Accounting Manager</b>	An accounting management system that allows for usage tracking, charge accounting, and allocation enforcements for resource usage in technical computing environments. Required by Moab Workload Manager and Moab Web Services.
<b>Moab Web Services (MWS)</b>	A component of the Moab HPC Suite that enables programmatic interaction with Moab Workload Manager via a RESTful interface. MWS lets you create and interact with Moab objects and properties such as jobs, nodes, virtual machines, and reservations. MWS is the preferred method for those wishing to create custom user interfaces for Moab and is the primary method by which Moab Viewpoint communicates with Moab. Required by Moab Viewpoint.
<b>Reprise License Manager Server (RLM)</b>	A flexible and easy-to-use license manager with the power to serve enterprise users. Required by Moab Elastic Computing, Nitro, and Remote Visualization.

Component	Description
<b>Moab Insight</b>	A component of the Moab HPC Suite that collects the data that Moab emits on its message queue and stores it in a database. The message queue is efficient, can be encrypted, and tolerates disconnections and restarts on either side. Required by Moab Viewpoint and Kafka Master.
<b>Nitro</b>	A highly powerful, yet simple task launching solution which operates as an independent product but can also integrate seamlessly with any HPC scheduler. In the Moab HPC Suite, Nitro is fully integrated with Viewpoint for seamless high-throughput job submission and monitoring.
<b>Nitro Web Services</b>	Enables programmatic interaction with Nitro for obtaining Nitro job status information within Viewpoint. Required by Moab Viewpoint.
<b>Moab Viewpoint</b>	A rich, easy-to-use portal for end-users and administrators, designed to increase productivity through its visual web-based interface, powerful job management features, and other workload functions. The portal provides greater self-sufficiency for end-users while reducing administrator overhead in High Performance Computing. Nitro, Remote Visualization, Elastic Computing, Moab Passthrough, and Reporting and Analytics features are also licensable for use with Viewpoint. Required by Remote Visualization.
<b>Remote Visualization Gateway</b>	Manages Remote Visualization sessions on the Remote Visualization Session servers. Remote Visualization is an extension of Viewpoint. Required by Viewpoint and Remote Visualization.
<b>Remote Visualization Session</b>	Remote Visualization sessions provide access into remote applications, rendering remotely and transferring the pixels to the local browser. Required by Viewpoint and Remote Visualization Gateway.
<b>Reporting Web Services (RWS)</b>	A component of Adaptive Computing Suites that enables programmatic interaction with Moab Reporting and Analytics via a RESTful interface. RWS is the preferred method for those wishing to create custom user interfaces for Moab Reporting and Analytics and is the primary method by which Moab Viewpoint communicates with Moab Reporting and Analytics.
<b>Reporting and Analytics</b>	Streams in massive amounts of workload and resource usage data from your High Performance Computing (HPC), High Throughput Computing (HTC) and Grid Computing environments, and then correlates that information against users, groups, and accounts, organizations so you can gain insights into exactly how your investment is being used and how well it aligns with your goals.
<b>MongoDB</b>	A free and open-source cross-platform document-oriented database program. Classified as a NoSQL database program, MongoDB uses JSON-like documents with schemas. Required by Moab Workload Manager, Moab Passthrough, Moab Web Services, Nitro Web Services, Reporting Web Services, and Spark Worker.

Component	Description
<b>PostgreSQL</b>	An object-relational database (ORDBMS) – i.e. an RDBMS, with additional (optional use) object features – with an emphasis on extensibility and standards compliance. Required by Moab Workload Manager, Moab Passthrough, Moab Accounting Manager, Moab Web Services, and Moab Viewpoint.
<b>Drill</b>	Apache Drill is an open-source software framework that supports data-intensive distributed applications for interactive analysis of large-scale datasets. Required by Reporting Web Services.
<b>Hadoop</b>	The Apache Hadoop software library is a framework that allows for the distributed processing of large data sets across clusters of computers using simple programming models. It is designed to scale up from single servers to thousands of machines, each offering local computation and storage. Rather than rely on hardware to deliver high-availability, the library itself is designed to detect and handle failures at the application layer, so delivering a highly-available service on top of a cluster of computers, each of which may be prone to failures. Required by Spark Worker.
<b>Spark Master</b>	Apache Spark is a fast and general engine for large-scale data processing. Spark Streaming is an extension of the core Spark API that enables scalable, high-throughput, fault-tolerant stream processing of live data streams. The Spark Master uses one or more Spark Workers when processing live data streams. Data can be ingested from many sources like Kafka, Flume, Kinesis, or TCP sockets, and can be processed using complex algorithms expressed with high-level functions like map, reduce, join and window. Finally, processed data can be pushed out to filesystems, databases, and live dashboards. Required by Reporting Web Services.
<b>Spark Worker</b>	The Spark Worker is used by a Spark Master when processing live data streams. Required by Spark Master.
<b>Kafka Master</b>	Apache Kafka is used for building real-time data pipelines and streaming apps. It is horizontally scalable, fault-tolerant, wicked fast, and runs in production in thousands of companies. Kafka Master uses one or more Kafka Brokers when pipelining and processing live data streams. Required by Spark Worker, and Insight.
<b>Kafka Broker</b>	Kafka Broker is used by a Kafka Master to pipeline and process live data streams. Apache Kafka is used for building real-time data pipelines and streaming apps. It is horizontally scalable, fault-tolerant, wicked fast, and runs in production in thousands of companies. Required by Kafka Master.

## Hardware Requirements

The following tables show hardware requirements for Moab, Torque, and Reporting Framework environments of various deployment sizes.

## Moab and Torque Requirements

The following table identifies the minimum and recommended hardware requirements for the different environment types. Use this table as a guide when planing out your suite topology.

**i** Software requirements are listed per-component rather than suite-wide as the suite components reside on different hosts. See [Component Requirements on page 13](#)

Environment Type	# of Compute Nodes	Jobs/Week	Minimum Requirements (per Host Distribution)	Recommended Requirements (targeting minimum number of hosts)
Proof of Concept / Small Demo	50	<1k	<p><b>Moab Server+Torque Server Host</b></p> <ul style="list-style-type: none"> <li>• 4 Intel/AMD x86-64 cores</li> <li>• At least 8 GB RAM</li> <li>• At least 100 GB dedicated disk space</li> </ul> <p><b>Insight Server Host</b></p> <ul style="list-style-type: none"> <li>• 8 Intel/AMD x86-64 cores</li> <li>• At least 16 GB RAM</li> <li>• At least 512 GB dedicated disk space</li> </ul>	Same as minimum

Environment Type	# of Compute Nodes	Jobs/Week	Minimum Requirements (per Host Distribution)	Recommended Requirements (targeting minimum number of hosts)
Medium	500	<100k	<p><b>Moab Server+Torque Server Host</b></p> <ul style="list-style-type: none"> <li>• 8 Intel/AMD x86-64 cores</li> <li>• At least 16 GB RAM</li> <li>• At least 512 GB dedicated disk space</li> </ul> <p><b>Insight Server Host</b></p> <ul style="list-style-type: none"> <li>• 8 Intel/AMD x86-64 cores</li> <li>• At least 16 GB of RAM</li> <li>• At least 1024 GB disk</li> </ul>	<p><b>Moab Server+Torque Server Host</b></p> <ul style="list-style-type: none"> <li>• 16 Intel/AMD x86-64 cores</li> <li>• At least 32 GB RAM</li> <li>• At least 1 TB dedicated disk space</li> </ul> <p><b>Insight Server Host</b></p> <ul style="list-style-type: none"> <li>• 8 Intel/AMD x86-64 cores</li> <li>• At least 16 GB of RAM dedicated 1 Gbit channel between Insight and Moab</li> <li>• 128 GB local SSD for swap</li> <li>• At least 1024 GB disk</li> </ul>

Environment Type	# of Compute Nodes	Jobs/Week	Minimum Requirements (per Host Distribution)	Recommended Requirements (targeting minimum number of hosts)
Medium with High Throughput or Larger	>500	>100k	<p><b>Moab Server Host</b></p> <ul style="list-style-type: none"> <li>• 8 Intel/AMD x86-64 cores</li> <li>• At least 16 GB RAM</li> <li>• At least 512 GB dedicated disk space</li> </ul> <p><b>Torque Server Host</b></p> <ul style="list-style-type: none"> <li>• 8 Intel/AMD x86-64 cores</li> <li>• At least 16 GB RAM</li> <li>• At least 512 GB dedicated disk space</li> </ul> <p><b>Insight Server Host</b></p> <ul style="list-style-type: none"> <li>• 8 Intel/AMD x86-64 cores</li> <li>• At least 16 GB of RAM</li> <li>• At least 2048 GB disk</li> </ul>	<p>The Moab Server should <i>not</i> reside on the same host as the Torque Server.</p> <p>MWS Server <i>must</i> reside on the same host as the Moab Server (Moab Server Host).</p> <p>The MAM Server may reside on its own host, on the Moab Host (preferred), or another server's host (except for the Insight Host).</p> <p>The Viewpoint Server may reside on its own host, on the Moab Server Host (preferred), or another server's host (except for the Insight Server Host).</p> <p>Databases may also reside on the same or a different host from its server component.</p>

Please note the following:

- All requirements above (minimum and recommended) target a minimum number of management servers. Administrators are encouraged to separate the Torque Server and the Moab Server onto different hosts where possible for better results; especially when High Throughput is enabled.
- Although many factors may have an impact on performance (network bandwidth, intended use and configuration, etc.), we consider High Throughput as something that makes a significant enough difference

between minimum and recommended hardware requirements to merit mention in the table above.

- Moab and Torque are both multi-threaded and perform better with more processors.
- Due to the large amount of data Moab must send to Insight, Moab performs better without Insight enabled (for environments that do not require Viewpoint, or use Crystal Reporting).
- Regarding disk space, consideration should be given to requirements related to log files, log depth, number of jobs/nodes/reservations (more objects impact database journal size), average number of events generated (more events take more space), etc.

## Reporting Framework Requirements

The following table shows hardware requirements for the Reporting and Kafka hosts needed to support the addition of the Reporting Framework to a Moab environment. These requirements are *in addition* to the requirements shown in the table above.

Environment Type	Minimum Requirements (per Host Distribution)	Recommended Requirements (targeting minimum number of hosts)
Proof of Concept / Small Demo	<p><b>Reporting Master Host</b></p> <ul style="list-style-type: none"> <li>• 4 Intel/AMD x86-64 cores</li> <li>• At least 8 GB RAM</li> <li>• At least 512 GB dedicated disk space</li> </ul> <p><b>Reporting Worker Host</b></p> <ul style="list-style-type: none"> <li>• 8 Intel/AMD x86-64 cores</li> <li>• At least 16 GB RAM</li> <li>• At least 512 GB dedicated disk space</li> </ul> <p><b>Kafka Broker Host</b></p> <ul style="list-style-type: none"> <li>• 4 Intel/AMD x86-64 cores</li> <li>• At least 6 GB RAM</li> <li>• At least 512 GB dedicated disk space</li> </ul>	Same as minimum



Environment Type	Minimum Requirements (per Host Distribution)	Recommended Requirements (targeting minimum number of hosts)
Medium	<p><b>Reporting Master Host</b></p> <ul style="list-style-type: none"> <li>• 4 Intel/AMD x86-64 cores</li> <li>• At least 8 GB RAM</li> <li>• At least 1024 GB dedicated disk space</li> </ul> <p><b>Reporting Worker Host</b></p> <ul style="list-style-type: none"> <li>• 8 Intel/AMD x86-64 cores</li> <li>• At least 16 GB RAM</li> <li>• At least 512 GB dedicated disk space</li> </ul> <p><b>Kafka Broker Host</b></p> <ul style="list-style-type: none"> <li>• 4 Intel/AMD x86-64 cores</li> <li>• At least 6 GB RAM</li> <li>• At least 1024 GB dedicated disk space</li> </ul>	<p><b>Reporting Master Host</b></p> <ul style="list-style-type: none"> <li>• 4 Intel/AMD x86-64 cores</li> <li>• At least 16 GB RAM</li> <li>• At least 1024 GB dedicated disk space</li> </ul> <p><b>Reporting Worker Host</b></p> <ul style="list-style-type: none"> <li>• 8 Intel/AMD x86-64 cores</li> <li>• At least 32 GB RAM</li> <li>• At least 512 GB dedicated disk space</li> </ul> <p><b>Kafka Broker Host</b></p> <ul style="list-style-type: none"> <li>• 4 Intel/AMD x86-64 cores</li> <li>• At least 6 GB RAM</li> <li>• At least 1024 GB dedicated disk space</li> </ul>
Medium with High Throughput or Larger	<p><b>Reporting Master Host</b></p> <ul style="list-style-type: none"> <li>• 4 Intel/AMD x86-64 cores</li> <li>• At least 16 GB RAM</li> <li>• At least 2048 GB dedicated disk space</li> </ul> <p><b>Reporting Worker Host</b></p> <ul style="list-style-type: none"> <li>• 8 Intel/AMD x86-64 cores</li> <li>• At least 32 GB RAM</li> <li>• At least 512 GB dedicated disk space</li> </ul> <p><b>Kafka Broker Host</b></p> <ul style="list-style-type: none"> <li>• 4 Intel/AMD x86-64 cores</li> <li>• At least 6 GB RAM</li> <li>• At least 2048 GB dedicated disk space</li> </ul>	More than one Reporting Worker hosts is recommended.

## Component Requirements

This topic provides the various software requirements and dependencies for the suite components (servers) for SUSE 11-based systems.

**i** On SLES systems, you must be registered for a SUSE Linux Enterprise subscription in order to have access to required rpm package dependencies.

In this topic:

- [Torque on page 14](#)
- [Moab Workload Manager on page 15](#)
- [Moab Accounting Manager on page 16](#)
- [RLM Server on page 17](#)
- [Nitro on page 17](#); Nitro Web Services is not available for a SUSE 11-based system.

## Torque

**!** If you intend to use Torque 6.1 with Moab Workload Manager, you must run Moab version 8.0 or later. However, some Torque functionality may not be available. See [Compatibility Requirements](#) in the Moab HPC Suite Release Notes for more information.

In this section:

- [Supported Operating Systems on page 14](#)
- [Software Requirements on page 14](#)

## Supported Operating Systems

- CentOS 6.x, 7.x
- RHEL 6.x, 7.x
- Scientific Linux 6.x, 7.x
- SUSE Linux Enterprise Server 11, 12, 12-SP1

## Software Requirements

- libxml2-devel package (package name may vary)
- openssl-devel package (package name may vary)
- Tcl/Tk version 8 or later if you plan to build the GUI portion of Torque, or use a Tcl-based scheduler
- cpusets and cgroups  
cgroups are supported and cpusets are handled by the cgroup cpuset subsystem.

**i** It is recommended that you use `--enable-cgroups` instead of `--enable-cpuset`. `--enable-cpuset` is deprecated and no new features will be added to it.

- boost version: 1.41 or later
- libcgroup version: Red Hat-based systems must use libcgroup version 0.40.rc1-16.el6 or later; SUSE-based systems need to use a comparative libcgroup version.
- libhwloc version: 1.9.1 is the minimum supported, however NVIDIA K80 requires libhwloc 1.11.0. Instructions for installing hwloc are provided as part of the Torque Resource Manager install or upgrade instructions.
- if you build Torque from source (i.e. clone from github), the following additional software is required:
  - gcc
  - gcc-c++
  - posix-compatible version of make
  - libtool 1.5.22 or later
  - boost-devel 1.36.0 or later

**i** Red Hat 6-based systems come packaged with 1.41.0 and Red Hat 7-based systems come packaged with 1.53.0. If needed, use the `--with-boost-path=DIR` option to change the packaged boost version. See [Customizing the Install](#) in the *Torque Resource Manager Administrator Guide* for more information.


## Moab Workload Manager

In this section:

- [Supported Operating Systems on page 15](#)
- [Software Requirements on page 16](#)
- [Supported Resource Managers on page 16](#)

### Supported Operating Systems

- CentOS 6.x, 7.x
- RHEL 6.x, 7.x
- Scientific Linux 6.x, 7.x
- SUSE Linux Enterprise Server 11, 12, 12-SP1

 A SUSE 11-based OS is *only* supported for Moab Server if your configuration does *not* include MWS.


## Software Requirements

- libcurl (<http://curl.haxx.se/libcurl/>)
- Perl 5.8.8 or later
- perl-CPAN (package name may vary)
- libxml2-devel (package name may vary)
- *(Optional)* Moab Accounting Manager 8.1
- *(Optional)* MySQL, PostgreSQL, or Oracle with ODBC driver (see [Database Configuration](#) in the *Moab Workload Manager Administrator Guide* for details)

## Supported Resource Managers

- Torque 4.2.9 or later
- Slurm

## Moab Accounting Manager

 MAM is commonly installed on the same host as Moab Workload Manager; however, in some cases you might obtain better performance by installing them on different hosts.

In this topic:

- [Supported Operating Systems on page 16](#)
- [Software Requirements on page 17](#)
- [Depends On \(not necessarily on the same host\) on page 17](#)

## Supported Operating Systems

- CentOS 6.x, 7.x
- RHEL 6.x, 7.x
- Scientific Linux 6.x, 7.x
- SUSE Linux Enterprise Server 11, 12, 12-SP1

## Software Requirements

- gcc
- perl-suidperl
- httpd
- mod\_ssl
- rrdtool
- Moab Workload Manager 9.1.1
- Perl modules; see [Installing Moab Accounting Manager on page 43 \(Manual Installation\)](#) [1.1 Installing Moab Accounting Manager \(RPM Installation\)](#) for more details

### Depends On (not necessarily on the same host)

MAM uses an RDBMS as a back end. Adaptive Computing recommends that the database used by MAM does *not* reside on the same host as the database used by Insight.

- PostgreSQL 7.2 or later


## RLM Server

Moab's Elastic Computing Feature, Viewpoint's Remote Visualization Feature, and Nitro require access to a centralized Reprise License Manager (RLM) server.


Adaptive Computing *strongly* recommends that your RLM Server is version 12.1.2.

This server is not load-extensive so it may be installed on any host within your Moab HPC Suite environment. It may also be installed on its own host.

 If your company already utilizes an RLM Server, you do not have to install another as long as the Moab HPC Suite components can access it.

 The host on which you install RLM Server must always be on and should have High Availability (uptime).

## Nitro

 When integrated with the Moab HPC Suite, Nitro resides on the Torque compute nodes.

In this section:

- [Hardware Requirements on page 18](#)
- [Supported Operating Systems on page 18](#)
- [License Requirements on page 18](#)
- [Software Requirements on page 18](#)

## Hardware Requirements

- Nitro requires one or more multi-core processors per host. Generally the more processors (sockets) and/or OS cores a host has, the more tasks Nitro can execute simultaneously on each host; although this will be application-dependent.
- It is recommended that hosts should have sufficient memory to execute as many applications as possible so that Nitro can run them at a rate of one application instance per OS core (especially if they are not multi-threaded). This eliminates the need for users to have to request memory in their Nitro task definitions.

 See the *Nitro Administrator Guide* for information on specifying memory requirements.

## Supported Operating Systems

- CentOS 6.x, 7.x
- Red Hat 6.x, 7.x
- Scientific Linux 6.x, 7.x
- SUSE Linux Enterprise Server 11, 12, 12-SP1

## License Requirements

Nitro requires access to a centralized Reprise License Manager (RLM) server. See [RLM Server on page 17](#) for more information.

## Software Requirements

Nitro is built with all needed libraries statically linked. This provides for a quick and simple installation and helps avoid troublesome library mismatches. No additional packages need to be installed on the compute nodes.

However, users running nitrosub and/or the nitrostat utility require Python 2.6.6 or later on the system from which they are running it.

# RLM Licensing Model

The Moab HPC Suite uses Reprise License Manager (RLM) to enforce nodelocked and floating licenses.

## Definitions

Term	Definition
<b>Nodelocked License</b>	A nodelocked license is a license grant which allows software to be used on a particular computer, and on that computer only. A license server is not required to enforce a nodelocked license. However, a unique unmodifiable ID unique to the computer is required.
<b>Floating License</b>	Floating licensing is a software licensing approach in which a limited number of licenses for a software application are shared among a larger number of users/clients over time. When an authorized user wishes to run the application they request a license from a central license server. A license server is used for floating licenses and logging of usage data.
<b>RLM HostID</b>	<p>The host ID of the RLM server. Adaptive Computing requests that you provide the 'ether' RLM HostID when the RLM HostID is required.</p> <p>RLM's <code>rlmhostid</code> command can be used to obtain the 'ether' RLM HostID. However, the <code>rlmhostid</code> command may not be available on every host (typically only available where RLM server is installed). The <code>rlmhostid</code> command takes an optional parameter <code>hostid type</code>, which defaults to <code>ether</code> when not given.</p> <p>When the <code>rlmhostid</code> command is unavailable, provide the Ethernet MAC address of one of the host's Network Interface Cards (NICs).</p>

## License File Names and Installation Locations

You will be provided license files with file names depicted below. Please install each file in the designated Destination Directory and rename the files to the Installed File Name. When issued RLM-signed licenses, technically, the only Installed File Name requirement is that the file name end in `.lic`. However, we suggest you use the Installed File Name.

**i** Do not install more than one of each of the following licenses in the respective Destination Directory. Having more than one `moab-rlm--xxxxx.lic` file in `/opt/moab/etc`, for example, will cause problems. Thus, the suggested Installed File Names.

File Description	File Name	Destination Directory	Installed File Name	Host
Moab Workload Manager (version >= 9.1.0)	<code>moab-rlm--xxxxx.lic</code>	<code>/opt/moab/etc</code>	<code>moab-rlm.lic</code>	Moab Server Host
Moab Workload Manager (version < 9.1.0)	<code>moab_xxxx.lic</code> <code>moabl_xxxx.lic</code> <code>moabc_xxxx.lic</code>	<code>/opt/moab/etc</code>	<code>moab.lic</code>	Moab Server Host
Elastic Computing (Moab side)	<code>moab-rlm-elastic--xxxxx.lic</code>	<code>/opt/moab/etc</code>	<code>moab-rlm-elastic.lic</code>	Moab Server Host
Elastic Computing (RLM side)	<code>moab-rlm-elastic-tracking--xxxxx.lic</code>	<code>/opt/rlm</code>	<code>moab-rlm-elastic-tracking.lic</code>	RLM Host
Viewpoint	<code>moab-vp--xxxxx.lic</code>	<code>/opt/viewpoint/etc/license.d</code>	<code>license.lic</code>	Viewpoint Host
Nitro	<code>nitro-rlm--xxxxx.lic</code>	<code>/opt/rlm/</code>	<code>nitro.lic</code>	RLM Host
Nitro	<code>nitro-rlm--xxxxx.lic</code>	<code>/opt/nitro/etc</code>	<code>nitro.lic</code>	Nitro Coordinator Host - All Nitro Compute Nodes
Adaptive Computing ISV RLM	<code>adapt-iveco.set</code> <code>adapt-iveco.opt</code>	<code>/opt/rlm</code>	<code>adapt-iveco.set</code> <code>adapt-iveco.opt</code>	RLM Host
StartNet ISV RLM	<code>starnet.set</code>	<code>/opt/rlm</code>	<code>starnet.set</code>	RLM Host



## Licenses Issued by Adaptive Computing

Instructions for obtaining licenses from Adaptive Computing are given below.

### Nodelocked/Uncounted Licenses

#### *Moab License*

This license is limited by the number of sockets the scheduler will schedule. The socket count includes physical sockets, GPUs, and Xeon Phis in your cluster.

Moab does not delegate license enforcement by socket to an external licensing server. Doing so allows Moab to give you visibility into all resources reported by your resource managers, while scheduling within the limits of your licensing agreement.

Please send the following to `licenses@adaptivecomputing.com` to obtain a Moab License:

- RLM Hostid of the Moab Server host
- Total number of physical sockets, GPUs, and Xeon Phis in your cluster

Install this license as `/opt/moab/etc/moab-rlm.lic` on the Moab Server host.

#### *Viewpoint License*

Please send the following to `licenses@adaptivecomputing.com` to obtain a Viewpoint License:

- RLM Hostid of the Viewpoint Server host

Install this license as `/opt/viewpoint/etc/license.d/license.lic` on the Viewpoint Server host.

### Floating/Counted Licenses

#### *Remote Visualization License*

Please send the following to `licenses@adaptivecomputing.com` to obtain a Remote License:

- RLM Server Hostname
- RLM Hostid of the RLM Server host
- Max number of concurrent Remote Viz sessions

Install this license as `/opt/rlm/fastx.lic` on the RLM Server host.

#### *Elastic Computing License*

Note that the Elastic Computing feature is also part of the Moab and Viewpoint nodelocked licenses. Adaptive Computing will issue you new Moab/Viewpoint nodelocked licenses when needed.

Please send the following to [licenses@adaptivecomputing.com](mailto:licenses@adaptivecomputing.com) to obtain all the necessary licenses to enable Elastic Computing on your cluster:

- RLM Server Hostname
- RLM Hostid of the RLM server

Required to generate/regenerate the Moab License:

- RLM Hostid of the Moab Server host
- Total number of physical sockets, GPUs, and Xeon Phis in your cluster

If using Viewpoint, required to generate/regenerate the Viewpoint License:

- RLM Hostid of the Viewpoint Server host

Three Moab licenses will be generated:

1. Moab license - A nodelocked license with the Elastic Computing feature enabled
2. Moab Elastic license - A file that tells Moab where the RLM server is located. RLM will be used to track Elastic Computing usage statistics.
3. Moab Elastic Tracking license - A file that tells RLM to allow unlimited Elastic Computing usage and to track when Elastic nodes are added and removed from the cluster.

On the Moab Server Host:

- Install the Moab nodelocked license as `/opt/moab/etc/moab-rlm.lic`.
- Install the Moab Elastic license as `/opt/moab/etc/moab-elastic.lic`.

On the RLM Server Host:

- Install the Elastic Tracking licenses as `/opt/moab/etc/moab-elastic-tracking.lic`.

If using Viewpoint, on the Viewpoint Server Host:

- Install the Viewpoint nodelocked license as `/opt/viewpoint/etc/license.d/license.lic`.

### *Nitro License*

Nitro can be licensed based on number of concurrent Nitro Workers or number of cores.

Please send the following to [licenses@adaptivecomputing.com](mailto:licenses@adaptivecomputing.com) to obtain a Nitro License:

- RLM Server Hostname
- RLM Hostid of the RLM server
- Number of Nitro Workers or cores.

## Identify the Installation Methods

Adaptive Computing provides different methods for installing the Moab components, Manual Installation, RPM Installation or the new Automated Installation (uses RPM methodology).

Depending on your environment and which components you are installing (and on which host), you may need to use a combination of Manual Installation and RPM Installation. However, the automated installer is only available for systems that support RPM installations. See for more information on the automated installer.

### *Manual Installation*

This method provides advantages for administrators who want non-standard configure options.

- This method has more supported operating systems than the RPM Installation method.
- Some components can not be installed using the Manual Installation method.

### *RPM Installation*

This method provides advantages for administrator who want a standard installation, with little customization.

- This method is *not* applicable for SUSE 11-based systems.

### *Automated Installation*

This method provides advantages for systems who do not want the complexity of the Manual Installation or a RPM Typical or Offline Installation.

- This method is *not* applicable for SUSE 11-based systems.



## Chapter 2 Manual Installation

This chapter provides installation, configuration, and upgrading information using the Manual Installation method.

Be aware of the following:

- On SLES systems, you must be registered for a SUSE Linux Enterprise subscription in order to have access to required rpm package dependencies.
- Manual Installation is not available for Insight, Viewpoint, or Remote Visualization.
- Because many system-level files and directories are accessed during the installation, the instructions in this guide should be executed with root privileges. You will see that the instructions execute commands as the root user. Also be aware that the same commands will work for a non-root user with the `sudo` command.

Related Topics

- [Chapter 1 Planning your Installation on page 3](#)
- [Preparing for Manual Installation on page 26](#)

# Manual Installation

This section provides instructions and other information for installing your Moab components for SUSE 11-based systems using the Manual installation method.

In this section:

- [Preparing for Manual Installation on page 26](#)
- [Installing Torque Resource Manager on page 28](#)
- [Installing Moab Workload Manager on page 35](#)
- [Installing Moab Accounting Manager on page 43](#)
- [Installing RLM Server on page 53](#)
- [Nitro Integration on page 56](#)

## Preparing for Manual Installation

The manual installation process of the Moab includes installing the different components in the suite.

**i** Many individual components have dependencies on other components (see [Chapter 1 Planning your Installation on page 3](#)). However, if you do not require a certain component, you do not have to install it.

The install instructions for each component include information about system requirements and dependencies. Some include prerequisite instructions that you will need to complete before you begin the install. Please read this information carefully, and make sure you have installed all the dependencies and packages that are necessary in order to avoid errors during the Moab install process.

**i** Because many system-level files and directories are accessed during the installation, the instructions in this guide should be executed with root privileges.

You will see that the instructions execute commands as the root user. Please note that the same commands will work for a non-root user with the `sudo` command.

This topic contains prerequisite instructions that you will need to complete before you begin the installations.

In this topic:

- [Set Up Proxies on page 27](#)
- [Enable Extra Packages for the Repository on page 27](#)
- [Update Your System Software to the Latest Version on page 27](#)
- [Ensure Hostname Resolution for all Hosts on page 28](#)
- [Install the Moab HPC Suite Software Components on page 28](#)

## Set Up Proxies

If your site uses a proxy to connect to the internet, configure yum to use a proxy by editing the `/etc/yum.conf` file as follows:

```
proxy=http://<proxy_server_id>:<port>
```

If your site uses an external repository to install python dependencies (for example, the host where you install Viewpoint might need to download extra packages), you will need to set up pip to use a proxy. Do the following:

```
export http_proxy=http://<proxy_server_id>:<port>
export https_proxy=http://<proxy_server_id>:<port>
```

## Enable Extra Packages for the Repository

Many individual components have dependencies that are found in the optional add-on repositories for the distribution. You must enable the respective repository for your distribution on all hosts upon which you install Adaptive Computing software components.

Do the following:

1. Verify that you have a licensed installation of SLES 11 and that you are registered for a SUSE Linux Enterprise subscription.
2. Download the SUSE Linux Enterprise 11 Software Development Kit e-Media Kit and add the ISO to the repository.
3. Add the `devel:languages:perl` and the `devel:languages:python` repositories.

```
[root]# zypper addrepo --refresh --repo
http://download.opensuse.org/repositories/devel:/languages:/perl/SLE_11_
SP4/devel:languages:perl.repo
[root]# zypper addrepo --refresh --repo
http://download.opensuse.org/repositories/devel:/languages:/python/SLE_11_
SP4/devel:languages:python.repo
```

## Update Your System Software to the Latest Version

It is recommended that you update your system software to the latest version before installing Moab HPC Suite components.

On *each* host where you will install the Moab HPC Suite components, do the following:

```
[root]# zypper update
```

## Ensure Hostname Resolution for all Hosts

Each host should be resolvable from all other hosts in the cluster. Usually this is implemented by having all hosts in DNS. Alternatively, each host may include all other hosts (with the correct IP address) in its `/etc/hosts` file.

## Install the Moab HPC Suite Software Components

To install the Moab, install the packages in the following order:

1. Torque. See [Installing Torque Resource Manager on page 28](#).
2. Moab Workload Manager. See [Installing Moab Workload Manager on page 35](#).
3. Moab Accounting Manager. See [Installing Moab Accounting Manager on page 43](#).
4. RLM Server. See [Installing RLM Server on page 53](#).
5. Integrate Nitro with your Moab HPC Suite. See [Nitro Integration on page 56](#).

## Installing Torque Resource Manager



If you intend to use Torque Resource Manager 6.1.1.1 with Moab Workload Manager, you must run Moab version 8.0 or later. However, some Torque functionality may not be available. See [Compatibility Requirements](#) in the Moab HPC Suite Release Notes for more information.

This topic contains instructions on how to install and start Torque Resource Manager (Torque).



**i** For Cray systems, Adaptive Computing recommends that you install Moab and Torque Servers (head nodes) on commodity hardware (*not* on Cray compute/service/login nodes).

*However, you must install the Torque pbs\_mom daemon and Torque client commands on Cray login and "mom" service nodes since the pbs\_mom must run on a Cray service node within the Cray system so it has access to the Cray ALPS subsystem.*

See [Installation Notes for Moab and Torque for Cray](#) in the *Moab Workload Manager Administrator Guide* for instructions on installing Moab and Torque on a non-Cray server.

In this topic:

- [Open Necessary Ports on page 29](#)
- [Install Dependencies, Packages, or Clients on page 30](#)
- [Install Torque Server on page 31](#)
- [Install Torque MOMs on page 33](#)
- [Install Torque Clients on page 34](#)
- [Configure Data Management on page 35](#)

## Open Necessary Ports

Torque requires certain ports to be open for essential communication.

If your site is running firewall software on its hosts, you will need to configure the firewall to allow connections to the necessary ports.

Location	Ports	Functions	When Needed
Torque Server Host	15001	Torque Client and MOM communication to Torque Server	Always
Torque MOM Host (Compute Nodes)	15002	Torque Server communication to Torque MOMs	Always
Torque MOM Host (Compute Nodes)	15003	Torque MOM communication to other Torque MOMs	Always

See also:

- [Opening Ports in a Firewall on page 63](#) for general instructions and an example of how to open ports in the firewall.
- [Configuring Ports](#) in the *Torque Resource Manager Administrator Guide* for more information on how to configure the ports that Torque uses for communication.

## Install Dependencies, Packages, or Clients

In this section:

- [Install Packages on page 30](#)
- [Install hwloc on page 30](#)

### Install Packages

On the Torque Server Host, use the following commands to install the `libxml2-devel`, `openssl-devel`, and `boost-devel` packages.

```
[root]# zypper install libopenssl-devel libtool libxml2-devel boost-devel gcc gcc-c++
make gmake postfix
```

### Install hwloc



Using "zypper install hwloc" may install an older, non-supported version.

When cgroups are enabled (recommended), hwloc version 1.9.1 or later is required. NVIDIA K80 requires libhwloc 1.11.0. If cgroups are to be enabled, check the Torque Server Host to see if the required version of hwloc is installed. You can check the version number by running the following command:

```
[root]# hwloc-info --version
```

The following instructions are for installing version 1.9.1.

If hwloc is not installed or needs to be upgraded to the required version, do the following:

1. On the Torque Server Host, *each* Torque MOM Host, and *each* Torque Client Host, do the following:
  - a. Download `hwloc-1.9.1.tar.gz` from <https://www.openmpi.org/software/hwloc/v1.9>.

- b. Run each of the following commands in order.

```
[root]# zypper install gcc make
[root]# tar -xzvf hwloc-1.9.1.tar.gz
[root]# cd hwloc-1.9.1
[root]# ./configure
[root]# make
[root]# make install
```

2. Run the following commands on the Torque Server Host, *each* Torque MOM Host, and *each* Torque Client Host.

```
[root]# echo /usr/local/lib >/etc/ld.so.conf.d/hwloc.conf
[root]# ldconfig
```

## Install Torque Server

**i** You **must** complete the tasks to install the dependencies, packages, or clients before installing Torque Server. See [Install Dependencies, Packages, or Clients on page 30](#).

If your configuration uses firewalls, you *must also* open the necessary ports before installing the Torque Server. See [Open Necessary Ports on page 29](#).

On the Torque Server Host, do the following:

1. Download the latest 6.1.1.1 build from the [Adaptive Computing](#) website. It can also be downloaded via command line (github method or the tarball distribution).
  - Clone the source from github.

**i** If git is not installed:

```
[root]# zypper install git
```

You will *also* need to install automake:

```
[root]# zypper install autotools automake pkg-config
```

```
[root]# git clone https://github.com/adaptivecomputing/torque.git -b 6.1.1.1
6.1.1.1
[root]# cd 6.1.1.1
[root]# ./autogen.sh
```

- Get the tarball source distribution.

```
[root]# zypper install wget
[root]# wget http://www.adaptivecomputing.com/download/torque/torque-
6.1.1.1.tar.gz -O torque-6.1.1.1.tar.gz
[root]# tar -xzvf torque-6.1.1.1.tar.gz
[root]# cd torque-6.1.1.1/
```

2. Depending on your system configuration, you will need to add `./configure` command options.

At a minimum, you add:

- `--enable-cgroups`
- `--with-hwloc-path=/usr/local` See [Torque on page 14](#) for more information.

**i** These instructions assume you are using `cgroups`. When `cgroups` are supported, `cpusets` are handled by the `cgroup cpuset` subsystem. If you are not using `cgroups`, use `--enable-cpusets` instead.

See [Customizing the Install](#) in the *Torque Resource Manager Administrator Guide* for more information on which options are available to customize the `./configure` command.

3. Run each of the following commands in order.

```
[root]# ./configure --enable-cgroups --with-hwloc-path=/usr/local # add any other
specified options
[root]# make
[root]# make install
```

4. Source the appropriate profile file to add `/usr/local/bin` and `/usr/local/sbin` to your path.

```
[root]# . /etc/profile.d/torque.sh
```

5. Initialize `serverdb` by executing the `torque.setup` script.

```
[root]# ./torque.setup root
```

6. Add nodes to the `/var/spool/torque/server_priv/nodes` file. See [Specifying Compute Nodes](#) in the *Torque Resource Manager Administrator Guide* for information on syntax and options for specifying compute nodes.

7. Configure `pbs_server` to start automatically at system boot, and then start the daemon.

```
[root]# chkconfig --add pbs_server
[root]# service pbs_server restart
```

## Install Torque MOMs

In most installations, you will install a Torque MOM on each of your compute nodes.

**i** See [Specifying Compute Nodes](#) or [Configuring Torque on Compute Nodes](#) in the Torque Resource Manager Administrator Guide for more information.

Do the following:

1. On the Torque Server Host, do the following:
  - a. Create the self-extracting packages that are copied and executed on your nodes.

```
[root]# make packages
Building ./torque-package-clients-linux-x86_64.sh ...
Building ./torque-package-mom-linux-x86_64.sh ...
Building ./torque-package-server-linux-x86_64.sh ...
Building ./torque-package-gui-linux-x86_64.sh ...
Building ./torque-package-devel-linux-x86_64.sh ...
Done.
```

The package files are self-extracting packages that can be copied and executed on your production machines. Use `--help` for options.

- b. Copy the self-extracting MOM packages to each Torque MOM Host.

Adaptive Computing recommends that you use a remote shell, such as SSH, to install packages on remote systems. Set up shared SSH keys if you do not want to supply a password for each Torque MOM Host.

```
[root]# scp torque-package-mom-linux-x86_64.sh <mom-node>:
```

- c. Copy the `pbs_mom` startup script to each Torque MOM Host.

```
[root]# scp contrib/init.d/suse.pbs_mom <mom-node>:/etc/init.d/pbs_mom
```

**i** Not all sites see an inherited `ulimit` but those that do can change the `ulimit` in the `pbs_mom` init script. The `pbs_mom` init script is responsible for starting and stopping the `pbs_mom` process.

2. On *each* Torque MOM Host, confirm that `cgroups` have been mounted; if not, mount them.
  - a. Run `lssubsys -am`.
  - b. If the command is not found, or you do not see something similar to the following, then `cgroups` are *not* mounted, continue with these instructions.

```
ns
perf_event
net_prio
cpuset /cgroup/cpuset
cpu /cgroup/cpu
cpuacct /cgroup/cpuacct
memory /cgroup/memory
devices /cgroup/devices
freezer /cgroup/freezer
net_cls /cgroup/net_cls
blkio /cgroup/blkio
```

c. Do the following:

i. Install the cgroup library package.

```
[root]# zypper install libcgroup1
```

ii. Edit `/etc/cgconfig.conf` and add the following:

```
mount {
    devices = /mnt/cgroups/devices;
    cpuset = /mnt/cgroups/cpuset;
    cpu = /mnt/cgroups/cpu;
    cpuacct = /mnt/cgroups/cpuacct;
    memory = /mnt/cgroups/memory;
}
```

iii. Mount cgroups.

```
[root]# service cgconfig start
[root]# chkconfig cgconfig on
```

d. Run `lsusb -am` again and confirm cgroups are mounted.

3. On *each* Torque MOM Host, do the following:

a. Install cgroup-tools.

b. Install the self-extracting MOM package.

```
[root]# ./torque-package-mom-linux-x86_64.sh --install
```

c. Configure `pbs_mom` to start at system boot, and then start the daemon.

```
[root]# chkconfig --add pbs_mom
[root]# service pbs_mom start
```

## Install Torque Clients

If you want to have the Torque client commands installed on hosts other than the Torque Server Host (such as the compute nodes or separate login nodes), do the following:

1. On the Torque Server Host, do the following:
  - a. Copy the self-extracting client package to *each* Torque Client Host.

**i** Adaptive Computing recommends that you use a remote shell, such as SSH, to install packages on remote systems. Set up shared SSH keys if you do not want to supply a password for each Torque Client Host.

```
[root]# scp torque-package-clients-linux-x86_64.sh <torque-client-host>:
```

- b. Copy the trqauthd startup script to *each* Torque Client Host.

```
[root]# scp contrib/init.d/suse.trqauthd <torque-client-host>:/etc/init.d/trqauthd
```

2. On *each* Torque Client Host, install the self-extracting client package:

```
[root]# ./torque-package-clients-linux-x86_64.sh --install
```

## Configure Data Management

When a batch job completes, stdout and stderr files are generated and placed in the spool directory on the master Torque MOM Host for the job instead of the submit host. You can configure the Torque batch environment to copy the stdout and stderr files back to the submit host. See [Configuring Data Management](#) in the *Torque Resource Manager Administrator Guide* for more information.

Related Topics

- [Preparing for Manual Installation on page 26](#)

## Installing Moab Workload Manager

This topic contains instructions on how to install and start Moab Workload Manager (Moab).

**i** For Cray systems, Adaptive Computing recommends that you install Moab and Torque Servers (head nodes) on commodity hardware (*not* on Cray compute/service/login nodes).

*However, you must install the Torque pbs\_mom daemon and Torque client commands on Cray login and "mom" service nodes since the pbs\_mom must run on a Cray service node within the Cray system so it has access to the Cray ALPS subsystem.*

See [Installation Notes for Moab and Torque for Cray](#) in the *Moab Workload Manager Administrator Guide* for instructions on installing Moab and Torque on a non-Cray server.

In this topic:

- [Understand Licenses on page 36](#)
- [Open Necessary Ports on page 37](#)
- [Install Dependencies, Packages, or Clients on page 37](#)
- [Obtain and Install the Elastic Computing License on page 37](#)
- [Install Moab Server on page 39](#)
- [Configure Torque to Trust Moab on page 42](#)
- [Verify the Installation on page 42](#)
- [\(Optional\) Install Moab Client on page 42](#)

## Understand Licenses

As part of the Moab modularity, introduced in version 9.0.1, Moab features can be licensed separately. See [Module-Based Features](#) in the Moab Workload Manager Administrator Guide.

With the 9.1.0 release, Moab now uses an RLM Server to manage licenses. For the Moab core and for most Moab features, an RLM Server is not required. The new Moab "core" license will have a new name to reflect the RLM generation. Do *not* rename this license to moab.lic. See [RLM Licensing Model on page 19](#) for more information about RLM licensing.

Elastic Computing, beginning with 9.1.0, requires an RLM Server as part of your configuration.

**i** The 9.1.0 licensing change does not affect legacy licenses; however, a module-based licensed may be required to use newer functionality.



## Open Necessary Ports

If your site is running firewall software on its hosts, you will need to configure the firewall to allow connections to the necessary ports.

Location	Ports	Functions	When Needed
Moab Server Host	42559	Moab Server Port	If you intend to run client commands on a host different from the Moab Server Host <i>or</i> if you will be using Moab in a grid

See [Opening Ports in a Firewall on page 63](#) for general instructions and an example of how to open ports in the firewall.

## Install Dependencies, Packages, or Clients

In this section:

- [Dependencies and Packages on page 37](#)
- [Torque Client on page 37](#)

### Dependencies and Packages

On the Moab Server Host, use the following commands to install the required Moab dependencies and packages.

```
[root]# zypper install make curl libxml2-devel gcc
```

### Torque Client

If you are using Torque and are installing the Torque Server on a different host (Torque Server Host) from the Moab Server (Moab Server Host), you will need to install the Torque client on the Moab Server Host in order for Moab to interact with Torque.

Follow the instructions in [Install Torque Clients on page 34](#) using the Moab Server Host as the Torque Client Host; with the exception that you must copy and install the torque-package-devel-linux-**<arch>**.sh self-extracting package in addition to the torque-package-clients-linux-**<arch>**.sh package.

## Obtain and Install the Elastic Computing License

If using Elastic Computing, Moab requires access to an RLM license server to record usage.



These instructions assume you already have access to an RLM Server. See [Installing RLM Server on page 53](#) for instructions on how to set up a new RLM Server.

Do the following:

1. On the RLM server, obtain the hostid and hostname.

- hostid

```
[root]# /opt/rlm/rlmhostid
```

You should see output similar to the following.

```
rlmhostid v12.1
Copyright (C) 2006-2016, Reprise Software, Inc. All rights reserved.

Hostid of this machine: 00259096f004
```

- hostname

```
[root]# /opt/rlm/rlmhostid host
```

You should see output similar to the following.

```
rlmhostid v12.1
Copyright (C) 2006-2016, Reprise Software, Inc. All rights reserved.

Hostid of this machine: host=<your-host-name>
```

2. Email licenses@adaptivecomputing.com for a license and include the hostid and hostname you just obtained.
3. Adaptive Computing will generate the license and send you the Elastic Computing license file (.lic) file in a return email.
4. On the RLM server, do the following:
  - a. Download and install the license file.

```
[root]# cd /opt/rlm
[root]# chown rlm:rlm <licenseFileName>.lic
```

- b. If the RLM Server in your configuration uses a firewall, edit the license file to reference the ISV adaptiveco port for the Adaptive license-enabled products. This is the same port number you opened during the RLM Server installation. See the instructions to open necessary ports in the [Installing RLM Server on page 53](#) (manual installation method) or [1.1 Installing RLM Server](#) (RPM installation method) for more information.

```
[root]# vi /opt/rlm/moab_elastic_tracking.lic
```

```
ISV adaptiveco port=5135
```

The license file already references the RLM Server port (5053 by default).

**i** If the RLM Server in your configuration uses different ports, you will need to modify the license file to reflect the actual ports. See the instructions to open necessary ports in the [Installing RLM Server on page 53](#) (manual installation method) or [1.1 Installing RLM Server](#) (RPM installation method) for more information.

- c. If you did *not* install an RLM Server using the file available from Adaptive Computing (for example, because your system configuration already uses one), do the following:
  - i. Download the 'adaptiveco.set' file from [the Adaptive Computing Moab HPC Suite Download page](#) (<http://www.adaptivecomputing.com/support/download-center/moab-hpc-suite-download/>).
  - ii. Install the 'adaptiveco.set' file.

```
[root]# chown rlm:rlm adaptiveco.set
```

- iii. Place the 'adaptiveco.set' file in the *same* directory where the Elastic Computing license resides. Typically, this is the RLM Server base directory (/opt/rlm); but may be different depending on your configuration
- d. Perform a reread on the RLM Server base directory to update the RLM Server with your license. For example:

```
[root]# /opt/rlm/rlmreread
```

## Install Moab Server

**i** You **must** complete the tasks to install the dependencies, packages, or clients before installing Moab Server. See [Install Dependencies, Packages, or Clients on page 37](#).

If your configuration uses firewalls, you *must also* open the necessary ports before installing the Moab Server. See [Open Necessary Ports on page 37](#).

On the Moab Server Host, do the following:

1. Download the latest Moab build from the [Adaptive Computing Moab HPC Suite Download Center](#) (<https://www.adaptivecomputing.com/support/download-center/moab-hpc-suite-download/>).

- As the root user, run each of the following commands in order.

```
[root]# tar xzvf moab-9.1.1-<OS>.tar.gz
[root]# cd moab-9.1.1-<OS>
```

**i** The variable marked `<os>` indicates the OS for which the build was designed.

If Elastic Computing is part of your Moab Workload Manager configuration, install **deps/acpython-base\***.

```
[root]# zypper install deps/acpython-base*
```

- Configure Moab. If you are installing Moab Accounting Manager, configure Moab with the `--with-am` option.

```
[root]# ./configure <options>
```

**i** See [Moab Workload Manager Configuration Options on page 63](#) for a list of commonly used options or use `./configure --help` for a complete list of available options.

- ONLY* if you are using green computing, or if you are using a resource manager other than Torque.

Run the `make perldeps` command to install the necessary perl modules using CPAN. When first running CPAN, you will be asked for configuration information. It is recommended that you choose an automatic configuration. You will be prompted to provide input during module installation; running the `make perldeps` command with a script is not recommended.

```
[root]# make perldeps
```

- Install Moab.

```
[root]# make install
```

- Modify the Moab configuration file.

```
[root]# vi /opt/moab/etc/moab.cfg
```

Do one of the following:

- If using Torque Resource Manager:
  - Verify that **SUBMITCMD** is set up for your Torque resource manager and that it points to a valid `qsub` executable. For example:

```
RMCFG[torque] SUBMITCMD=/usr/local/bin/qsub
```

- If you installed the Torque Server on a different host (Torque Server Host), configure the RMCFG HOST parameter to tell Moab the host on which Torque Server is running.

```
RMCFG[torque] HOST=<torque_server_hostname>
```

- b. If using Moab to schedule Slurm Resource Manager, see [Moab-SLURM Integration Guide](#) in the *Moab Workload Manager Administrator Guide* for configuration information. If using the Moab Passthrough feature for Viewpoint on Slurm, see [Moab Passthrough](#) for additional information and installation instructions.
  - c. If using a NATIVE resource manager, see [Managing Resources Directly with the Native Interface](#) in the *Moab Workload Manager Administrator Guide* for configuration information.
7. Source the appropriate profile script to add the Moab executable directories to your current shell `$PATH` environment.

```
[root]# . /etc/profile.d/moab.sh
```

8. Copy your license file into the same directory as `moab.cfg` (`/opt/moab/etc/` by default).

```
[root]# cp moab.lic $MOABHOMEDIR/etc/moab.lic
```

To verify the current status of your license, run the following command:

```
[root]# moab --about 2>&1 | grep License
```

You should get something similar to the following in the response:

- New RLM-Based License (version 9.1.0 or after)

```
$ moab --about | grep License
Moab Workload Manager Version 'master' License Information:
Current License: (moab_license) Valid Until - 15-jan-2018
Current License: Max Sockets = 1000000
Current License: (moab_grid) Valid Until - 15-jan-2018
Current License: (moab_green) Valid Until - 15-jan-2018
Current License: (moab_provision) Valid Until - 15-jan-2018
Current License: (moab_vms) Valid Until - 15-jan-2018
Current License: Max VMs = 1000000
Current License: (moab_elastic) Valid Until - 15-jan-2018
Current License: (moab_groupsharing) Valid Until - 15-jan-2018
Current License: (moab_advancedrm) Valid Until - 15-jan-2018
Current License: (moab_workflow) Valid Until - 15-jan-2018
Current License: (moab_accounting) Valid Until - 15-jan-2018
```

- Legacy License Format

```
Moab Workload Manager Version '9.1.1' License Information:
Current License: Max Procs = 10000
Current License: Valid Until - Jul 13 19:42:10 2018
```

**i** A license is required for Moab. A trial license may be included in your Moab installation enabling you to run Moab for a limited time and with limited features. Email [licenses@adaptivecomputing.com](mailto:licenses@adaptivecomputing.com) for information on obtaining licenses.

## 9. Start Moab.

```
[root]# chkconfig moab on
[root]# service moab start
```

## Configure Torque to Trust Moab

If you are using Torque as a resource manager and you installed the Torque Server on a different host (Torque Server Host); recommended, do the following:

- On the Torque Server Host, add the name of the Moab Server Host (where Moab Server is installed) as a manager and as a submit host.

```
[root]# qmgr
Qmgr: set server managers += root@<moab_server_hostname>
Qmgr: set server submit_hosts += <moab_server_hostname>
Qmgr: exit
```

## Verify the Installation

If you have a resource manager configured, verify that the scheduler is able to schedule a job. Do the following:

- Submit a sleep job as a non-root user (adaptive is used in this example) and verify the job is running.

```
[root]# su - adaptive
[adaptive]$ echo sleep 150 | msub
[adaptive]$ showq
[adaptive]$ exit
```

## (Optional) Install Moab Client

After you have installed Moab Server, you can create a client tarball to install just the Moab client commands on a login/client host. This tarball uses a single `tar` command to install the binary Moab client command files and their man pages. The tarball also contains a `moab.cfg` file configured with the Moab Server host name and port number so you do not have to manually configure this information on the login/client node.

**i** If your site needs secure communication and authentication between Moab Client Host and the Moab Server Host, create a site-specific key and place it in the same directory as your `moab.cfg` file. By default, this would be `$MOABHOMEDIR/etc/.moab.key`. When the Moab server and client commands detect the presence of those two files they will use the key in those files to authenticate and communicate, instead of the default key. See [Mauth Authentication](#) in the *Moab Workload Manager Administrator Guide* for more information.

Do the following:

1. On the Moab Server Host, create the client tarball.

```
[root]# make client-pkg
```

2. Copy the tarball to the root directory of the Moab Client Host.
3. On the Moab Client Host, run the tarball to install the Moab client commands.

```
[root]# tar xvf client.tgz
```

Related Topics

- [Preparing for Manual Installation on page 26](#)

## Installing Moab Accounting Manager

This topic contains instructions on how to install and start Moab Accounting Manager (MAM).

Perform the following in order:

1. [Plan Your Installation](#)
2. [Open Necessary Ports](#)
3. [Install and Initialize PostgreSQL Server](#)
4. [Install Dependencies, Packages, or Clients](#)
5. [Install MAM Server](#)
6. [Configure the MAM GUI](#)
7. [Configure MAM Web Services](#)
8. [Access the MAM GUI](#)
9. [Access MAM Web Services](#)
10. [Configure Moab Workload Manager to Use Moab Accounting Manager](#)
11. [Initialize Moab Accounting Manager](#)

## Plan Your Installation

The first step is determining the number of different hosts (physical machines) required for your MAM installation.

Your MAM installation includes:

- MAM Server
- MAM Database
- MAM Clients (possibly several hosts)
- MAM GUI (optional)
- MAM Web Services (optional)

Each of these components can be installed on their own hosts (meaning the actual physical machine) or can be combined on same hosts. For example, the MAM Database can be installed on the same *host* as the MAM Server. Or the MAM Server may be installed on the same host on which you installed the Moab Server.

Once you have determined which components are installed on which hosts, complete the rest of the instructions for the MAM installation.

**i** The instructions that follow in this topic will use the term *host* after each component to indicate the physical machine on which the component is installed (for example, MAM Server Host and MAM Database Host). Depending on your configuration, the *host* may refer to the component installed on its own machine or installed on the same machine as another component.

## Open Necessary Ports

If your site is running firewall software on its hosts, you will need to configure the firewall to allow connections to the necessary ports.

Location	Ports	Functions	When Needed
MAM Server Host	7112	MAM Server Port	If you will be installing the MAM Server on a different host from where you installed the Moab Server <i>or</i> you will be installing the MAM Clients on other hosts
MAM GUI Host	443	HTTPS Port	If using the MAM GUI



Location	Ports	Functions	When Needed
MAM Web Services Host	443	HTTPS Port	If using MAM Web Services
MAM Database Host	5432	MAM PostgreSQL Server Port	If you will be installing the MAM Database on a different host from the MAM Server

See [Opening Ports in a Firewall on page 63](#) for general instructions and an example of how to open ports in the firewall.

## Install and Initialize PostgreSQL Server

Moab Accounting Manager uses a database for transactions and data persistence. The PostgreSQL database may be installed on a different host from the MAM Server; however, it is often convenient to install them on the same host. For example, the PostgreSQL database may be installed on:

- the same host as the MAM Server.
- a separate PostgreSQL database host.
- a separate *shared* PostgreSQL database host.

On the host where the MAM PostgreSQL database will reside, do the following:

1. Install and initialize the PostgreSQL Server.

```
[root]# zypper install postgresql-server
[root]# service postgresql start
```

2. Configure trusted connections.

Edit or add a "host" line in the `pg_hba.conf` file for the interface from which the MAM Server will be connecting to the database and ensure that it specifies a secure password-based authentication method (for example, `md5`).

```
[root]# vi /var/lib/pgsql/data/pg_hba.conf

# Replace 127.0.0.1 with the IP address of the MAM Server Host if the
# MAM PostgreSQL server is on a separate host from the MAM server.
host    all             all             127.0.0.1/32     md5
host    all             all             ::1/128          md5
---
```

3. If the MAM Database Host is installed on a *different* host from where you will install the MAM Server, configure PostgreSQL to accept connections from the MAM Server Host.

```
[root]# vi /var/lib/pgsql/data/postgresql.conf

# Replace <mam-server-host> with the TCP/IP address on which the database server is
# to listen for connections
# from the MAM server. This will normally be the hostname or ip address of the MAM
# Database Host.
listen_addresses = '<mam-database-host>'
```

4. If your PostgreSQL database version is prior to version 9.1, configure postgresql to avoid interpreting backslashes as escape characters.

```
[root]# vi /var/lib/pgsql/data/postgresql.conf

standard_conforming_strings = on
```

5. Start or restart the database.

```
[root]# chkconfig postgresql on
[root]# service postgresql restart
```

## Install Dependencies, Packages, or Clients

Use the following instructions to install the required Moab Accounting Manager dependencies, packages, or clients.

**i** Depending on your configuration, the MAM Server Host and the MAM GUI Host may be installed on the same host. The MAM Client Host is automatically installed on the same host as the MAM Server Host; however, you can also install the MAM Client Host on any other hosts on which you want to have the MAM client commands available to users or administrators.

1. On the MAM Server Host, the MAM GUI Host, the MAM Web Services Host, and the MAM Client Hosts, do the following:

```
[root]# zypper install gcc lsb-release perl-Config-Tiny perl-Crypt-CBC perl-Crypt-DES
perl-Crypt-DES_EDE3 perl-Digest-HMAC perl-Error perl-JSON perl-Log-Dispatch
perl-Log-Dispatch-FileRotate perl-Log-Log4perl perl-Params-Validate perl-XML-LibXML
perl-YAML
```

2. On the MAM Server Host, do the following:

```
[root]# zypper install postgresql postgresql-libs perl-DBD-Pg perl-Date-Manip perl-DBI
```

3. On the MAM GUI Host, do the following:

```
[root]# zypper install apache2 perl-CGI perl-CGI-Session
```

4. On the MAM Web Services Host, do the following:

```
[root]# zypper install apache2 apache2-mod_perl
```

- On each of the MAM Client Hosts (including the MAM Server Host), do the following:

```
[root]# zypper install libopenssl-devel perl-TermReadLine-Gnu perl-TermReadKey
[root]# chmod 4755 /usr/bin/sperl*
```

**i** If any of the Perl module packages fail to install or are unavailable for your system, you can install it from CPAN by running `cpan MODULENAME` where **MODULENAME** is the respective perl module name.

## Install MAM Server

On the MAM Server Host, do the following:

- Create a user called `mam` and switch to that user.

```
[root]# useradd -m mam
[root]# su - mam
[mam]$ mkdir src
[mam]$ cd src
```

- Download the latest MAM build from the [Adaptive Computing Moab HPC Suite Download Center](https://www.adaptivecomputing.com/support/download-center/moab-hpc-suite-download/) (<https://www.adaptivecomputing.com/support/download-center/moab-hpc-suite-download/>).
- As the `mam` user, run each of the following commands in order.

```
[mam]$ tar -zxvf mam-9.1.1.tar.gz
[mam]$ cd mam-9.1.1
```

- Configure the software. For a list of all the configuration options, see [Moab Accounting Manager Configuration Options on page 65](#).

```
[mam]$ ./configure
```

- Compile the software.

```
[mam]$ make
```

- i** Depending on your configuration, you may need to replace "make" with a **make** command that includes additional functionality. Specifically:
- If you only need to install the clients on a particular system, use **make clients-only**.
  - If you only need to install the web GUI on a particular system, use **make gui-only**.
  - If you only need to install the web services on a particular system, use **make ws-only**.

## 6. Install the software.

```
[mam]$ exit
[root]# cd ~mam/src/mam-9.1.1
[root]# make install
```

- i** Depending on your configuration, you may need to replace "make install" with a **make** command that includes additional functionality. Specifically:
- If you only need to install the clients on a particular system, use **make install-clients-only**.
  - If you only need to install the web GUI on a particular system, use **make install-gui-only**.
  - If you only need to install the web services on a particular system, use **make install-ws-only**.

## 7. As the database user, create a database called `mam` and grant database privileges to the `mam` user.

- i** PostgreSQL should have previously been installed using the instructions in [Preparing for Manual Installation on page 26](#).

```
[root]# su - postgres
[postgres]$ psql

create database mam;
create user mam with password 'changeme!';
\q

[postgres]$ exit
```

The password you define must be synchronized with the `database.password` value in `/opt/mam/etc/mam-server.conf`

```
[root]# vi /opt/mam/etc/mam-server.conf
database.password = changeme!
```

8. Run the `hpc.sql` script to populate the Moab Accounting Manager database with objects, actions, and attributes necessary to function as an Accounting Manager.

```
[root]# su - mam
[mam]$ cd src/mam-9.1.1
[mam]$ psql mam < hpc.sql
[mam]$ exit
```

9. Configure MAM to automatically start up at system boot; start the `mam` service.

```
[root]# chkconfig --add mam
[root]# service mam start
```

## Configure the MAM GUI

If you plan to use the web GUI, then on the MAM GUI Host, do the following:

1. As `root`, add or edit the SSL virtual host definition as appropriate for your environment. To do so, configure the `cgi-bin` directory in `ssl.conf`. Below the `cgi-bin` directory element, create an alias for `/cgi-bin` pointing to your `cgi-bin` directory. If you chose to install to a `cgi-bin` sub-directory, you might want to create an alias for that as well. Also, add `index.cgi` to the `DirectoryIndex` so you can use the shorter sub-directory name.

```
[root]# a2enflag SSL
[root]# cp /etc/apache2/vhosts.d/vhost-ssl.template /etc/apache2/vhosts.d/mam-ssl.conf
[root]# vi /etc/apache2/vhosts.d/mam-ssl.conf

<Directory "/srv/www/cgi-bin">
## Add these lines
  Options ExecCGI
  AddHandler cgi-script .cgi
  AllowOverride All
  Order allow,deny
  Allow from all
</Directory>

# Aliases for /cgi-bin
Alias /cgi-bin/ /srv/www/cgi-bin/
Alias /mam /srv/www/cgi-bin/mam/

# Make shorter sub-dir name available
DirectoryIndex index.cgi
```

2. For the highest security, it is recommended that you install a public key certificate that has been signed by a certificate authority. The exact steps to do this are specific to your distribution and the chosen certificate authority. An overview of this process for CentOS 7 is documented at [https://www.centos.org/docs/7/html/7\\_1/sect02.html#sect02.01](#)

[https://access.redhat.com/documentation/en-US/Red\\_Hat\\_Enterprise\\_Linux/7/html/System\\_Administrators\\_Guide/ch-Web\\_Servers.html#s2-apache-mod\\_ssl](https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html/System_Administrators_Guide/ch-Web_Servers.html#s2-apache-mod_ssl).

Alternatively, if your network domain can be secured from man-in-the-middle attacks, you could use a self-signed certificate.

If your configuration uses self-signed certificates, do the following:

```
[root]# cd /etc/apache2
[root]# openssl genrsa -out ssl.key/server.key 1024
[root]# openssl req -new -key ssl.key/server.key -x509 -out ssl.crt/server.crt
```

### 3. Start or restart the HTTP server daemon.

```
[root]# chkconfig apache2 on
[root]# service apache2 restart
```

## Configure MAM Web Services

If you plan to use MAM Web Services, then on the MAM Web Services Host, do the following:

### 1. Edit the SSL virtual host definition in `ssl.conf` to include the `mamws` location. For example:

```
[root]# a2enflag SSL
[root]# a2enmod perl

# Do not repeat the following copy step if performed previously on the same host
when configuring the web server
[root]# cp /etc/apache2/vhosts.d/vhost-ssl.template /etc/apache2/vhosts.d/mam-
ssl.conf

[root]# vi /etc/apache2/vhosts.d/mam-ssl.conf

# Place the following within the 443 VirtualHost definition
PerlOptions +Parent
PerlSwitches -Mlib=/opt/mam/lib
PerlModule MAM::WSResponseHandler
PerlModule MAM::WSAuthenHandler
<Location /mamws>
    SetHandler perl-script
    PerlResponseHandler MAM::WSResponseHandler
    Options +ExecCGI

    AuthName MAM
    PerlAuthenHandler MAM::WSAuthenHandler
    Require valid-user

    Order allow,deny
    Allow from all
</Location>
```

### 2. For the highest security, it is recommended that you install a public key certificate that has been signed by a certificate authority. The exact steps to

do this are specific to your distribution and the chosen certificate authority. An overview of this process for CentOS 7 is documented at [https://access.redhat.com/documentation/en-US/Red\\_Hat\\_Enterprise\\_Linux/7/html/System\\_Administrators\\_Guide/ch-Web\\_Servers.html#s2-apache-mod\\_ssl](https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html/System_Administrators_Guide/ch-Web_Servers.html#s2-apache-mod_ssl).

Alternatively, if your network domain can be secured from man-in-the-middle attacks, you could use a self-signed certificate. Often this does not require any additional steps since in many distributions, such as Red Hat, the Apache SSL configuration provides self-signed certificates by default.

If your configuration uses self-signed certificates, do the following:

```
[root]# cd /etc/apache2
[root]# openssl genrsa -out ssl.key/server.key 1024
[root]# openssl req -new -key ssl.key/server.key -x509 -out ssl.crt/server.crt
```

### 3. Start or restart the HTTP server daemon.

```
[root]# chkconfig apache2 on
[root]# service apache2 restart
```

## Access the MAM GUI

If you plan to use the web GUI, then on the MAM Server Host, do the following:

### 1. Create a password for the `mam` user to be used with the MAM Web GUI.

```
[root]# su - mam
[mam]$ mam-set-password
[mam]$ exit
```

### 2. Verify the connection.

- a. Open a web browser and navigate to `https://<mam-server-host>/cgi-bin/mam`.
- b. Log in as the `mam` user with the password you set in step 1.

## Access MAM Web Services

If you plan to use MAM web services, then on a MAM Client Host, do the following:

### 1. Create a password for the `mam` user that you wish to access MAM Web Services.

```
[root]# su - mam
[mam]$ mam-set-password
[mam]$ exit
```

### 2. Make a call to web services.

```
[root]# curl -k -X GET --basic -u mam:changeme! 'https://<mam-web-services-host>/mamws/system'
```

Alternatively, for queries, you can use the browser to access the URL. For example: 'https://<mam-web-services-host>/mamws/system'.

## Configure Moab Workload Manager to Use Moab Accounting Manager

If integrating with Moab Workload Manager, do the following:

### 1. Configure Moab to talk to MAM

Do *one* of the following:

- **MAM Option.** If you are will be using the MAM (direct network) accounting manager interface with Moab Workload Manager (this is the default), do the following:
  - a. On the Moab Server Host, edit the Moab configuration file, uncomment the AMCFG lines and set the TYPE to MAM and set the HOST. If the Moab Server and the MAM Server are on the same host, set HOST to 'localhost'; otherwise, set HOST to the host name for the MAM Server (MAM Server Host).

```
[root]# vi /opt/moab/etc/moab.cfg
AMCFG[mam] TYPE=MAM HOST=<mam_server_host>
```

Customize additionally as needed. See [Accounting, Charging, and Allocation Management](#) in the *Moab Workload Manager Administrator Guide*

- b. Configure Moab to authenticate with MAM using the MAM secret key.
  - i. On the MAM Server Host, copy the auto-generated secret key from the token.value value in the `/opt/mam/etc/mam-site.conf` file.
  - ii. On the Moab Server Host, add the secret key to the moab-private.cfg file as the value of the CLIENTCFG KEY attribute.

```
[root]# vi /opt/moab/etc/moab-private.cfg
CLIENTCFG[AM:mam] KEY=<MAMSecretKey>
```

- **Native Option.** If you are will be using the Native (custom script) accounting manager interface with Moab Workload Manager, do the following:
  - a. On the Moab Server Host, edit the Moab configuration file, uncomment the AMCFG lines and set the TYPE to NATIVE.



```
[root]# vi /opt/moab/etc/moab.cfg
AMCFG[mam] TYPE=NATIVE
```

- b. If you are installing Moab Accounting Manager on a different host (MAM Server Host) from the Moab Server (Moab Server Host), you will need to install the Moab Accounting Manager client on the Moab Server Host in order for the custom scripts to use the MAM API.

On the *Moab* Server Host, follow the instructions in [Install Dependencies, Packages, or Clients on page 46](#) and [Install MAM Server on page 47](#); with the following exceptions:

- Install only the dependent packages applicable to MAM Client Hosts.
- Use the configure option `--without-init`.
- Instead of running `make`, use `make clients-only`.
- Instead of running `make install`, use `make install-clients-only`.
- Omit the step to create the database and all of the steps thereafter.

2. On the Moab Server Host, restart Moab.

```
service moab restart
```

## Initialize Moab Accounting Manager

You will need to initialize Moab Accounting Manager to function in the way that is most applicable to the needs of your site. See [Initial Setup](#) in the *Moab Accounting Manager Administrator Guide* to set up Moab Accounting Manager for your desired accounting mode.

Related Topics


- [Preparing for Manual Installation on page 26](#)


## Installing RLM Server

The RLM Server can run multiple licenses.

Access to a Reprise License Manager (RLM) server is required when using Moab's Elastic Computing Feature, Viewpoint's Remote Visualization Feature, or Nitro.

As the RLM Server can run multiple licenses, it is recommended that you install *one* RLM Server for your configuration. If your company already uses an RLM Server, you do not need to install a new one for Adaptive Computing products. However, Adaptive Computing *strongly* recommends that your RLM Server is version 12.1.2 and the Adaptive Computing products may use a different port than the default RLM Server port (5053).

 If your system configuration requires more than one RLM Server, additional configuration may be needed. See [Using Multiple RLM Servers on page 66](#) for more information.


 Expired RLM license files must be removed and the RLM service restarted whenever a new license is installed.

This topic contains instructions on how to install an RLM Server.

In this topic:

- [Open Necessary Ports on page 54](#)
- [Install the RLM Server on page 55](#)
- [Change the Default Passwords on page 55](#)

## Open Necessary Ports

 These instructions assume you are using the default ports. If your configuration will use other ports, then substitute your port numbers when opening the ports.

If your site is running firewall software on its hosts, you will need to configure the firewall to allow connections to the necessary ports.

Location	Ports	Functions	When Needed
RLM Server Host	5053	RLM Server Port	Always
RLM Server Host	5054	RLM Web Interface Port	Always
RLM Server Host	57889	Remote Visualization Port	If Remote Visualization is part of your configuration
RLM Server Host	5135	ISV adaptiveco Port (for the Adaptive license-enabled products)	For Moab Workload Manager <i>and</i> if Nitro is part of your configuration.

See [Opening Ports in a Firewall on page 63](#) for general instructions and an example of how to open ports in the firewall.

## Install the RLM Server

On the host where the RLM Server will reside, do the following:

1. Download the latest RLM build from the [Adaptive Computing Moab HPC Suite Download Center](https://www.adaptivecomputing.com/support/download-center/moab-hpc-suite-download/) (<https://www.adaptivecomputing.com/support/download-center/moab-hpc-suite-download/>).

2. Create a non-root user and group (rlm is used in the example).

```
[root]# groupadd -r rlm
[root]# useradd -r -g rlm -d /opt/rlm -c "A non-root user under which to run
Reprise License Manager" rlm
```

3. Create a directory and install the tarball files in that location (we are using /opt/rlm as the install location in the example).

```
[root]# mkdir -p -m 0744 /opt/rlm
[root]# cd /opt/rlm
[root]# tar -xzvf /tmp/ac-rlm-12.1.tar.gz --strip-components=1
[root]# chown -R rlm:rlm /opt/rlm
```

**i** The `--strip-components=1` removes the "ac-rlm-12.1/" from the relative path so that they are extracted into the current directory.

4. Install the startup scripts.

**i** If you are using a user:group other than rlm:rlm or a location other than /opt/rlm, then edit the following files to reflect those changes after copying them.

```
[root]# cp init.d/rlm /etc/init.d
```

5. Start the services and configure the RLM Server to start automatically at system reboot.

```
[root]# chkconfig --add rlm
[root]# chkconfig rlm on
[root]# service rlm start
```

## Change the Default Passwords

The RLM Web interface includes two usernames (admin and user) by default. These usernames have the default password "changeme!".

**!** If you do not change this password, RLM, and Remote Visualization, will not be secure. For tips on choosing a good password, see <https://www.us-cert.gov/ncas/tips/ST04-002>.

Do the following for *both* the user and the admin usernames:

1. Using a web browser, navigate to your RLM instance. (`http://<RLM_host>:5054`; where `<RLM_host>` is the IP address or name of the RLM Server Host).
2. Log in.
3. Select **Change Password** and change the password according to your password security process.

**i** The password for "user" will be needed as part of the Remote Visualization installation.

## Nitro Integration

This section provides instructions on integrating Nitro as part of your Moab configuration.

- [Preparing for Nitro Manual Installation on page 83](#)
- [Installing Nitro on page 57](#); Nitro Web Services is not available for a SUSE 11-based system.

### Preparing for Nitro Manual Installation

This topic contains instructions on how to download and unpack the Nitro Tarball Bundle for all the hosts in your configuration.

**i** Whether you are installing tarballs on one host or on several hosts, each host (physical machine) on which a server is installed (Nitro, Nitro Web Services) *must* have the Nitro Tarball Bundle.

**⚠** Nitro Web Services is currently not available for SUSE 11-based systems; it is not in the Nitro Tarball Bundle for that OS.

## Set Up Proxies

If your site uses a proxy to connect to the Internet, do the following:


```
export http_proxy=http://<proxy_server_id>:<port>
export https_proxy=http://<proxy_server_id>:<port>
```

## Download and Unpack the Nitro Tarball Bundle

The Nitro Tarball Bundle contains all the tarballs available for Nitro. However, not every tarball may be installed on the same host.

On each host (physical machine), do the following:

1. Using a web browser, navigate to the [Adaptive Computing Nitro Download Center](http://www.adaptivecomputing.com/support/download-center/nitro/) (<http://www.adaptivecomputing.com/support/download-center/nitro/>).
2. Download the Nitro Tarball Bundle `nitro-tarball-bundle-<version>-<OS>.tar.gz`.

 The variable marked `<version>` indicates the build's version, revision, and changeset information. The variable marked `<OS>` indicates the OS for which the build was designed.

3. Unpack the Nitro Tarball Bundle.

```
[root]# tar xzvf nitro-tarball-bundle-<version>-<OS>.tar.gz
```

#### Related Topics


- [Nitro Integration on page 56](#)

## Installing Nitro

This topic contains instructions on how to install Nitro.

### Nitro:

- needs to be available to all of the nodes that will be used as part of the Nitro job.
- can be installed either to each node individually *or* to a shared file system that each node can access.

 Some Nitro functionality, such as using the `nitrosub` command, is not available unless you are using a shared file system.

- can be installed to integrate with a scheduler, such as Moab, or without (Nitro standalone). The instructions are the same.

A shared file system allows Nitro task files to be copied to the compute nodes, which is where they will be needed when the job runs. Without a shared file system the Nitro task file will have to be copied to the location where Nitro expects it to be on the compute node manually. This is challenging if it is not known in advance on which compute node the job will run. In addition, a shared file system allows Nitro logs to be seen and opened on the head node, which allows Viewpoint to access them.


### In this topic:

- [Obtain a Nitro License on page 58](#)
- [Open Necessary Ports on page 59](#)

- [Install Nitro on page 60](#)
- [Verify Network Communication on page 62](#)

## Obtain a Nitro License

The Nitro license file is installed on an RLM Server. See [RLM Licensing Model on page 19](#) for more information about RLM licensing.

 These instructions assume you already have access to an RLM Server. See [Installing RLM Server on page 53](#) for instructions on how to set up a new RLM Server.

Do the following:

1. On the RLM server, obtain the hostid and hostname.

- hostid

```
[root]# /opt/rlm/rlmhostid
```

You should see output similar to the following.

```
rlmhostid v12.1  
Copyright (C) 2006-2016, Reprise Software, Inc. All rights reserved.  
  
Hostid of this machine: 00259096f004
```

- hostname

```
[root]# /opt/rlm/rlmhostid host
```

You should see output similar to the following.

```
rlmhostid v12.1  
Copyright (C) 2006-2016, Reprise Software, Inc. All rights reserved.  
  
Hostid of this machine: host=<your-host-name>
```

2. Email licenses@adaptivecomputing.com for a license and include the hostid and hostname you just obtained.
3. Adaptive Computing will generate the license and send you the Nitro license file (typically, nitro.lic) file in a return email.
4. On the RLM server, do the following:
  - a. Download and install the license file.

```
[root]# cd /opt/rlm  
[root]# chown rlm:rlm nitro.lic
```

- b. If the RLM Server in your configuration uses a firewall, edit the license file to reference the ISV adaptiveco port for the Adaptive license-enabled products. This is the same port number you opened during the RLM

Server installation. See the instructions to open necessary ports in the [Installing RLM Server on page 53](#) (manual installation method) or [1.1 Installing RLM Server](#) (RPM installation method) for more information.

```
[root]# vi /opt/rlm/nitro.lic
```

```
ISV adaptiveco port=5135
```

The license file already references the RLM Server port (5053 by default).

**i** If the RLM Server in your configuration uses different ports, you will need to modify the license file to reflect the actual ports. See the instructions to open necessary ports in the [Installing RLM Server on page 53](#) (manual installation method) or [1.1 Installing RLM Server](#) (RPM installation method) for more information.

- c. If you did *not* install an RLM Server using the file available from Adaptive Computing (for example, because your system configuration already uses one), do the following:
  - i. Download the 'adaptiveco.set' file from [the Adaptive Computing Nitro Download Center](https://www.adaptivecomputing.com/support/download-center/nitro/) (<https://www.adaptivecomputing.com/support/download-center/nitro/>).
  - ii. Copy the 'adaptiveco.set' file into the same directory where the Nitro license resides (/opt/rlm).
- d. Perform a reread to update the RLM Server with your license.

```
[root]# /opt/rlm/rlmreread
```

## Open Necessary Ports

Nitro uses several ports for communication between the workers and the coordinator.

If your site is running firewall software on its hosts, you will need to configure the firewall to allow connections to the necessary ports.

**i** The listed ports is for configurations that have only one coordinator. If multiple coordinators are run on a single compute host, then sets of ports (range of 4) must be opened for the number of expected simultaneous coordinators.

Location	Ports	Functions	When Needed
Compute Hosts (Nitro Coordinator)	47000	Coordinator/Worker communication	Always
Compute Hosts (Nitro Coordinator)	47001	Coordinator PUB/SUB channel - publishes status information	Always
Compute Hosts (Nitro Coordinator)	47002	Reserved for future functionality	
Compute Hosts (Nitro Coordinator)	47003	API communication channel	Always

See [Opening Ports in a Firewall on page 63](#) for general instructions and an example of how to open ports in the firewall.

## Install Nitro

**i** You **must** complete the tasks to obtain a Nitro license before installing Nitro. See [Obtain a Nitro License on page 58](#).

If your configuration uses firewalls, you *must also* open the necessary ports before installing Nitro. See [Open Necessary Ports on page 59](#).

On the host where Nitro will reside, do the following:

1. If you have not already done so, complete the steps to prepare the host. See [Preparing for Nitro Manual Installation on page 83](#).

2. Change the directory to the root of the unpacked Nitro tarball bundle.

```
[root]# cd nitro-tarball-bundle-<version>-<OS>
```

3. Identify the Nitro product tarball (nitro-<version>-<OS>.tar.gz).

4. As the root user, run each of the following commands in order.

```
[root]# mkdir /opt/nitro
[root]# tar xzvpf nitro-<version>-<OS>.tar.gz -C /opt/nitro --strip-components=1
```

5. Copy the license file you generated earlier in this topic to each compute node (coordinator). On each compute node, *or* on the shared file system, do the



following:

```
[root]# cp nitro.lic /opt/nitro/bin/
```

6. Copy the provided scripts and the nitrosub command from the /opt/nitro/scripts directory.

**i** This is a "copy" file operation and not a "move" operation. This allows you to customize your version and always have the factory version available for consultation and/or comparison.

- a. Copy the **launch\_nitro.sh** and **launch\_worker.sh** scripts for your resource manager to the bin directory. Each resource manager has a subdirectory with the scripts directory that contains the scripts. This example uses Torque as the resource manager.

```
[root]# cp /opt/nitro/scripts/torque/launch_nitro.sh /opt/nitro/bin/
[root]# cp /opt/nitro/scripts/torque/launch_worker.sh /opt/nitro/bin/
```

- b. Copy the nitrosub command to the bin directory.

```
[root]# cp /opt/nitro/scripts/nitrosub /opt/nitro/bin/
```

- c. Copy the **nitro\_job.sh** and the **worker\_job.sh** scripts to the etc directory.

```
[root]# cp /opt/nitro/scripts/nitro_job.sh /opt/nitro/etc/
[root]# cp /opt/nitro/scripts/worker_job.sh /opt/nitro/etc/
```

7. Now that you have copied the scripts and the nitrosub command, edit the copies for your site's administrative policies.
  - **bin/nitrosub** command (applicable only if using a shared file system). At a *minimum*, do the following:
    - a. Uncomment the "\_resource\_manager" line for your resource manager.
    - b. Uncomment the "resource\_type" line for your licensing model's allocation (nodes or cores).
    - c. If your system will be using dynamic jobs, set the "\_dynamic\_size" value to the number of resources to allocate to a dynamic job.

See nitrosub Command in the *Nitro Administrator Guide* for more information.
  - **bin/launch\_nitro.sh** and **bin/launch.worker.sh** scripts. See Launch Scripts in the *Nitro Administrator Guide* for more information.

8. If your system configuration allows multiple coordinators on the same node, additional configuration may be needed. See [Running Multiple Coordinators on the Same Node on page 67](#) for more information.
9. If you are *not* using a shared file system, copy the Nitro installation directory to *all* hosts.

```
[root]# scp -r /opt/nitro root@host002:/opt
```

**i** If you are not using a shared file system, you may not be able to use the nitrosub client command.

## Verify Network Communication

*Verify* that the nodes that will be running Nitro are able to communicate with the Nitro ports *and* that the nodes are able to communicate with one another.

Related Topics

- [Nitro Integration on page 56](#)

## Additional Configuration

In this section:

- [Opening Ports in a Firewall on page 63](#)
- [Moab Workload Manager Configuration Options on page 63](#)
- [Moab Accounting Manager Configuration Options on page 65](#)
- [Using Multiple RLM Servers on page 66](#)
- [Running Multiple Coordinators on the Same Node on page 67](#) (if Nitro is part of your configuration)

### Opening Ports in a Firewall

If your site is running firewall software on its hosts, you will need to configure the firewall to allow connections to the products in your installation.

This topic provides an example and general instructions for how to open ports in your firewall. The actual port numbers for the various products will be provided in the installation instructions for that product.

SUSE 11-based systems use SuSEfirewall2 as the default firewall software. If you use different firewall software, refer to your firewall documentation for opening ports in your firewall.

The following is an example of adding port 1234 when using SuSEfirewall2.

```
[root]# vi /etc/sysconfig/SuSEfirewall2
FW_SERVICES_EXT_TCP="1234"
[root]# service SuSEfirewall2_setup restart
```

### Moab Workload Manager Configuration Options

The following is a list of commonly used configure options. For a complete list, use `./configure --help` when configuring Moab.

Option	Description	Example
<code>--prefix</code>	Specifies the location of the binaries and libraries of the Moab install. The default location is <code>/opt/moab</code> .	<pre>[root]# ./configure --prefix=/usr/local</pre>

Option	Description	Example
<b>--with-am</b>	Specifies that you want to configure Moab with Moab Accounting Manager.	<pre>[root]# ./configure --with-am</pre>
<b>--with-am-dir</b>	Uses the specified prefix directory for the accounting manager if installed in a non-default location.	<pre>[root]# ./configure --with-am-dir=/opt/mam-9.1.1</pre>
<b>--with-flexlm</b>	Causes Moab to install the <b>license.mon.flexLM.pl</b> script in the <b>/opt/moab/tools</b> directory. For more information about this script, see the <a href="#">Interfacing to FLEXlm</a> section in the Moab Administrator Guide.	<pre>[root]# ./configure --with-flexlm</pre>
<b>--with-homedir</b>	Specifies the location of the Moab configuration directory and the MOABHOMEDIR environment variable. The default location is <b>/opt/moab</b> .  <div style="border: 1px solid black; padding: 5px; width: fit-content;"> <p><b>i</b> By default, MOABHOMEDIR is automatically set during installation. Use the <b>--without-profile</b> option to disable installed scripts.</p> </div>	<pre>[root]# ./configure --with-homedir=/var/moab</pre> <p><i>The Moab home directory will be /var/moab instead of the default /opt/moab.</i></p>
<b>--without-init</b>	Disables the installation of a distribution-specific, Moab service startup file. By default, make install will install an <b>init.d</b> or <b>systemd</b> service startup file as appropriate for your distribution. The installed file ( <b>/etc/init.d/moab</b> or <b>/usr/lib/systemd/system/moab.service</b> ) may be customized to your needs. If you do not want this file to be installed, use this option to exclude it.	<pre>[root]# ./configure --without-init</pre>

Option	Description	Example
<b>--without-profile</b>	Disables the installation of a distribution-specific shell profile for bash and C shell. By default, make install will install the Moab shell initialization scripts as appropriate for your operating system. These scripts help to establish the MOABHOMEDIR, PERLSLIB, PATH and MANPATH environment variables to specify where the new moab configuration, scripts, binaries and man pages reside. The installed scripts ( <code>/etc/profile.d/moab.{csh,sh}</code> ) may be customized to your needs. If you do not want these scripts to be installed, use this option to exclude them.	<pre>[root]# ./configure --without-profile</pre>

## Moab Accounting Manager Configuration Options

The following table comprises commonly-used configure options.

Option	Description
<b>-h,--help</b>	Run <code>./configure --help</code> to see the list of configure options.
<b>--localstatedir=DIR</b>	Home directory where per-configuration subdirectories (such as <code>etc</code> , <code>log</code> , <code>data</code> ) will be installed (defaults to <code>PREFIX</code> ).
<b>--prefix=PREFIX</b>	Base installation directory where all subdirectories will be installed unless otherwise designated (defaults to <code>/opt/mam</code> ).
<b>--with-cgi-bin=DIR</b>	If you intend to use the web GUI, use <code>--with-cgi-bin</code> to specify the directory where you want the Moab Accounting Manager CGI files to reside (defaults to <code>/var/www/cgi-bin/mam</code> ).
<b>--with-db-name=NAME</b>	Name of the SQL database that the server will sync with (defaults to <code>mam</code> ).
<b>--with-legacy-links</b>	Creates symbolic links allowing the use of the old client and server command names (for example, <code>mam-list-users</code> would be created as symbolic link to <code>mam-list-users</code> ). When running a command under its old name, the command will issue a deprecation warning. This warning can be disabled by setting <code>client.deprecationwarning = false</code> in the <code>mam-client.conf</code> file. The default is not to install the legacy links.

Option	Description
<b>--with-mam-libs=local site</b>	Use <code>--with-mam-libs</code> to indicate whether you want to install the Perl MAM modules in a local directory ( <code>\${exec_prefix}/lib</code> ) or in the default system site-perl directory (defaults to <code>local</code> ).
<b>--with-promotion=mamauth suidperl</b>	Command-line clients and scripts using the API need to use a security promotion method to authenticate and encrypt the communication using the symmetric key. The default is <code>suidperl</code> if it is installed on the system, otherwise the default is <code>mamauth</code> . See the description for the <code>security.promotion</code> configuration parameter in the <a href="#">Client Configuration</a> section for more information about the two security promotion methods.
<b>--with-user=USER</b>	Use <code>--with-user</code> to specify the accounting admin userid that the server will run under and who will have full administrative privileges (defaults to <code>mam</code> ). It is recommended that this be a non-privileged user for the highest security.
<b>--without-gui</b>	Specifies whether to install the CGI web GUI. If you do not intend to use the CGI web GUI, you can specify <code>--without-gui</code> to not install the CGI scripts. Otherwise, the default is to install the GUI CGI scripts.
<b>--without-init</b>	If you do not intend to use the <code>mam init.d</code> service, you can use <code>--without-init</code> to specify that Moab should not install the <code>mam init.d</code> script. Otherwise, the script is installed by default.
<b>--without-profile</b>	If you do not intend to use the <code>mam profile.d</code> environment scripts, you can use <code>--without-profile</code> to specify that Moab should not install the <code>mam profile.d</code> scripts. Otherwise, the scripts are installed by default.

## Using Multiple RLM Servers

As the RLM Server can run multiple licenses, it is recommended that you install *one* RLM Server for your configuration.

However, if your configuration requires more than one RLM Server, you will *need* to configure the Adaptive Computing products to connect to a specific RLM Server. If not configured to connect to a specific RLM Server, the Adaptive Computing product will scan the network and connect to the first RLM Server it finds listening to request the license. If the first RLM Server does *not* have the product's license, the RLM connection will fail.

If you are using multiple RLM Servers, do the following to configure the an Adaptive Computing product to connect to a specific RLM Server:

1. Modify the RLM Server not to accept the network search connections.
  - Edit the init script in `/opt/rlm/` to add `-noudp`.

```
start() {
su -l $rlmuser -s /bin/bash -c "$rlmdir/rlm -l -dlog $debuglog -noudp &"
}
```

2. Enable the Adaptive Computing product to connect to a specific RLM.

On the host where the Adaptive Computing product resides, do the following:

- a. Create a new text file and name it with the `.lic` extension (typically, `remote.lic`) and save it in the same location as the other Adaptive Computing licenses. Be careful not to override an existing license.
- b. Edit the new `remote.lic` file to point to the specific RLM Server hostname and port. Port 5053 is the default. If you use a different port number for the RLM Server, specify that port number in the `remote.lic` file.

```
HOST <hostname> ANY 5053
```

Repeat as needed for each Adaptive Computing product that you want to connect to a specific RLM Server.

## Running Multiple Coordinators on the Same Node

Nitro provides the ability to run multiple coordinators on the same node.

**i** Running multiple coordinators on the same node is not available if your system configuration uses a policy to limit nodes to a single job (i.e., `NODEACCESSPOLICY=SINGLEJOB` on Moab).

If your system is configured to allow multiple coordinators on the node:


- It is recommended that you instruct your users to submit Nitro jobs using the `nitrosub` command. See [nitrosub Command](#) for more information.
- If you prefer that your users do *not* use the `nitrosub` command, and instead you prefer that they submit the Nitro jobs directly to your scheduler/resource manager, then you will need to add the `--port-file` option to the `bin/launch_nitro.sh` and `bin/launch_worker.sh` scripts to ensure that all coordinators will be able to run.


```
NITRO_OPTIONS="--port-file --job-id ${NITROJOBID} ${NITRO_OPTIONS}"
```

Add the `--port-file` option *before* the `--job-id` information.

# Manual Upgrade

This section provides instructions and other information when upgrading your for installing your Moab components for SUSE 11-based systems using the Manual upgrade method.

 It is highly recommended that you *first* perform upgrades in a *test environment*. Installation and upgrade procedures are tested prior to release; however, due to customizable variations that may be utilized by your configuration, it is not recommended to drop new versions of software directly into production environments. This is especially true when the workload has vital bearing. Contact Adaptive Computing Professional Services for more information.

 Because many system-level files and directories are accessed during the upgrade, the upgrade instructions in this guide should be executed with root privileges.

You will see that the instructions execute commands as the root user. Please note that the same commands will work for a non-root user with the `sudo` command.

In this section:

- [Upgrading Torque Resource Manager on page 68](#)
- [Upgrading Moab Workload Manager on page 75](#)
- [Upgrading Moab Accounting Manager on page 77](#)
- [Upgrading RLM Server on page 82](#)
- [Upgrading Your Nitro Integration on page 83](#)
- [Migrating the MAM Database from MySQL to PostgreSQL on page 86](#)

## Upgrading Torque Resource Manager

Torque 6.1 binaries are backward compatible with Torque 5.0 or later. However they are not backward compatible with Torque versions prior to 5.0. When you upgrade to Torque 6.1.1.1 from versions prior to 5.0, all MOM and server daemons must be upgraded at the same time.

The job format is compatible between 6.1 and previous versions of Torque and any queued jobs will upgrade to the new version. It is not recommended to upgrade Torque while jobs are in a running state.



This topic contains instructions on how to upgrade and start Torque Resource Manager (Torque).

**i** If you need to upgrade a Torque version prior to 4.0, contact Adaptive Computing.

**i** See [1.1 Considerations Before Upgrading](#) in the *Torque Resource Manager Administrator Guide* for additional important information, including about how to handle running jobs during an upgrade, mixed server/MOM versions, and the possibility of upgrading the MOMs without having to take compute nodes offline.

In this topic:

- [Before You Upgrade on page 69](#)
- [Stop Torque Services on page 70](#)
- [Upgrade the Torque Server on page 71](#)
- [Update the Torque MOMs on page 72](#)
- [Update the Torque Clients on page 73](#)
- [Start Torque Services on page 74](#)
- [Perform Status and Error Checks on page 74](#)

## Before You Upgrade

This section contains information of which you should be aware before upgrading.

In this section:

- [Running Jobs on page 69](#)
- [Cray Systems on page 70](#)
- [hwloc on page 70](#)


## Running Jobs

Before upgrading the system, all running jobs must complete. To prevent queued jobs from starting, nodes can be set to offline or all queues can be disabled (using the "started" queue attribute). See [pbsnodes](#) or [Queue Attributes](#) in the *Torque Resource Manager Administrator Guide* for more information.

## Cray Systems

For upgrading Torque to 6.1.1.1 on a Cray system, refer to the [Installation Notes for Moab and Torque for Cray](#) in Appendix G of the *Moab Workload Manager Administrator Guide*.

## hwloc

 Using "zypper install hwloc" may install an older, non-supported version.

When cgroups are enabled (recommended), hwloc version 1.9.1 or later is required. NVIDIA K80 requires libhwloc 1.11.0. If cgroups are to be enabled, check the Torque Server Host to see if the required version of hwloc is installed. You can check the version number by running the following command:

```
[root]# hwloc-info --version
```

The following instructions are for installing version 1.9.1.

If hwloc is not installed or needs to be upgraded to the required version, do the following:

1. On the Torque Server Host, *each* Torque MOM Host, and *each* Torque Client Host, do the following:
  - a. Download hwloc-1.9.1.tar.gz from <https://www.open-mpi.org/software/hwloc/v1.9>.
  - b. Run each of the following commands in order.

```
[root]# zypper install gcc make
[root]# tar -xzvf hwloc-1.9.1.tar.gz
[root]# cd hwloc-1.9.1
[root]# ./configure
[root]# make
[root]# make install
```

2. Run the following commands on the Torque Server Host, *each* Torque MOM Host, and *each* Torque Client Host.

```
[root]# echo /usr/local/lib >/etc/ld.so.conf.d/hwloc.conf
[root]# ldconfig
```


## Stop Torque Services

Do the following:

1. On the Torque Server Host, shut down the Torque server.

```
[root]# service pbs_server stop
```

2. On *each* Torque MOM Host, shut down the Torque MOM service.


 Confirm all jobs have completed before stopping `pbs_mom`. You can do this by typing `"momctl -d3"`. If there are no jobs running, you will see the message "NOTE: no local jobs detected" towards the bottom of the output. If jobs are still running and the MOM is shutdown, you will only be able to track when the job completes and you will not be able to get completion codes or statistics.

```
[root]# service pbs_mom stop
```

3. On *each* Torque Client Host (including the Moab Server Host, the Torque Server Host, and the Torque MOM Hosts, if applicable), shut down the `trqauthd` service.

```
[root]# service trqauthd stop
```

## Upgrade the Torque Server

 You *must* complete all the previous upgrade steps in this topic before upgrading Torque server. See the list of steps at the beginning of this topic.

On the Torque Server Host, do the following:

1. Back up your `server_priv` directory.

```
[root]# tar -cvf backup.tar.gz TORQUE_HOME/server_priv
```


2. If not already installed, install the Boost C++ headers.

```
[root]# zypper install boost-devel
```

3. Download the latest Torque build from the [Adaptive Computing](#) website.
4. Depending on your system configuration, you will need to add `./configure` command options.

At a minimum, you add:

- `--enable-cgroups`
- `--with-hwloc-path=/usr/local` See [Torque on page 14](#) for more information.

 These instructions assume you are using `cgroups`. When `cgroups` are supported, `cpusets` are handled by the `cgroup cpuset` subsystem. If you are not using `cgroups`, use `--enable-cpusets` instead.

See [Customizing the Install](#) in the *Torque Resource Manager Administrator Guide* for more information on which options are available to customize the `./configure` command.

## 5. Install the latest Torque tarball.

```
[root]# cd /tmp
[root]# tar xzvf torque-6.1.1.1.tar.gz
[root]# cd torque-6.1.1.1
[root]# ./configure --enable-cgroups --with-hwloc-path=/usr/local # add any other
specified options
[root]# make
[root]# make install
```

## Update the Torque MOMs

Do the following:

### 1. On the Torque Server Host, do the following:

- a. Create the self-extracting packages that are copied and executed on your nodes.

```
[root]# make packages
Building ./torque-package-clients-linux-x86_64.sh ...
Building ./torque-package-mom-linux-x86_64.sh ...
Building ./torque-package-server-linux-x86_64.sh ...
Building ./torque-package-gui-linux-x86_64.sh ...
Building ./torque-package-devel-linux-x86_64.sh ...
Done.
```

The package files are self-extracting packages that can be copied and executed on your production machines. Use `--help` for options.

- b. Copy the self-extracting mom package to *each* Torque MOM Host.

Adaptive Computing recommends that you use a remote shell, such as SSH, to install packages on remote systems. Set up shared SSH keys if you do not want to supply a password for each Torque MOM Host.

```
[root]# scp torque-package-mom-linux-x86_64.sh <torque-mom-host>:
```

### 2. On *each* Torque MOM Host, confirm that cgroups have been mounted; if not, mount them.

- a. Run `lssubsys -am`.
- b. If the command is not found, or you do not see something similar to the following, then cgroups are *not* mounted, continue with these instructions.

```
ns
perf_event
net_prio
cpuset /cgroup/cpuset
cpu /cgroup/cpu
cpuacct /cgroup/cpuacct
memory /cgroup/memory
devices /cgroup/devices
freezer /cgroup/freezer
net_cls /cgroup/net_cls
blkio /cgroup/blkio
```

- c. Do the following:
  - i. Install the cgroup library package.

```
[root]# zypper install libcgroup1
```

- ii. Edit `/etc/cgconfig.conf` and add the following:

```
mount {
    devices = /mnt/cgroups/devices;
    cpuset = /mnt/cgroups/cpuset;
    cpu = /mnt/cgroups/cpu;
    cpuacct = /mnt/cgroups/cpuacct;
    memory = /mnt/cgroups/memory;
}
```

- iii. Mount cgroups.

```
[root]# service cgconfig start
[root]# chkconfig cgconfig on
```

- d. Run `lssubsys -am` again and confirm cgroups are mounted.
3. On *each* Torque MOM Host, do the following:
  - a. Install cgroup-tools.
  - b. Install the self-extracting MOM package.

```
[root]# ./torque-package-mom-linux-x86_64.sh --install
```

## Update the Torque Clients

This section contains instructions on updating the Torque clients on the Torque Client Hosts (including the Moab Server Host and Torque MOM Hosts, if applicable).

1. On the Torque Server Host, do the following:
  - a. Copy the self-extracting client package to *each* Torque Client Host.

Adaptive Computing recommends that you use a remote shell, such as SSH, to install packages on remote systems. Set up shared SSH keys if you do not want to supply a password for each Torque MOM Host.

```
[root]# scp torque-package-clients-linux-x86_64.sh <torque-client-host>:
```

- b. If Moab Workload Manager is part of your configuration, copy the self-extracting devel package to the Moab Server Host.

```
[root]# scp torque-package-devel-linux-x86_64.sh <moab-server-host>:
```

2. On *each* Torque Client Host, do the following:

**i** This step can be done from the Torque server from a remote shell, such as SSH. Set up shared SSH keys if you do not want to supply a password for each Torque Client Host.

```
[root]# ./torque-package-clients-linux-x86_64.sh --install
```

3. If Moab Workload Manager is part of your configuration, do the following on the Moab Server Host:

```
[root]# ./torque-package-devel-linux-x86_64.sh --install
```

## Start Torque Services

Do the following:

1. On *each* Torque Client Host (including the Moab Server Host, Torque Server Host and Torque MOM Hosts, if applicable), start up the trqauthd service.

```
[root]# service trqauthd start
```

2. On *each* Torque MOM Host, start up the Torque MOM service.

```
[root]# service pbs_mom start
```

3. On the Torque Server Host, start up the Torque server.

```
[root]# service pbs_server start
```

## Perform Status and Error Checks

On the Torque Server Host, do the following:

- Verify that the status of the nodes and jobs are as expected.

```
[root]# pbsnodes
[root]# qstat
```

## Upgrading Moab Workload Manager

This topic provides instructions to upgrade Moab Workload Manager to the latest release version. Depending on which version of Moab you are presently running, upgrade instructions may vary.

Moab Workload Manager uses the standard `configure`, `make`, and `make install` steps for upgrades. This topic provides a number of sample steps referenced to a particular installation on a Linux platform using the bash shell. These steps indicate the user ID in brackets performing the step. The exact commands to be performed and the user that issues them will vary based on the platform, shell, installation preferences, and other factors.

It is highly recommended that you *first* perform upgrades in a *test environment*. See the warning in [1.1 Preparing for Upgrade](#). It is also recommend that you verify the policies, scripts, and queues work the way you want them to in this test environment. See [Testing New Releases and Policies](#) in the *Moab Workload Manager Administrator Guide* for more information.

If you are also upgrading Torque from an older version (pre-4.0), contact Adaptive Computing.

**i** Because many system-level files and directories are accessed during the installation, the instructions in this guide should be executed with root privileges.

You will see that the instructions execute commands as the root user. Please note that the same commands will work for a non-root user with the `sudo` command.

## Upgrade Moab Workload Manager

On the Moab Server Host, do the following:

1. If you have not already done so, install extra packages from the add-on repositories. See [Enable Extra Packages for the Repository on page 27](#) for more information.
2. Download the latest Moab build from the [Adaptive Computing](#) website.
3. Untar the distribution file. For example:

```
[root]# tar -xzvf moab-9.1.1-<os>.tar.gz
```

**i** The variable marked `<os>` indicates the OS for which the build was designed.

4. Change directory into the extracted directory.

```
[root]# cd moab-9.1.1-<OS>
```

5. Configure the installation package.

Use the same configure options as when Moab was installed previously. If you cannot remember which options were used previously, check the `config.log` file in the directory where the previous version of Moab was installed from.

For a complete list of configure options, use `./configure --help`.

6. Stop Moab.

```
[root]# service moab stop
```

**i** While Moab is down, all currently running jobs continue to run on the nodes, the job queue remains intact, and new jobs cannot be submitted to Moab.

7. Back up your Moab Workload Manager home directory (`/opt/moab/` by default) before continuing.
8. If you are using green computing, or if you are using a resource manager other than Torque, run the `make perldeps` command to install the necessary perl modules using CPAN.

**i** CPAN is installed on SUSE-based systems by default.

```
[root]# make perldeps
```

9. Install Moab.

```
[root]# make install
```

**i** Default configuration files are installed during `make install`. Existing configuration files are not overwritten and the new files are given a `.dist` extension.

10. If you use ODBC, you must confirm the database schema compatibility. See [Migrating Your Database to Newer Versions of Moab](#) in the *Moab Workload Manager Administrator Guide* for more information.
11. Verify the version number is correct before starting the new server version.

```
[root]# moab --about
```

You should get something similar to the following in the response:



```

Defaults:  server=:42559  cfgdir=/opt/moab (env)  vardir=/opt/moab
Build dir:  /tmp/jenkins/workspace/MWM-9.1.1/label/build-<OS>
Build host: us-devops-build10
Build date: Oct 09 13:00:00 MST 2017
Build args: NA
Compiler Flags:  -D_M64 -D_BUILDDATETIME="2017100913" -DMUSEZEROMQ -
DMUSEWEBSERVICES -DMUSEMONGODB -DMMAX_GRES=512 -DMMAX_RANGE=2048 -DMMAX_TASK=32768
-fPIC -gdwarf-3 -Wall -Wextra -DVALGRIND -Og -x c++ -std=c++11 -DDMAX_PJOB=512 -D_
GNU_SOURCE
Compiled as little endian.
Version: moab server 9.1.1 (revision 2017100913, changeset
14dee972ebcee919207e48054e9f285db9f6a555)

```

12. 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. See [AMCFG Parameters and Flags](#) in the *Moab Workload Manager Administrator Guide* for more information.

**i** In Moab Workload Manager 8.1 and after, Moab defaults to using a set of stock scripts that no longer need to be explicitly configured in the server configuration file.

13. Start Moab.

```
[root]# service moab start
```

## Upgrading Moab Accounting Manager

This topic provides instructions to upgrade MAM to the latest release version. It includes instructions for migrating your database schema to a new version if necessary.

Moab Accounting Manager uses the standard `configure`, `make`, and `make install` steps for upgrades. This document provides a number of sample steps referenced to a particular installation on a Linux platform using the bash shell. These steps indicate the user ID in brackets performing the step. The exact commands to be performed and the user that issues them will vary based on the platform, shell, installation preferences, and other factors.

### Upgrade Moab Accounting Manager

On the MAM Server Host, do the following:

1. Determine the MAM Accounting admin user and change to that user.
  - If you are upgrading MAM from a version *prior* to 9.0, use `glsuser`.

```
[root]# glsuser | grep 'Accounting Admin'
mam      True
Accounting Admin
[root]# su - mam
```

- If you are upgrading MAM from a version at or after 9.0, use `mam-list-users`.

```
[root]# mam-list-users | grep 'Accounting Admin'
mam      True
Accounting Admin
[root]# su - mam
```

2. Determine whether you need to migrate your database.

- a. Determine your database version.

- If you are upgrading MAM from a version *prior* to 9.0, run **goldsh System Query**.

```
[mam]$ goldsh System Query
```

- If you are upgrading MAM from a version at or after 9.0, run **mam-shell System Query**.

```
[mam]$ mam-shell System Query
```

- b. If the current version is lower than 9.1, you must migrate your database. The steps required to do so are incorporated in the remaining steps for this topic.

3. Stop the server daemon.

```
[mam]$ su -c "service mam stop"
```

4. If you determined that you must migrate your database, create a database backup.

```
[mam]$ pg_dump -U <mam_database_user> -W <old_database_name> > /tmp/<old_database_name>.sql
```

**i** MySQL is no longer a supported database for MAM. If you are using MySQL for your MAM database, follow the instructions in [Migrating the MAM Database from MySQL to PostgreSQL on page 86](#) to convert your database.

5. If your PostgreSQL database version is prior to version 9.1, update the `postgresql` configuration to avoid interpreting backslashes as escape

characters.


```
[root]# vi /var/lib/pgsqli/data/postgresql.conf
standard_conforming_strings = on
[root]# service postgresql restart
```

6. Verify that each of the prerequisites listed in [Installing Moab Accounting Manager on page 43](#) have been satisfied.
7. Download the latest MAM build from the [Adaptive Computing](#) website.
8. Unpack the tar archive and change directory into the top directory of the distribution.

```
[mam]$ tar -zxvf mam-9.1.1.tar.gz
[mam]$ cd mam-9.1.1
```

9. Configure Moab Accounting Manager by running `configure` with the desired options.

It is recommended that you use the same configure options that were used in the previous installation. You can examine the `config.log` file where you unpacked your previous distribution to help determine the configuration options that were used to install the prior version of MAM.

 Client and server command names changed beginning with 9.0. If you want to create symbolic links to enable you to continue to use the old client and server command names, use the `--with-legacy-links` option with `configure`. When running a command under its old name, the command will issue a deprecation warning. This warning can be disabled by setting `client.deprecationwarning = false` in the `mam-client.conf` file.

```
[mam]$ ./configure
```

10. Run `make` to compile the program.

```
[mam]$ make
```

- i** Depending on your configuration, you may need to replace "make" with a make command that includes additional functionality. Specifically:
- If you only need to install the clients on a particular system, use `clients-only`.
  - If you only need to install the web GUI on a particular system, use `make gui-only`.
  - If you only need to install the web services on a particular system, use `make ws-only`

## 11. Run `make install` as root to install Moab Accounting Manager.

```
[mam]$ su -c "make install"
```

- i** Depending on your configuration, you may need to replace "make install" with a make command that includes additional functionality. Specifically:
- If you only need to install the clients on a particular system, use `make install-clients-only`.
  - If you only need to install the web GUI on a particular system, use `make install-gui-only`.
  - If you only need to install the web services on a particular system, use `make install-ws-only`

## 12. Edit the configuration files as necessary. You may want to compare your existing configuration files with those distributed with the new release to determine if you want to merge and change any of the new options within your configuration files.

- If you are upgrading MAM from a version *prior* to 9.0, the install process will have saved your prior configuration files to `{goldd,gold,goldg}.conf.pre-9.0` and written new default server configuration file as `mam-{server,client,gui}.conf`. You will need to merge any non-default parameters from your prior config files to the new default config files.

```
[mam]$ diff /opt/mam/etc/goldd.conf.pre-9.0 /opt/mam/etc/mam-server.conf
[mam]$ vi /opt/mam/etc/mam-server.conf
[mam]$ diff /opt/mam/etc/gold.conf.pre-9.0 /opt/mam/etc/mam-client.conf
[mam]$ vi /opt/mam/etc/mam-client.conf
[mam]$ diff /opt/mam/etc/goldg.conf.pre-9.0 /opt/mam/etc/mam-gui.conf
[mam]$ vi /opt/mam/etc/mam-gui.conf
```

- If you are upgrading MAM from a version at or after 9.0, merge and change any of the new options supplied in the new default configuration

files (saved in `mam-{server,client,gui}.conf.dist`) into your existing configuration files (`mam-{server,client,gui}.conf`).

```
[mam]$ diff /opt/mam/etc/mam-server.conf /opt/mam/etc/mam-server.conf.dist
[mam]$ vi /opt/mam/etc/mam-server.conf
[mam]$ diff /opt/mam/etc/mam-client.conf /opt/mam/etc/mam-client.conf.dist
[mam]$ vi /opt/mam/etc/mam-client.conf
[mam]$ diff /opt/mam/etc/mam-gui.conf /opt/mam/etc/mam-gui.conf.dist
[mam]$ vi /opt/mam/etc/mam-gui.conf
```

- If you are upgrading MAM from a version at or after 9.1, and you are using MAM Web Services, merge and change any of the new options supplied in the new default MAM Web Services configuration file (saved in `mam-ws.conf.dist`) into your existing MAM Web Services configuration file (`mam-ws.conf`).

```
[mam]$ diff /opt/mam/etc/mam-ws.conf /opt/mam/etc/mam-ws.conf.dist
[mam]$ vi /opt/mam/etc/mam-ws.conf
```

### 13. Start the server daemon.

```
[mam]$ su -c "service mam start"
```

14. If you are migrating your database to 9.1, you will do so by running one or more migration scripts. You must run every incremental migration script between the version you are currently using and the new version (9.1). These scripts are designed to be rerunnable, so if you encounter a failure, resolve the failure and rerun the migration script. If you are unable to resolve the failure and complete the migration, contact Support.

For example, if you are migrating from Moab Accounting Manager version 7.2, you must run six migration scripts: the first to migrate the database schema from 7.2 to 7.3, the second to migrate from 7.3 to 7.5, the third to migrate the database schema from 7.5 to 8.0, the fourth to migrate the database schema from 8.0 to 8.1, the fifth to migrate the database schema from 8.1 to 9.0, and the sixth to migrate the database schema from 9.0 to 9.1.

```
[mam]$ sbin/migrate_7.2-7.3.pl
[mam]$ sbin/migrate_7.3-7.5.pl
[mam]$ sbin/migrate_7.5-8.0.pl
[mam]$ sbin/migrate_8.0-8.1.pl
[mam]$ sbin/migrate_8.1-9.0.pl
[mam]$ sbin/migrate_9.0-9.1.pl
```

### 15. Verify that the resulting database schema version is 9.1.

```
[mam]$ mam-shell System Query
```

Name	Version	Description
Moab Accounting Manager	9.1	Commercial Release

16. Verify that the executables have been upgraded to 9.1.1.

```
[mam]$ mam-server -v
Moab Accounting Manager version 9.1.1
```

17. If you are upgrading MAM from a version prior to 9.1.0, and you wish to use MAM Web Services, perform the following procedures (provided in the Installing Moab Accounting Manager topic):
  - [Configure MAM Web Services on page 50](#)
  - [Access MAM Web Services on page 51](#)

## Upgrading RLM Server

Adaptive Computing *strongly* recommends that your RLM Server is version 12.1.2.

In this topic:

- [Confirm if an Upgrade is Needed on page 82](#)
- [Upgrade the RLM Server on page 82](#)


### Confirm if an Upgrade is Needed

Run the following command to determine your current version of RLM Server.

```
[root]# /opt/rlm/rlm -v
```

If the version reported is less than 12.1.2, continue with the section to Upgrade the RLM Server later in this topic.

### Upgrade the RLM Server

 These instructions assume you used /opt/rlm as the install location.

On the RLM Server Host, do the following:

1. Download the latest RLM build from the [Adaptive Computing Moab HPC Suite Download Center](#) (<https://www.adaptivecomputing.com/support/download-center/moab-hpc-suite-download/>).

2. Stop the RLM service.

```
[root]# service rlm stop
```

3. Archive the existing RLM installation, including the license file(s).

```
[root]# mv /opt/rlm/ /opt/rlm-<archive_version>/
```

#### 4. Install the new tarball files.

```
[root]# mkdir -p -m 0744 /opt/rlm
[root]# cd /opt/rlm
[root]# tar -xzvf /<unpack-directory>/ac-rlm-12.1.tar.gz --strip-components=1
[root]# chown -R rlm:rlm /opt/rlm
```

**i** The `--strip-components=1` removes the "ac-rlm-12.1/" from the relative path so that they are extracted into the current directory.

#### 5. Install the startup scripts.

**i** If you are using a user:group other than `rlm:rlm` or a location other than `/opt/rlm`, then edit the following files to reflect those changes after copying them.

```
[root]# cp init.d/rlm /etc/init.d
```

#### 6. Restore the license file(s).

```
[root]# cp /opt/rlm-<archive_version>/*.lic /opt/rlm/
```

#### 7. Restart the RLM service.

```
[root]# service rlm restart
```

## Upgrading Your Nitro Integration

This section provides instructions on upgrading your Nitro Integration as part of your Moab configuration.


In this section:

- [Preparing for Nitro Manual Installation on page 83](#)
- [Upgrading Nitro on page 84](#); Nitro Web Services is not available for a SUSE 11-based system

### Preparing for Nitro Manual Installation

This topic contains instructions on how to download and unpack the Nitro Tarball Bundle for all the hosts in your configuration.

**i** Whether you are installing tarballs on one host or on several hosts, each host (physical machine) on which a server is installed (Nitro, Nitro Web Services) *must* have the Nitro Tarball Bundle.

 Nitro Web Services is currently not available for SUSE 11-based systems; it is not in the Nitro Tarball Bundle for that OS.

## Set Up Proxies

If your site uses a proxy to connect to the Internet, do the following:


```
export http_proxy=http://<proxy_server_id>:<port>
export https_proxy=http://<proxy_server_id>:<port>
```

## Download and Unpack the Nitro Tarball Bundle

The Nitro Tarball Bundle contains all the tarballs available for Nitro. However, not every tarball may be installed on the same host.

On each host (physical machine), do the following:

1. Using a web browser, navigate to the [Adaptive Computing Nitro Download Center](http://www.adaptivecomputing.com/support/download-center/nitro/) (<http://www.adaptivecomputing.com/support/download-center/nitro/>).
2. Download the Nitro Tarball Bundle `nitro-tarball-bundle-<version>-<OS>.tar.gz`.

 The variable marked `<version>` indicates the build's version, revision, and changeset information. The variable marked `<OS>` indicates the OS for which the build was designed.

3. Unpack the Nitro Tarball Bundle.

```
[root]# tar xzvf nitro-tarball-bundle-<version>-<OS>.tar.gz
```

Related Topics

- [Nitro Integration on page 56](#)

## Upgrading Nitro

This topic contains instructions on how to upgrade Nitro.

In this topic:

- [Upgrade Nitro on page 84](#)
- [Verify Network Communication on page 86](#)

## Upgrade Nitro

On the Nitro Host, do the following:



1. If you have not already done so, complete the steps to prepare the host. See [Preparing for Nitro Manual Installation on page 83](#).
2. Back up your existing launch script in `/opt/nitro/bin/`.
3. Change the directory to the root of the unpacked Nitro tarball bundle.

```
[root]# cd nitro-tarball-bundle-<version>-<OS>
```

4. Identify the Nitro product tarball (`nitro-<version>-<OS>.tar.gz`) and unpack the tarball into the same directory you created when you first installed Nitro (for example, `/opt/nitro`).

```
[root]# tar xzvpf nitro-<version>-<OS>.tar.gz -C /opt/nitro --strip-components=1
```

5. Copy the provided scripts and the `nitrosub` command from the `/opt/nitro/scripts` directory.

**i** This is a "copy" file operation and not a "move" operation. This allows you to customize your version and always have the factory version available for consultation and/or comparison.

- a. Copy the `launch_nitro.sh` and `launch_worker.sh` scripts for your resource manager to the `bin` directory. Each resource manager has a subdirectory with the scripts directory that contains the scripts. This example uses Torque as the resource manager.

```
[root]# cp /opt/nitro/scripts/torque/launch_nitro.sh /opt/nitro/bin/
[root]# cp /opt/nitro/scripts/torque/launch_worker.sh /opt/nitro/bin/
```

- b. Copy the `nitrosub` command to the `bin` directory.

```
[root]# cp /opt/nitro/scripts/nitrosub /opt/nitro/bin/
```

- c. Copy the `nitro_job.sh` and the `worker_job.sh` scripts to the `etc` directory.

```
[root]# cp /opt/nitro/scripts/nitro_job.sh /opt/nitro/etc/
[root]# cp /opt/nitro/scripts/worker_job.sh /opt/nitro/etc/
```

6. Merge any customizations from your existing launch scripts, job scripts, and the `nitrosub` command (if applicable) into the new launch scripts, job scripts, and the `nitrosub` command that you copied from the `scripts` directory.
7. If your system configuration allows multiple coordinators on the same node, additional configuration may be needed. See [Running Multiple Coordinators on the Same Node on page 67](#) for more information.
8. If you are not using a shared file system, copy the updated Nitro installation directory to *all* hosts.

```
[root]# scp -r /opt/nitro root@host002:/opt
```

**i** If you are not using a shared file system, you may not be able to use the nitrosub client command.

## Verify Network Communication

Verify that the nodes that will be running Nitro are able to communicate with the Nitro ports *and* that the nodes are able to communicate with one another.

Related Topics

- [Upgrading Your Nitro Integration on page 83](#)

## Migrating the MAM Database from MySQL to PostgreSQL

PostgreSQL is the preferred DBMS for MAM. Customers who have already installed MySQL as the DBMS for MAM are not required to migrate their database to use PostgreSQL at this time. However, MySQL is considered deprecated and new installations will only use PostgreSQL.

**i** PostgreSQL does not provide a standard procedure for migrating an existing database from MySQL to PostgreSQL. Adaptive Computing has had success using the py-mysql2pgsql tools for migrating/converting/exporting data from MySQL to PostgreSQL. See <https://github.com/philipsoutham/py-mysql2pgsq> for additional details.

### To Migrate the MAM Database

This procedure was successfully tested on an actual customer MySQL database with millions of transactions on CentOS 6.4. It completed in less than an hour.

1. Make a backup copy of your MySQL mam database.

```
[root]# mysqldump mam > /archive/mam.mysql
```

2. Follow the instructions to Install PostgreSQL.
  - **Manual Install** - [1.1 Installing Moab Web Services](#)
  - **RPM Install** - [1.1 Installing Moab Web Services](#)
3. Install the prerequisite packages.
4. Install pg-mysql2pgsql (from source).

```
[root]# cd /software
[root]# git clone git://github.com/philipsoutham/py-mysql2pgsql.git
[root]# cd py-mysql2pgsql
[root]# python setup.py install
```

5. Run `pg-mysql2pgsql` once to create a template yaml config file.

```
[root]# py-mysql2pgsql -v
```

6. Edit the config file to specify the MySQL database connection information and a file to output the result.

```
[root]# vi mysql2pgsql.yml
```

```
mysql:
hostname: localhost
port: 3306
socket:
username: mam
password: changeme
database: mam
compress: false
destination:
# if file is given, output goes to file, else postgres
file: /archive/mam.pgsql
postgres:
hostname: localhost
port: 5432
username:
password:
database:
```

7. Run the `pg-mysql2pgsql` program again to convert the database.

```
[root]# py-mysql2pgsql -v
```

8. Create the `mam` database in PostgreSQL.

```
[root]# su - postgres
[postgres]$ psql
postgres=# create database "mam";
postgres=# create user mam with password 'changeme!';
postgres=# \q
[postgres]$ exit
```

9. Import the converted data into the PostgreSQL database.

```
[root]# su - mam
[mam]$ psql mam < /archive/mam.pgsql
```

10. Point MAM to use the new postgresql database.

```
[mam]$ cd /software/mam-latest
[mam]$ ./configure # This will generate an etc/mam-
server.conf.dist file
[mam]$ vi /opt/mam/etc/mam-server.conf # Merge in the database.datasource from
etc/mam-server.conf.dist
```

### 11. Restart Moab Accounting Manager.

```
[mam]$ mam-server -r
```

## Chapter 3 Troubleshooting

This chapter details some common problems and general solutions. Additional troubleshooting may be found in the individual Moab component documentation.

In this chapter:

- [General Issues on page 89](#)
- [Moab Workload Manager Issues on page 95](#)

### General Issues

This topic details some common problems and general solutions.

In this topic:

- [Where do I need to set credentials and what are the default values? on page 89](#)
  - [Database Credentials on page 90](#)
  - [Product Credentials on page 92](#)

### Where do I need to set credentials and what are the default values?

Communication and cooperation between various components of the Moab requires credentials to be properly configured. For ease of use, the credential information, including where credentials are set, default values, and where they are used are grouped by database and product.

In this section:

[Database Credentials on page 90](#)

[Product Credentials on page 92](#)

## Database Credentials

### MongoDB

Database	User	Default Password	Used By	Parameters
admin	admin_user	secret1	system admins	NA
moab	moab_user	secret2	/opt/moab/etc/moab-private.cfg	MONGOUSER, MONGOPASSWORD
moab	mws_user	secret3	/opt/mws/etc/mws-config.groovy	grails.-mongo.username, grails.-mongo.password
moab	insight_user	secret4	/opt/insight/etc/config.groovy	moab.-mongo.username, moab.-mongo.password
mws	mws_user	secret3	/opt/mws/etc/mws-config.groovy	grails.-mongo.username, grails.-mongo.password
insight	insight_user	secret4	/opt/insight/etc/config.groovy	mongo.username, mongo.password
insight	mws_user	secret3	http://<mws_server->:8080/mws/admin/plugins/edit/viewpoint-query-helper	user, password
nitro-db	nitro_user	secret5	/opt/nitro-web-services/etc/nitro.cfg	db_username, db_password
reporting	reporting_user	secret6	/opt/reporting/application.conf	database.username, database.password

**i** The following characters must be escaped in strings in the `/opt/insight/etc/config.groovy` and `/opt/mws/etc/mws-config.groovy` files (such as when used in a password): `\` (backslash), `"` (double quote), `'` (single quote), `$` (dollar sign). Example: `mongo.password="my\$cool\$password"`. It is recommended that you avoid using these characters.

## PostgreSQL

Database	User	Default Password	Used By	Parameters
moab_viewpoint	moab_viewpoint	changeme!	/opt/viewpoint/etc/viewpoint.cfg	VIEWPOINT_DATABASE_USER, VIEWPOINT_DATABASE_PASSWORD
mam	mam	changeme!	/opt/mam/etc/mam-server.cfg	database.user, database.password

## Apache Drill

The Drill host should have a user that Reporting Web Services can use to authenticate to Drill

Host	User	File	Parameter Name	Default Value
Drill host	drilluser	/opt/reporting-web-services/etc/application.properties	reporting.rest.drill.username	changeme!

## Product Credentials

### Moab Workload Manager

Declared Parameter		Used By		Default Value
File	Parameter Name	File	Parameter Name	
/opt/moab/etc/moab-private.cfg	MESSAGEQUEUESECRETKEY	/opt/mws/etc/mws-config.groovy	moab.messageQueue.secretKey	NA
		/opt/insight/etc/config.groovy	messageQueue.secretKey	
/opt/moab/etc/.moab.key	NA	/opt/mws/etc/mws-config.groovy	moab.secretKey	NA

### Moab Accounting Manager

Declared Parameter		Used By		Default Value
File	Parameter Name	File	Parameter Name	
/opt/mam/etc/mam-site.conf	token.value	/opt/moab/etc/moab-private.cfg	CLIENTCFG [AM:mam] KEY	NA



## Moab Web Services

Declared Parameter		Used By		Default Value
File	Parameter Name	File	Parameter Name	
/opt/mws/etc/mws-config.groovy	auth.defaultUser.username	http://<viewpoint_server>:8081/configuration/	Username	moab-admin
		/opt/moab/etc/moab-private.cfg	CLIENTCFG [RM:mws] USERNAME	
/opt/mws/etc/mws-config.groovy	auth.defaultUser.password	http://<viewpoint_server>:8081/configuration/	Password	change-me!
		/opt/moab/etc/moab-private.cfg	CLIENTCFG [RM:mws] PASSWORD	
/opt/mws/etc/mws-config.groovy	grails.plugin.springsecurity.oauthProvider.clients[0].clientSecret	http://<viewpoint_server>:8081/configuration/	Client Secret	NA

**i** The following characters must be escaped in strings in the `/opt/insight/etc/config.groovy` and `/opt/mws/etc/mws-config.groovy` files (such as when used in a password): `\` (backslash), `"` (double quote), `'` (single quote), `$` (dollar sign). Example: `mongo.password="my\$cool\$password"`. It is recommended that you avoid using these characters.

## Nitro Web Services

Declared Parameter		Used By	Default Value
File	Parameter Name		
/opt/nitro-web-services/etc/nitro.cfg	ws_admin_password	Installation - default NWS API user creation	ChangeMe2!
/opt/nitro-web-services/etc/nitro.cfg	ws_readonly_username	Installation - default NWS API user creation http://<viewpoint_server>:8081/configuration/ -> Nitro Services -> Username	nitro-readonly-user
/opt/nitro-web-services/etc/nitro.cfg	ws_readonly_password	Installation - default NWS API user creation http://<viewpoint_server>:8081/configuration/ -> Nitro Services -> Password	ChangeMe3!
/opt/nitro-web-services/etc/nitro.cfg	ws_writeonly_username	Installation - default NWS API user creation /opt/nitro-web-services/etc/zmq_job_status_adapter.cfg -> username	nitro-writeonly-user
/opt/nitro-web-services/etc/nitro.cfg	ws_writeonly_password	Installation - default NWS API user creation /opt/nitro-web-services/etc/zmq_job_status_adapter.cfg -> password	ChangeMe4!

## Viewpoint

Declared Parameter		Used By	Default Value
File	Parameter Name		
/opt/viewpoint/etc/viewpoint.cfg	username	http://<viewpoint_server>:8081/login/	viewpoint-admin
/opt/viewpoint/etc/viewpoint.cfg	password	http://<viewpoint_server>:8081/login/	changeme!

## Moab Workload Manager Issues

This topic details some common problems and general solutions for Moab Workload Manager.

See also Troubleshooting and System Maintenance in the *Moab Workload Manager Administrator Guide*.

In this topic:

- [Moab error: "cannot determine local hostname" on page 95](#)
- [Moab error: "Moab will now exit due to license file not found" on page 95](#)

### Moab error: "cannot determine local hostname"

```
# service moab start
Starting moab: ERROR:      cannot determine local hostname - node is misconfigured
                        [FAILED]
```

```
...
SCHEDCFG[Moab]           SERVER=<moab-hostname>:42559
...
```

Also check `/etc/hosts` to be sure the host name resolves, at least with localhost:

```
...
127.0.0.1  <moab-hostname> localhost localhost.localdomain localhost4
localhost4.localdomain4
...
```

### Moab error: "Moab will now exit due to license file not found"

```
# service moab start
Starting moab: Moab will now exit due to license file not found
Please contact Adaptive Computing (sales@adaptivecomputing.com) to get a license for
your system
                        [FAILED]
```

If you encounter this error when starting Moab, make sure your Moab license file is named `moab.lic` and is located in the `/opt/moab/etc/` directory.

Also make sure the license is not expired. The expiration date is listed in the license file. For example:

```
# cat /opt/moab/etc/moab.lic
...
# Expires after Tue Dec 31 10:43:46 2013
...
```

