Data Management Platform (DMP) Administrator's Guide

S–2327–C

RECORD OF REVISION

S–2327–C Published May 2014. Supports ESM XX-3.0.0 (SLES11SP3, Bright 6.1), ESF-XX-2.2.0, ESL-XX-2.2.0 software releases.

S–2327–B Published November 2013. Updated to support new features in ESM XX-2.1.0.

S–2327–A Published June 2013. Original printing.

Changes to this Document

Data Management Platform (DMP) Administrator's Guide S–2327–C

This rewrite of the *Data Management Platform (DMP) Administrator's Guide* supports the ESM XX-3.0.0 (SLES11SP3, Bright 6.1), ESF XX-2.2.0, ESL XX-2.2.0 software releases.

S–2327–C

Added information

- Added support for DNE and esfsmon 2.0.0.
- A new finalize script (`esf_finalize.sh`) supports all CLFS nodes.
- Added independent procedures for updating slave node distributions, ESL or ESF software, and CIMS node distribution.

Revised information

- Slave node BIOS settings changed to prevent slave nodes from powering up automatically after a power failure (CIMS node should power on first).
- Corrections to Lustre migration procedure (1.8.x to 2.x)
- Incorporated corrections to Lustre Monitoring Tool (LMT) section.

Contents

Page

Page

Page

Page

Page

Page

Procedures

Page

Page

Page

Examples

Page

Tables

Figures

Page

Page

Introduction [1]

The Cray Data Management Platform (DMP) integrates Cray XC30, or Cray XE and Cray XK systems into a customer user environment using 1U and 2U commodity rack-mounted service nodes. DMP nodes run a combination of commercial off-the-shelf Linux™ software and Cray proprietary software to replicate the Cray Linux Environment (CLE) for application development and testing. These specialized *external* login and service nodes expand the role of the Cray system *internal* login nodes and provide a development platform for shared externalized file systems, data movers, or high-availability configurations. A Cray Integrated Management Services (CIMS) node runs the commercially available Bright Cluster Manager® software (Bright). Bright software enables administrators to manage many DMP nodes using the Bright CMDaemon (`cmd`), cluster management shell (`cmsh`) or the cluster management GUI (`cmgui`). External shared file systems remain accessible to Cray DMP nodes, regardless of the state of the Cray system.

An overview of the Bright software and how it is used to manage a Cray DMP system is discussed in Chapter 2, Bright Cluster Manager® on page 41.

1.1 Scope of the Manual

This document provides procedures and common administration tasks that target Cray's implementation of Bright to manage a DMP system. Cray DMP systems are designed to function as special purpose service nodes for a Cray system, and are not designed as compute clusters. Note that Cray's implementation of Bright leverages its capability to manage Cray's specialized service nodes and not a traditional compute cluster. It follows some features of Bright (such as *cloudbursting*) are not implemented in Cray DMP systems.

Refer to the *Bright Cluster Manager 6.1 Administrator Manual* stored on the CIMS node in the `/cm/shared/docs/cm` directory. This document provides detailed information about the Bright software, and describes how to use the user interface (`cmgui`) and management shell (`cmsh`) to perform common administrative tasks.

1.2 Related Publications

The following documents contain additional information that may be helpful:

- *Installing Cray Integrated Management Services (CIMS) Software* (S–2522)

- *Installing Cray Development and Login (CDL) Software* (S–2520)
- *Installing Lustre File System by Cray (CLFS) Software* (S–2521)
- *Installing CLE Support Package on a Cray Development and Login (CDL) Node* (S–2528)
- *Managing System Software for the Cray Linux Environment* (S–2393), which is provided with your CLE release package
- *Managing Lustre for the Cray Linux Environment (CLE)* (S–0010), which is provided with your Cray Linux Environment (CLE) operating system release package
- *Bright Cluster Manager 6.1 Administrator Manual*. A PDF of this document is stored on the CIMS node in `/cm/shared/docs/cm`
- DELL™ R620, R720, R820, and R815 *Hardware Owners Guides*, available from www.dell.com/support

1.3 External Services Node Name Changes

The Cray® External Services product line has been rebranded to Cray Data Management Platform (DMP). The node names in Bright Cluster Manager® (Bright) `esms`, `esfs`, and `eslogin` will continue to be used throughout the documentation examples as long as these node names are consistent with the installation and management software. The new nomenclature is used when nodes are described in a generic sense.

Table 1. Data Management Platform Node Name Changes

Old Name	New Name	Description
esMS node	CIMS node	A Cray Integrated Management Services (CIMS) node is the centralized management node that aggregates data and manages slave node power, provisioning, reporting, health, and software images.
esLogin node	CDL node	A Cray Development and Login (CDL) node is a secure multi-user application development platform with access to the same shared file system as internal Cray login nodes.
esFS node	CLFS node	A Cray Lustre® File System (CLFS) node is a customized Lustre storage node that provides connectivity to DataDirect™ Networks (DDN) or NetApp™ storage devices. CLFS nodes are typically configured as MGS, MDS, or OSS nodes.

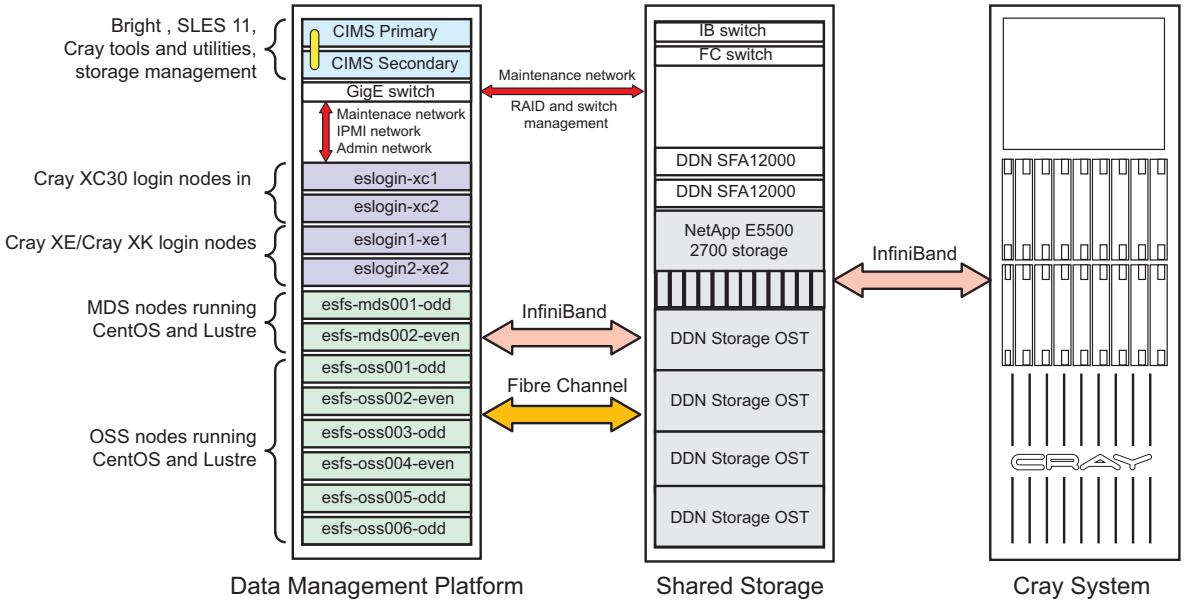
1.4 System Overview

Cray Data Management Platform (DMP) systems expand the role of Cray internal login nodes by providing a software development platform and shared storage for users. In order to expand the role of Cray system *internal* login nodes, a Cray DMP system must provide a software login and development platform for two distinct software environments (Cray XE and Cray XC30 systems), provide high-performance shared file systems that are accessible to both the Cray DMP system and Cray system. It follows, that to simulate the software development environment for both Cray XE and Cray XC30 systems, two DMP Cray Development and Login (CDL) node types must exist: one for Cray XE series systems and one for Cray XC30 series systems. The Cray Linux Environment (CLE) and Cray Developer Toolkit software for either a Cray XE or Cray XC30 system is installed on the CDL nodes, along with other commands and utilities to emulate the software development environment that exists on a Cray internal login node.

DMP systems support Cray Lustre® file system (CLFS) nodes that are configured to support a shared Lustre file system for the Cray system and each CDL node. This file system remains available to developers on the DMP system regardless of the operational state of the Cray system.

Cray DMP systems use Bright software on a Cray Integrated Management Services (CIMS) node to manage all of the nodes, software images, and networks in the system. Two CIMS nodes can be configured for high-availability (HA) configurations. Bright enables a system administrator to create custom categories and groups for system objects such as nodes, networks, software images, and other system entities, and manage them all from a single location. The Bright GUI (`cmgui`) or Bright command-line shell (`cmsh`) can manage a very large system with many different node types, software images, networks, and hardware configurations from a single interface.

Figure 1. DMP System Hardware Overview



1.5 Major Software Components of Cray DMP Systems

Each node type is supported by a separate software release that includes an operating system and custom Cray software to support its role in the DMP system.

1.5.1 SLES11SP3 Operating System

The SUSE Linux Enterprise Server (SLES™) Service Pack 3 (SP3) operating system is the base operating system for the Cray Integrated Management Services (CIMS) node and the Cray Development and Login (CDL) node. The SLES11SP3 operating system is installed using the Cray Linux Environment (CLE) release media.

1.5.2 Bright Cluster Manager 6.1 Software

The Bright Cluster Manager® 6.1 software (Bright) manages the hardware and software for all the devices and nodes in a system. It supports a GUI (`cmgui`) and command line shell (`cmsh`). The Bright 6.1 software and SLES11SP3 operating system are installed on the CIMS node and provided as bootable media with the ESM XX-3.0.0 software release.

1.5.3 Cray Integrated Management Services (CIMS) Software

Cray Integrated Management Services (CIMS) software (ESM) includes CIMS software, operating system updates, and Cray tools and utilities. CIMS software is provided with the ESM XX-3.0.0 software release.

1.5.4 CDL Node Software

The Cray Development and Login (CDL) software (ESL) requires the SLES11SP3 base operating system from the Cray Linux Environment (CLE 5.2) release media. CLE 5.2 media is used to install SLES11SP3 on either a managed (controlled by a CIMS) or unmanaged CDL node (a node that is not managed by the CIMS). Lustre client software is included to support Lustre 2.5 file systems. OFED is built by Cray from CLE and OFED.

After the base operating system is installed, ESL XC-2.2.0 and ESL XE-2.2.0 software is installed to emulate the development environment of both Cray XE and Cray XC30 internal login nodes (CDL nodes may also be referred to as external login nodes). ESL releases are tied to specific CLE releases.

The CLE Support Package software (CLESP) is installed on the CDL node. CLESP enables the Cray Programming Environment software to be installed. The Cray Programming Environment software enables users to develop, compile, execute, debug, and analyze code on Cray XE, Cray XK, Cray XC30 or Cray XC30-AC systems.

1.5.5 Lustre File System by Cray (CLFS) Software

The Cray CLFS node software release (ESF XX-2.2.0) requires the Community Enterprise Operating System (CentOS 6.4) and Lustre 2.5 software. This software is obtained directly from Wamcloud. Cray builds the Linux kernel and OFED package.

1.5.6 Install DMP Software

All Cray DMP software is installed from Cray DVD media or ISO images. Software installation procedures are documented for each type of DMP node. Refer to the installation documentation listed in Related Publications on page 19.

1.5.7 Determine the Current Software Releases

The current ESM, ESF, or ESL software release installed is stored in the `/etc/opt/cray/release/` directory. For example, the contents of the `/etc/opt/cray/release/ESMrelease` file may contain `ESM-XX-3.0.0-201404062004`. Refer to Distribution Media on page 23 for a description of the release naming nomenclature.

1.5.8 Distribution Media

The following release media is available for DMP system nodes:

CIMS node media

- `ESM-XX-`*`N.N.N-DatestampVer`*`.iso` — ESM XX-3.0.0 release media. Required for CIMS node initial installations and upgrades. This media contains SLES security updates and CIMS software and tools.
- `bright6.1-sles11sp1.iso` — This media is required to upgrade Bright 6.0 to version 6.1 on a CIMS node running SLES11SP1 and Cray XE CDL nodes. This upgrade must occur before the operating system is upgraded to SLES11SP2.
- `bright6.1-sles11sp2.iso` — This media is required to upgrade Bright 6.0 to version 6.1 on a CIMS node running SLES11SP2. This upgrade must occur before the operating system is upgraded to SLES11SP3 and ESM software is upgraded to ESM XX-3.0.0.
- `bright6.1-sles11sp3.iso` — Bright software and base operating system release media. This bootable ISO (DVD) is required to do an initial install of the SLES11SP3 base operating system and Bright 6.1 software on a CIMS node. In addition, the ESL installer requires this ISO to install Bright RPMs on the CDL software image.
- `bright6.1-centos6u4.iso` — This media is required to upgrade ESF software images to Bright 6.1.
- `SLES-11-SP3-DVD-x86_64-GM-DVD1.iso` — SLES11SP3 release media. Required to upgrade ESM software to SLES11SP3.
- `SLE-11-SP3-SDK-DVD-x86_64-GM-DVD1.iso` — The SLE 11 SDK release media contains the SUSE Linux Enterprise software developer kit. Required to upgrade ESM software to SLES11SP3.

Cray uses the following convention for DMP ISO file names:

`PROD-ARCH-N.N.N-DatestampVer``.iso`

PROD	Product name, such as `ESM`
ARCH	Supported architecture: `XX` for all architectures, `XC` for Cray XC30 systems, and `XE` for Cray XE and XK systems
N.N.N	Release number, such as `3.0.0` for ESM XX-3.0.0
Datestamp	Unique date stamp in the ISO name, in the format *YYYYMMDDHHmm* (such as `201309252116`)
Ver	Installer version, such as `a12`

CDL node media

- The new CLE 5.2 release media contains the SUSE Linux Enterprise Server (SLES) Service Pack 3 (SP3) operating system and CLE base software. The bootable DVD/ISO used to install SLES11SP3 is labeled `Cray-CLEbase11SP3-20140319.iso`.

The CLE 5.2 release media also includes a DVD labeled `Cray CLE 5.2.UP`*nn* `Software` or an ISO image named `xc-sles11sp3-5.2.`*nnavv*`.iso`, or `xe-sles11sp3-5.2.`*nnavv*`.iso`, where `xe` or `xc` indicate the architecture, `5.2.`*nn* indicates the CLE release build level, and *avv* indicates the installer version. This media provides the `CrayCLEinstall.sh` installer software and utilities to upgrade the base operating system software.

Specifically, CLE 5.2 release media is used to:

- Perform an initial install of SLES11SP3 on a managed CDL node using the ESL installer script `ESLinstall`. A managed CDL node is one which has a software image managed by a CIMS.
- Update a managed CDL node from SLES11SP1 or SLES11SP2 to SLES11SP3 to enable ESL XC-2.2.0 or ESL XE-2.2.0 to be installed.
- Burn a bootable DVD and install the SLES11SP3 base operating system on a disconnected or stand-alone CDL node. A disconnected CDL node was originally installed using an CIMS and then disconnected. After SLES11SP3 is installed, install the ESL XC-2.2.0 or ESL XE-2.2.0 software, then the Cray Linux Environment Support Package (CLESP), and finally, the Cray Developer Environment.
- The ESL release media contains the CDL node software and tools.

  For ESL XC-2.2.0, this item is available as the ISO `ESL XC-2.2.0-`*DatestampVer*`.iso` (for example, `ESL-XC-2.2.0-201401311221a12.iso`.

  For ESL XE-2.2.0, this item is available as the ISO `ESL XE-2.2.0-`*DatestampVer*`.iso`.
- The CLE 5.2 Support Package is required for the Cray Developer Environment:
  - For ESL XC-2.2.0, use the CLE 5.2 release media, for example `xc-sles11sp3-5.2.08b01.iso`.
  - For ESL XE-2.2.0, use the CLE 5.2 release media, for example `xc-sles11sp3-5.2.08b01.iso`
- The Cray Developer Environment:
  - For ESL XC-2.2.0, use the latest Cray Developer Toolkit (CDT) release media.
  - For ESL XE-2.2.0, use the latest Cray Application Developer's Environment (CADE) release media.

CLFS node media

- The Cray ESF software release media contains CLFS software and CentOS 6.4.

For ESF XX-2.2.0, this item is available as an ISO such as `ESF-XX-2.2.0-`*DatestampVer*`.iso`

- ESF XX-2.2.0 requires that the `bright6.1-centos6u4.iso` ISO file exist in `/root/isos` on the CIMS node.
- The CentOS 6.4 release media contains the CentOS™ operating system.

  ESF XX-2.2.0 requires that the `CentOS-6.4-x86_64-bin-DVD1.iso` ISO file exist in `/root/isos` on the CIMS node.

1.5.9 Tools and Utilities

Refer to the UNIX® man pages for each of the utilities in this section for command line options.

1.5.9.1 `esdumpsys`

The `esdumpsys` command performs a dump of the specified CLFS and/or CDL in a Cray DMP system.

`esdumpsys` optionally generates a system memory dump if `kdump` is configured on the specified system(s). Refer to Configure kdump on CDL Nodes (SLES) on page 170 to configure `kdump` for SLES. Refer to Configure kdump on CentOS™ on page 217 to configure `kdump` for CentOS. If `kdump` is not configured, the nodes will crash and reboot, but no memory dump will be generated. Information prints to the console in either case.

The `esdumpsys` command uses `ssh` and `scp` to gather information from the specified systems. Password-less `ssh` is used to connect to the specified systems.

1.5.9.2 `ESMupdateimage`

The `ESMupdateimage` command updates slave node software images from the ESM media after ESM software is updated, but within the same Bright version.

Refer to the man page or Update Slave Node RPMs From ESM Media on page 71 for more information about the `ESMupdateimage` command.

1.5.9.3 `CIMSupgradeImages`

The `CIMSupgradeImages` command upgrades slave node software images created in Bright 6.0 to software images that support Bright 6.1.

1.5.9.4 eswrap

The eswrap utility is a wrapper that lets users access a subset of Cray Linux Environment (CLE) and Programming Environment (PE) commands from a CDL node. eswrap uses Secure Shell (SSH) to launch the wrapped command on the Cray system, then displays the output on the CDL node so that it appears to the user that the wrapped command is actually running on a Cray internal login node.

The CDL installation process creates a symbolic link for each wrapped command in the directory /opt/cray/eslogin/eswrap/default/bin. Each symbolic link points to the eswrap command, so that running a wrapped command (such as apstat) actually runs eswrap with the wrapped command as an argument. eswrap uses ssh to run the command on the specified node of the Cray system (by default, the internal login node with the hostname login, unless the $ESWRAP_LOGIN environment variable specifies a different node). Refer to the man page for eswrap on the CDL for more information.

1.5.9.5 cray-esfs-catman

The cray-esfs-catman utility is a script that helps change the Bright category settings for multiple CLFS nodes at the same time. cray-esfs-catman creates and runs the necessary Bright cmsh commands to change a category setting for either metadata server (MDS) or object storage server (OSS) nodes in the specified Lustre file system.

1.5.9.6 esfsmon_failback

The esfsmon_failback command returns a failed CLFS node to operational status. When a CLFS node has been failed over to its backup node, the failed node is automatically powered down and placed into a failed node category. After the failed CLFS node has been repaired, the administrator must use esfsmon_failback to return the node to service. Refer to the esfsmon_failback man page on the CIMS for more information.

You must be root user to run the esfsmon_failback command. Refer to Configure CLFS Failover (esfsmon 2.0.0) on page 203 for more information.

1.5.9.7 update_excludelist

The update_excludelist script changes a Bright exclude list for all slave nodes in the specified category or categories. An exclude list controls which files in a slave image are retained or excluded during image synchronization, such as when a slave node is rebooted. update_excludelist composes and issues the necessary cmsh commands to change all nodes in a category at the same time. Refer to the man page on the CIMS for more information.

1.6 DMP Networks Overview

Cray DMP software uses *internal* and *external* designations to classify networks. The `esmain-net`, `ipmi-net`, for example are internal networks accessible only to the CIMS. External networks in a DMP system are `site-user-net`, and `site-admin-net`, which enable users from outside the system to gain access.

Refer to Network Settings on page 103 for common network configuration tasks.

Figure 2 shows an overview of the hardware components and networks used in a Cray DMP system. The list below describes the primary networks used in a DMP system. The Bright software provides built-in classifications for the various networks in a system, and a Cray DMP system uses primarily the internal, external, and management classifications. There are other network classifications within Bright, such as cloud and global, but these are not used. There may be additional networks defined, depending on the requirements of the system.

`esmaint-net`

An internal management network that connects the CIMS server with the slave nodes, switches, and RAID controllers. This network enables Bright to manage and provision the slave nodes and other devices in the DMP system. When using the Bright GUI (`cmgui`) or Cluster Management Shell (`cmsh`) this network is classified as the internal management network.

`ipmi-net` Internal Intelligent Platform Management Interface (IPMI) or DELL™ Remote Access Controller (DRAC) network that provides remote console and power management of the slave nodes from the CIMS.

`site-admin-net`

External administration network used by site administrators to log in to the CIMS server (typically on the same network as the Cray SMW). The name and IP address of this network are customized during installation. The CIMS IPMI interface (iDRAC) may also be on this network to provide remote console and power management of the CIMS node.

`site-user-net`

External user (site) network used by the slave nodes. On CDL nodes, this network provides user access and authentication services such as LDAP. On CLFS MDS nodes, this network connects to the site LDAP for file ownership authentication. The name and IP address of this network are customized during installation. Connections to additional site-specific networks are optional.

`wlm-net` External user (site) network used by CDL nodes to access Cray SDB node or Cray internal login nodes.

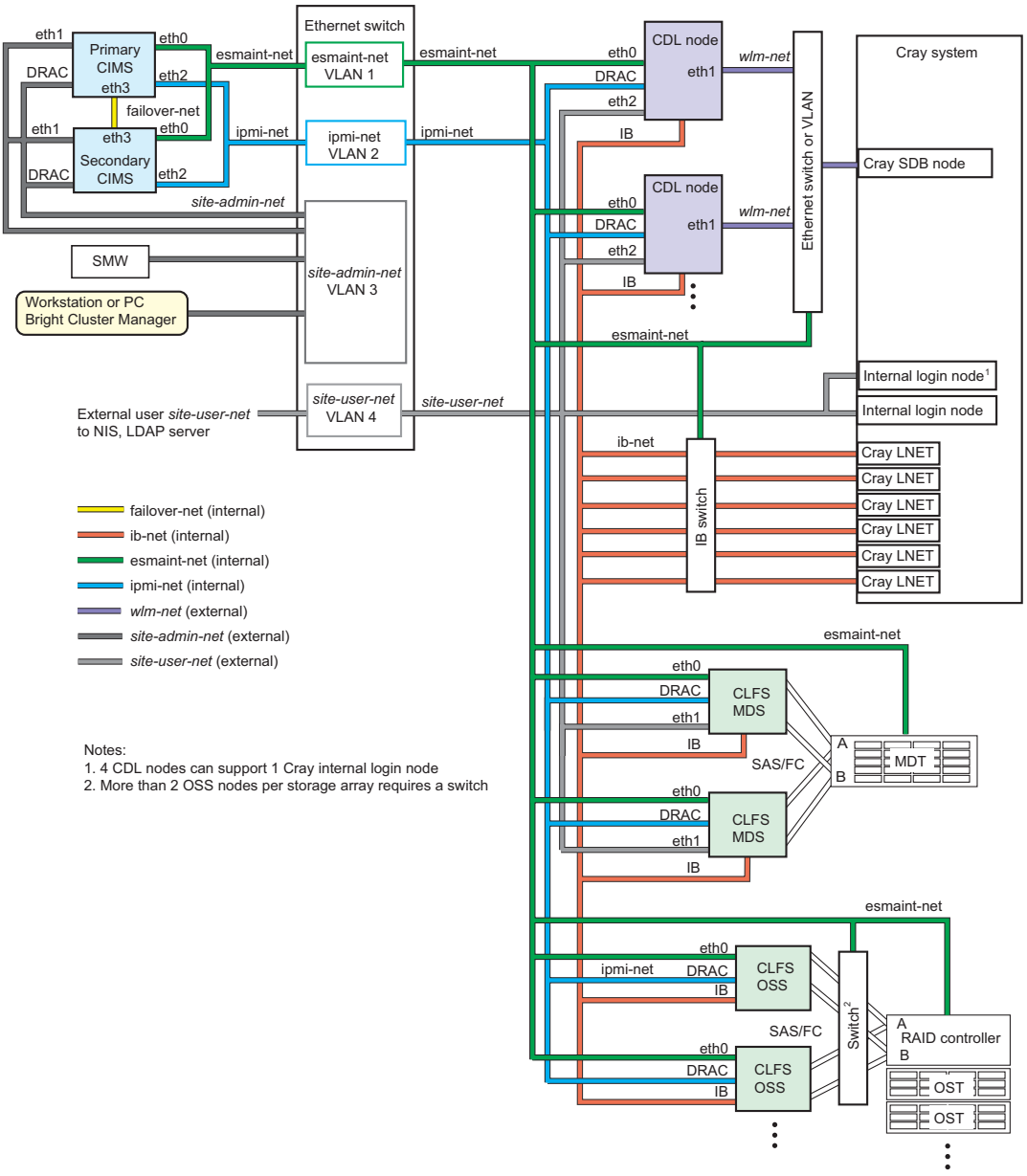
`ib-net` Internal InfiniBand® network used by the slave nodes for Lustre LNET traffic.

`failover-net`

Internal failover network used between two CIMS servers in an HA configuration for heartbeats between the active/passive CIMS nodes. This network does not connect to a managed switch.

Depending on system configuration, additional networks may be required.

Figure 2. DMP Hardware and Networks Overview



1.6.1 CIMS Network Configuration

The CIMS node requires the network interfaces shown in Figure 3 and listed in Table 2. Depending on the system-specific network configuration, additional interfaces may be required. The `esmaint-net` (`eth0` interface) is a private network that connects all the managed devices in a DMP system and is defined as 10.141.*x.x*. The primary function of `esmaint-net` is to enable node provisioning and management. It may be helpful to follow the IP addressing scheme shown in Figure 3 to managed the various devices on the 10.141 network. The `site-admin-net` (`eth1` interface) is the site administration network. The `ipmi-net` (`eth2` interface) is a private

network enables power control and remote console for all of the slave nodes in the system. The CIMS IPMI interface (DRAC) may also be on this network to provide remote console and power management of the CIMS node. The `failover-net` (`eth3` interface) is a private network used to direct heartbeats between CIMS nodes in an HA configuration.

Figure 3. CIMS Network Interfaces and Default Addresses

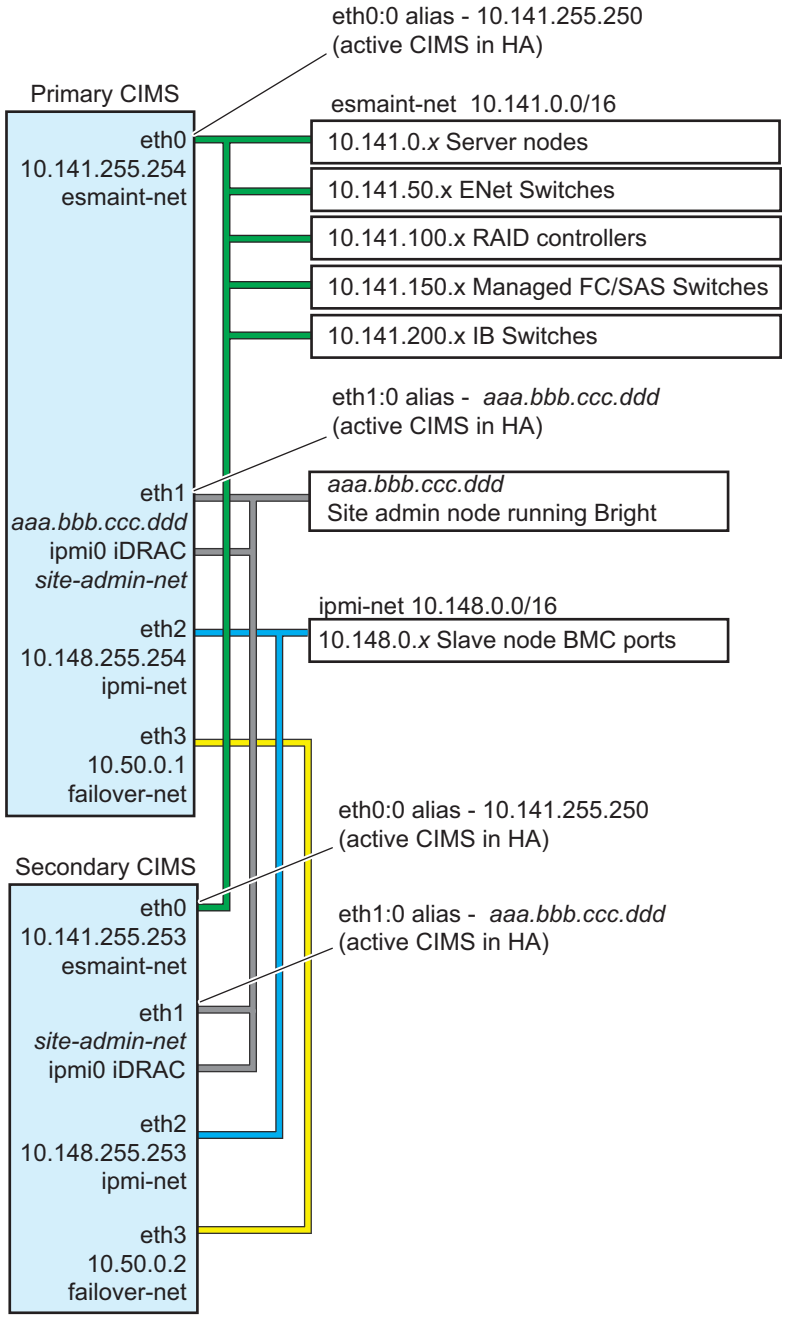


Table 2. CIMS Network Interfaces

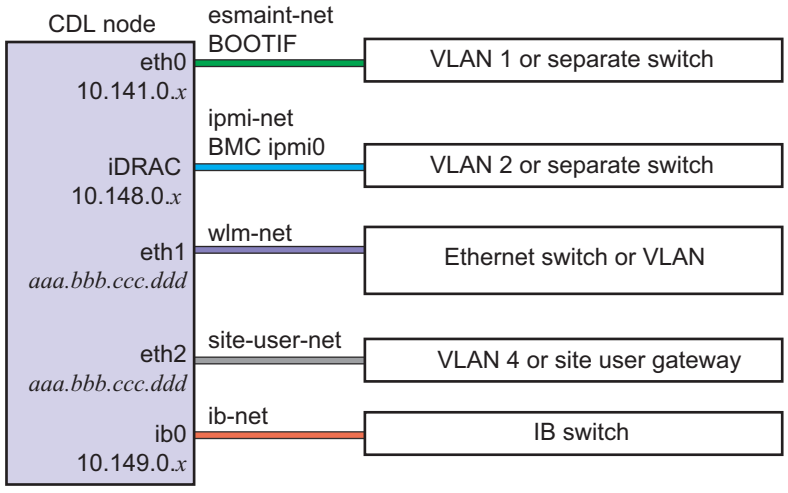
Interface	Network	Description
`eth0`	`esmaint-net`	Interface to the internal management network for maintenance and provisioning. The IP address for this interface is set to 10.141.0.0/16. Other interfaces on the CIMS have the following IP addresses: 10.141.0.*x*/16 Server node addresses. 10.141.50.*x*/16 Managed Ethernet switch addresses. The switch management IP assignments start at 10.141.50.1. 10.141.100.*x*/16 Storage array controller addresses. 10.141.150.*x*/16 Fibre Channel (FC) or serial attached SCSI (SAS) switches. 10.141.200.*x*/16 InfiniBand® (IB) switches. On a system with two CIMS nodes in an HA configuration, the following IP addresses are used for `eth0`: 10.141.255.254 Primary CIMS 10.141.255.253 Secondary CIMS 10.141.255.250 `eth0:0` on both CIMS servers in HA configuration.
`eth1`	`site-admin-net`	Interface to the administration network for the CIMS node (typically on the same network that the SMW is on). In an HA configuration, the alias `eth1:0` is configured to connect to the active CIMS node.
`eth2`	`ipmi-net`	Interface to the remote console and power management network. Its IP address is set to 10.148.0.0/16.

Interface	Network	Description
		10.148.255.254 Primary CIMS 10.148.255.253 Secondary CIMS
`eth3`	`failover-net`	Interface to the internal failover network for CIMS nodes in an HA configuration. The following IP addresses are assigned: 10.50.0.1 Primary CIMS 10.50.0.2 Secondary CIMS
`ipmi0` (DRAC port)	`site-admin-net`	Remote console and power management of the CIMS (typically on the same network that the SMW is on).

1.6.2 CDL Network Configuration

`BOOTIF` is a special name for the `eth0` interface. The node installer automatically translates `BOOTIF` into the name of the device (such as `eth0`), used for network booting. There can be only one interface configured as `BOOTIF`.

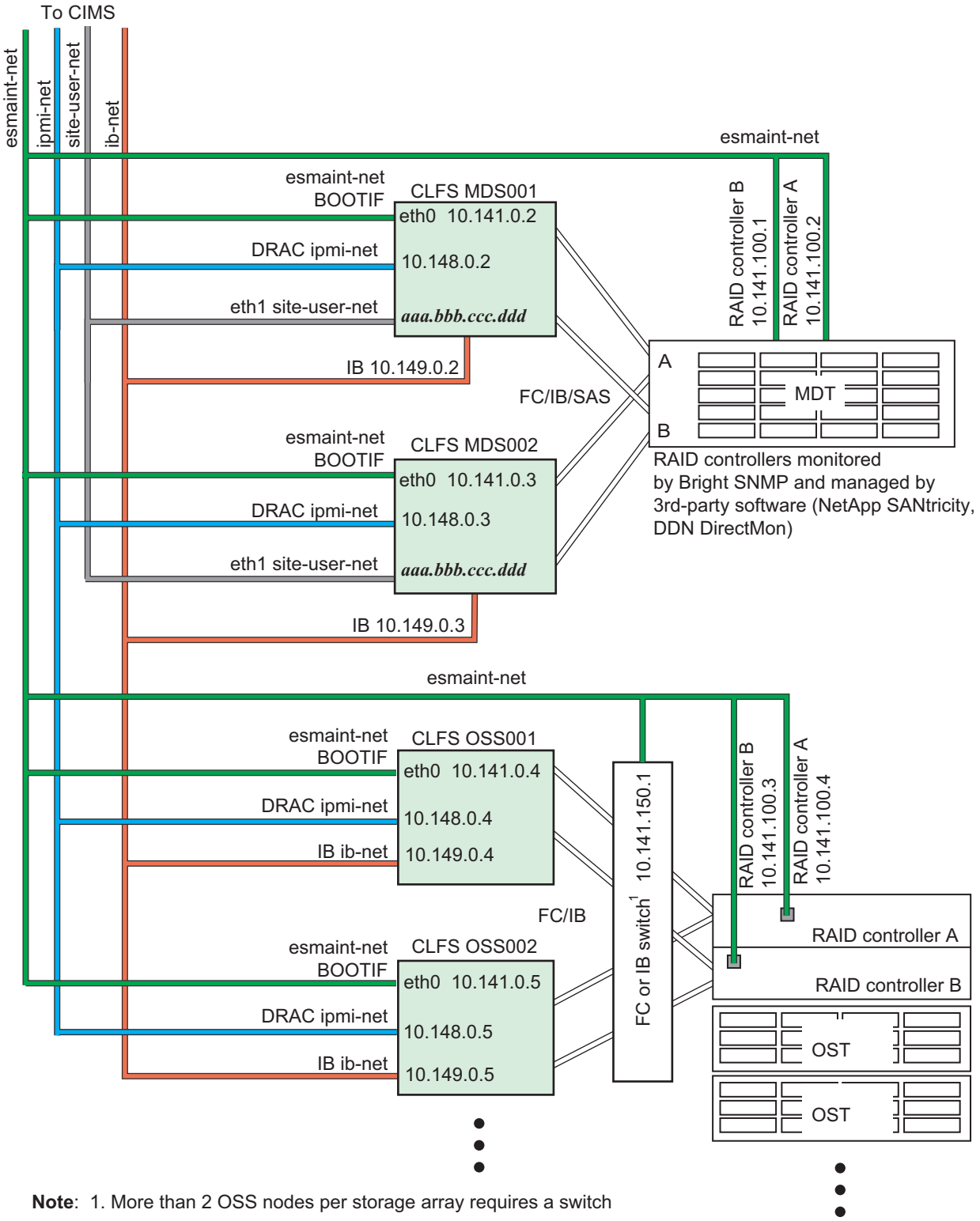
Figure 4. CDL Network Interfaces and Default Addresses



1.6.3 CLFS Network Configuration

`BOOTIF` is a special name for the `eth0` interface. The node installer automatically translates `BOOTIF` into the name of the device (such as `eth0`), used for network booting. There can be only one interface configured as `BOOTIF`.

Figure 5. CLFS Network Interfaces and Default Addresses



Note: 1. More than 2 OSS nodes per storage array requires a switch

1.7 Hardware Components

A DMP system is comprised of specialized service nodes (see Figure 2), network switches, and storage arrays. These devices include, but are not limited to:

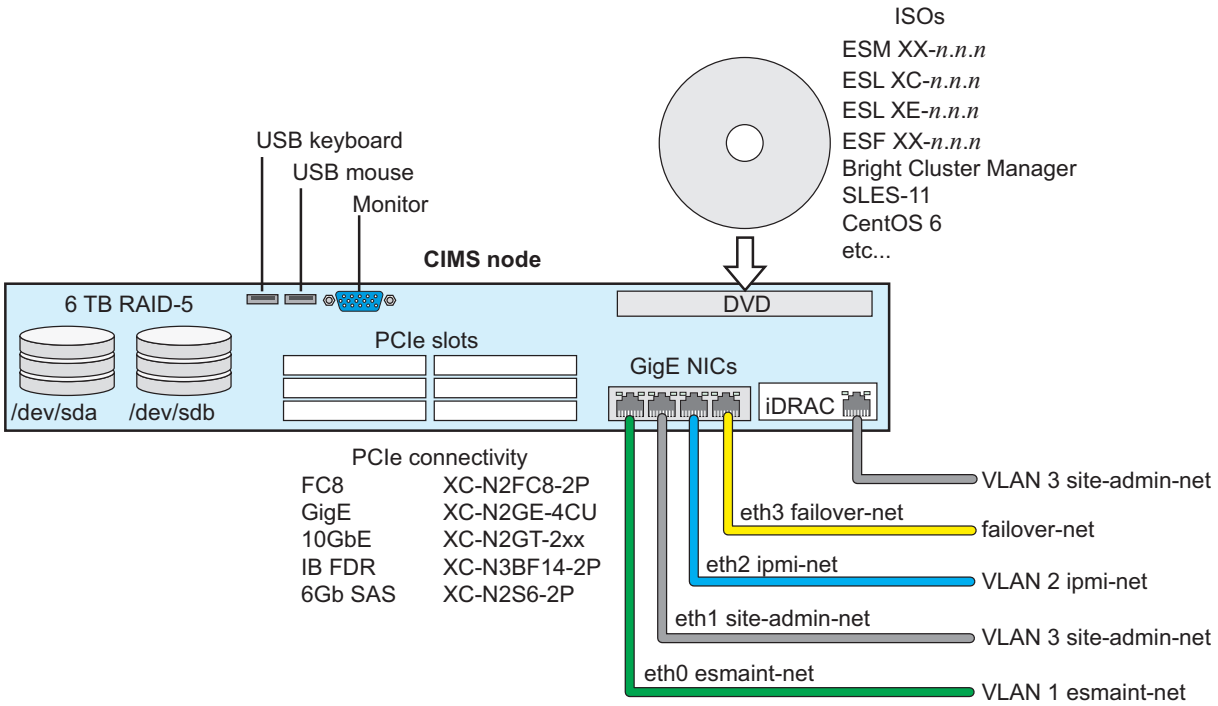
- 1U or 2U rack-mounted servers configured with the necessary hardware and software to perform a specific role in the system, such as a software development platform or file server node
- Ethernet switches to provide connectivity, maintenance access, and zones (VLANs) for each of the networks in the system
- InfiniBand® (IB) switches for high-speed network connectivity
- Fibre Channel (FC) or serial-attached SCSI (SAS) switches typically used for storage networks
- Lustre-based storage arrays
- Uninterruptible power sources or other power management equipment

1.7.1 Cray Management Server (CIMS) Hardware Overview

The CIMS is a 1U or 2U rack mounted server that runs the Bright software and provides a centralized platform to manage the system hardware and software. Nodes that are managed by the CIMS are called *slave nodes*. Figure 6 shows a generic representation of a 2U CIMS node. A DVD drive provides a method for installing software from DVD release media. All of the Cray Data Management Platform (DMP) system management software, slave node software, and software updates and upgrades are installed from the CIMS DVD drive.

A CIMS node is typically configured with six 1-TB disks, configured as two RAID-5 virtual disks (`/dev/sda` and `/dev/sdb`). PCIe slots can be used to add IB, FC, SAS, or GigE connectivity to the CIMS. All slaves nodes in a DMP system are managed, monitored, and provisioned by the CIMS over the `esmaint-net` network. An Intelligent Platform Management Interface (IPMI) network (`ipmi-net`) is used to control power and monitor hardware using simple network monitoring protocol (SNMP). Figure 2 shows how the CIMS is connected to other nodes in the system either through a GigE switch divided into VLANs, or via separate GigE switches.

Figure 6. CIMS Node Hardware Overview

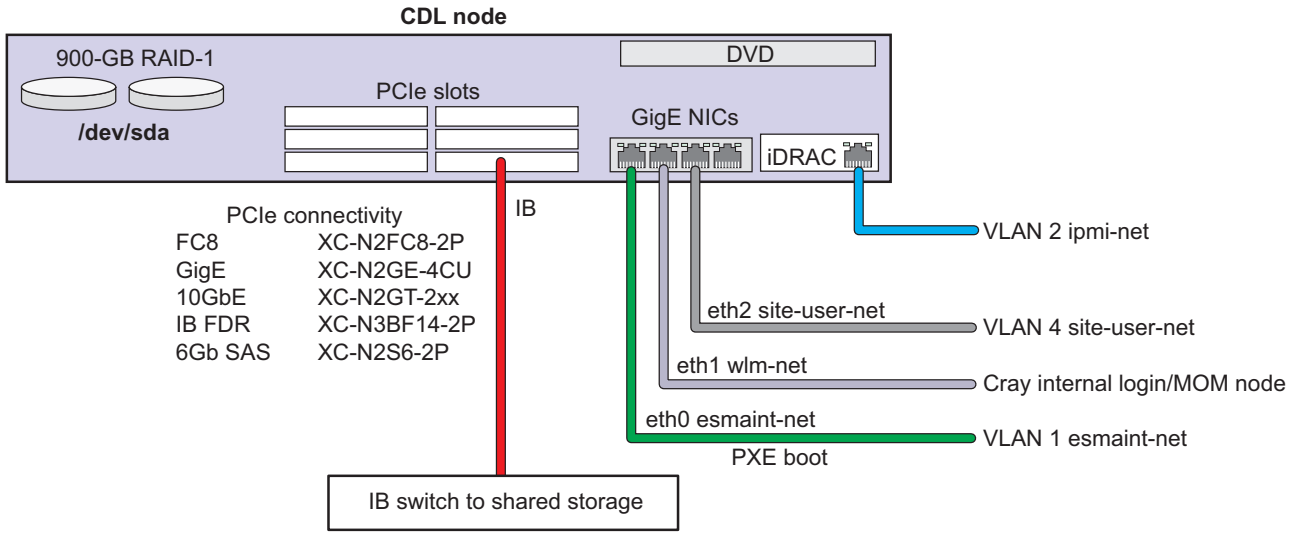


The CIMS disk partitioning is configured differently for a stand-alone CIMS or a high-availability (HA) CIMS during the initial installation procedure of the Bright and ESM software. The HA CIMS configurations make use of Distributed Replicated Block Device (DRBD) shared storage.

1.7.2 Cray Development and Login (CDL) Node Hardware Overview

User development and login (CDL) nodes provide the same programming environment as an internal login node on a Cray system. Each CDL node operates independently of the Cray system and are capable of accessing the same shared file system. The `site-user-network` provides user authentication and user access. The `esmaint-net` network enables the CDL node to use the preboot execution environment (PXE), or PXE boot, and together with the `ipmi-net` network, provides the means to manage and control the node.

Figure 7. CDL Node Hardware Overview



The iDRAC port (`ipmi-net`) enables remote console and power management control from the CIMS node.

There are two 900-GB disk drives in a CDL node that are configured as a RAID1.

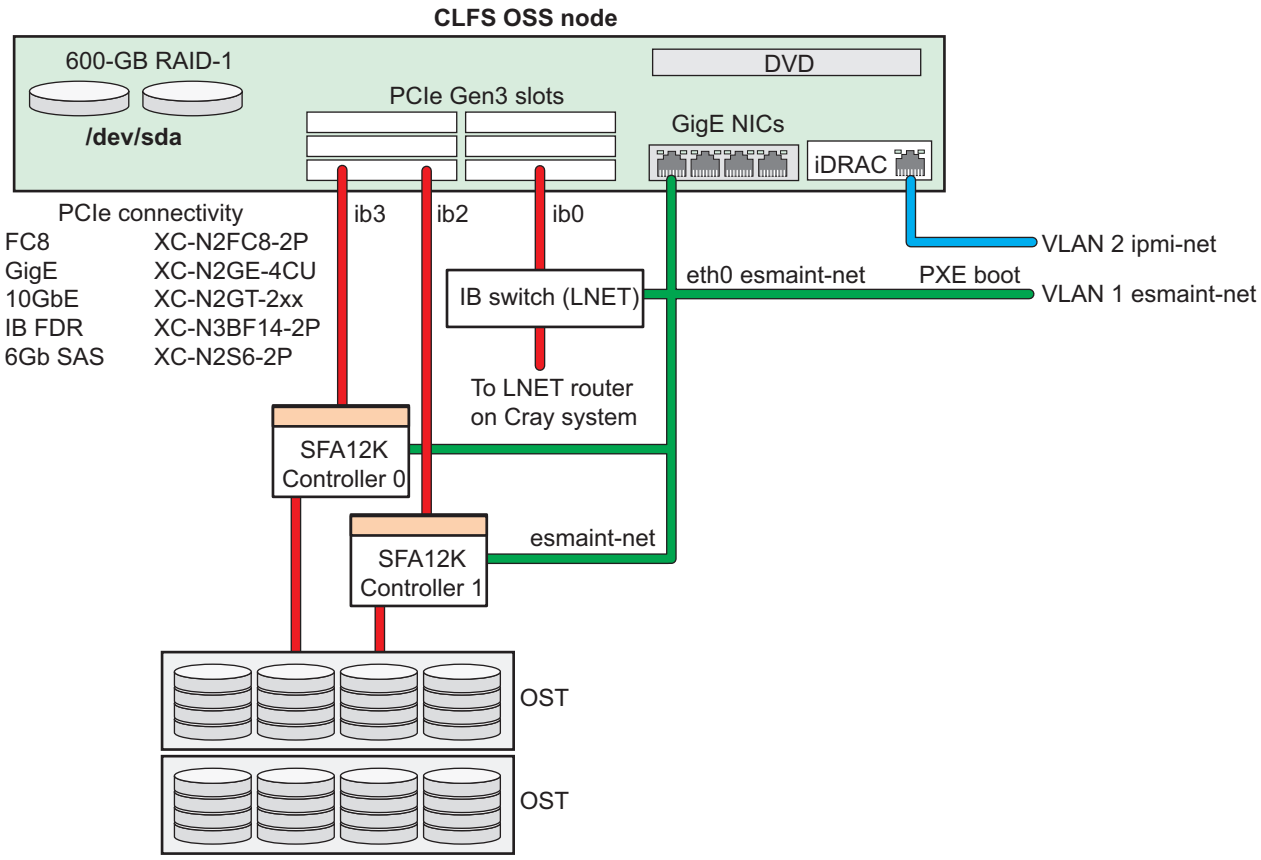
Two file systems `master:/cm/shared` and `master:/home` are NFS® mounted from the CIMS node (`master`). The `master` device can be used as an alias to designate the primary CIMS node.

1.7.3 Lustre® File Server (CLFS) Node Hardware Overview

Lustre file system (CLFS) nodes provide a high-performance Lustre shared file system that operates independently of the Cray system. A DMP file system is comprised of metadata server nodes (MDS), object storage servers (OSS) and object storage target (OST) devices from DataDirect™ Networks (DDN) or NetApp™ within an InfiniBand® interconnect. Each CLFS node file system requires its own set of servers and block storage and are always configured with full redundancy in controllers, servers, and cabling for high-availability.

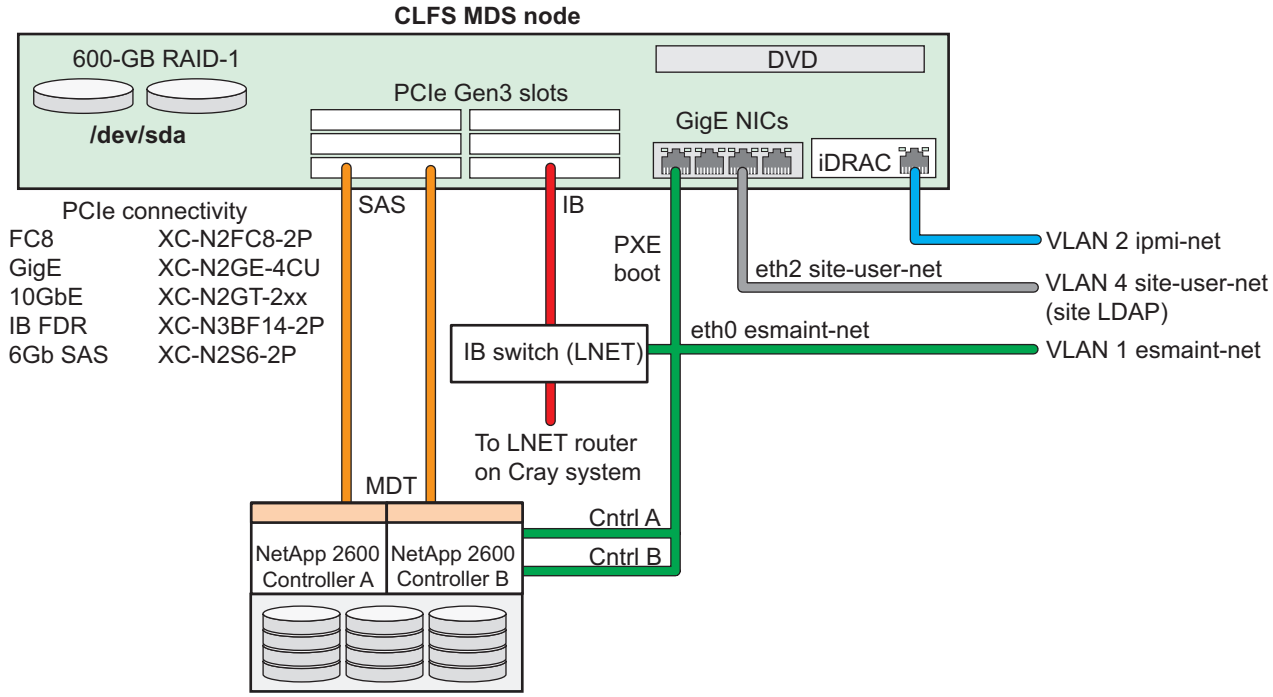
The CLFS OSS servers are typically connected to the DDN or NetApp block storage controllers.

Figure 8. CLFS OSS Node Hardware Overview



CLFS MDS servers are typically connected to the NetApp block storage controllers via InfiniBand® (IB) and are configured with a Mellanox dual-port InfiniBand (IB) host bus adapters (HBAs) installed in the CLFS node's PCIe GEN3 slot. CLFS MDS servers also connect to Cray XC30 systems through fourteen data rate (FDR) IB connection, and thus, include a Mellanox dual-port FDR IB HCA installed in a PCIe GEN3 slot.

Figure 9. CLFS MDS Hardware Overview



1.7.4 Switches and PDUs

Ethernet, InfiniBand, and Fibre Channel switches communicate with Bright using Simple Network Management Protocol (SNMP) using the device management port, which typically an Ethernet connection to `esmaint-net`.

Other devices such as power distribution units (PDUs) can also be configured and managed by Bright. The SNMP community strings should be configured to public read, and private write access. Other device configuration settings such as administrative password, hostname, and IP address should be configured using the device console configuration commands.

Uplink ports (switch ports that are connected to other switches) must be configured in Bright. The CMDaemon (`cmd`) must be told about any switch ports that are uplink ports, or the traffic passing through an uplink port will lead to mistakes in what `cmd` knows about port and MAC correspondence. Uplink ports are thus ports that `cmd` is told to ignore.

Bright Cluster Manager® [2]

2.1 Manage a System with Bright

This chapter provides and overview of how Cray implements Bright to manage a DMP system.

Important: Do not delete or modify the default objects in Bright. These objects are cloned to make other customized objects and must not be deleted or modified.

`node001` The default node (`node001`). The default node is cloned to create other customized DMP nodes.

`esFS-MDS`, `esLogin-XC`, `esLogin-XE`

The default categories that are created when ESF and ESL software is installed. These categories are cloned to create other customized Bright node categories.

`/opt/cray/esms`

Default administrative scripts. The default scripts in this directory are copied from `./default` to `./etc` so that software updates do not overwrite site customizations.

Refer to the *Bright Cluster Manager 6.1 Administrator Manual* for detailed information about Bright software management. PDF files for the Bright manuals are stored on the CIMS in `/cm/shared/docs/cm`, and linked to from the `/root` directory.

2.1.1 Back Up Important Files

Bright includes the capability to make a back up of the system configuration database to an XML file. The Bright database stores all of the configuration and state information for the system. Backup the Bright database before software upgrades and periodically to store system configuration settings. See Back Up System Configuration Settings to an XML File on page 147 for more information.

The `/cm/images` directory on the CIMS node is the default location to store all of the software images that boot slave nodes. Make backup copies of production software images used for slave nodes. These software images must exit in `/cm/images` on the CIMS node3 if the system is being restored from the database XML file.

Site customization files such as `/etc/fstab`, `/etc/hosts`, `LDAP configuration`, etc... should also be backed up regularly and before performing software upgrades.

2.1.2 Bright Interfaces

A CIMS running Bright provides a management interface that is used by:

- Cluster management shell (`cmsh`) — A command line interface to manage and control the system.
- Cluster management user interface (`cmgui`) — A graphical user interface (GUI) to manage and control the system.
- Cluster management daemon (`cmd` command, or CMDaemon) — A process that runs on all nodes in the DMP system. CMDaemon on the CIMS responds to requests from `cmsh` or `cmgui`, and communicates with the CMDaemon processes running on each slave node. The CMdaemon processes on each slave node communicate only with the CMDaemon processes running on the CIMS.

Either the `cmgui` or `cmsh` can be used to manage the system and there may be certain tasks are more easily visualized using `cmgui`, and other tasks are more efficient using the `cmsh`. The following procedures use the Bright management shell (`cmsh`). They may also be performed using the Bright GUI (`cmgui`). The `cmsh` command prompt displays an asterisk (`*`) when changes have not been committed. Be sure to commit your changes using the `commit` command before exiting `cmsh`, or your changes will be lost. Alternatively, the `cmgui` enables the **Save** button.

General information about how to use the `cmgui` are described in The Bright GUI on page 56.

2.1.3 Device Names in Bright

A *device* in a DMP system represents a physical hardware component. A device can be any of the following types:

- Head nodes — Typically named `esms1`, `esms2`, `tas-cims`.
- Ethernet switches — Typically named according to their function in the cluster such as `switch-esfs1`, `switch-esmaint-net`, `switch-eslogin`.
- InfiniBand® switches — Typically named `switch-ib1-scratch`, `tas-ibsw1`.
- Fibre Channel switches — Typically named `fc-switch1`, `tas-fcsw2`.
- Slave nodes — Typically named according to their function in the cluster, such as `cdl-001` (login node), `esfs-mds001` (metadata controller), `esfs-oss001` (object storage server).

Important: The ESM- XX-3.0.0 release provides a single finalize script, `esf_finalize.sh`, that is customized to configure both MDS and OSS nodes. The node names (`$HOSTNAME`) **must** contain the string `mds` or `oss` string so that a customized `site.esf_finalize.sh` script can configure both node types. CLFS node names **must** contain the string `mds` or `oss` and be assigned to the `esfs-odd-`*filesystem* or `esfs-even-`*filesystem* categories to support esfsmon 2.0.0.

- Storage array RAID controllers — Storage controllers are added as a generic device in `cmsh` or under the **Other** resource in the `cmgui`. Typically named by manufacturer model number, rack location or purpose, `netapp5500-cntlA`, `netapp3992-cntlA`, `rack1-ddnsfa12k-cntrl0`, `netapp2700-cntlA`, `netapp2700-cntlB`.

2.1.4 Node Organization

Cray DMP systems along with Bright software support the concept of *node categories* and *node groups*.

Categories specify a number of parameters that are common to all members. Among these are the management network, as well as the software image and scripts that are run by the node-installer to customize each node's image during provisioning. A slave node is associated with one node category. Category parameters can be overridden on a per node basis, if desired, by configuring the node, instead of the node category. Slave nodes can belong to several different node groups, and there are no parameters associated with node groups. Node groups are typically used to invoke commands across several nodes simultaneously.

The Bright software configures a separate interface for each node because the IP addresses that Bright uses are specific to each node. Software images are common across multiple nodes, so the Bright interface files must reside in the Bright database and be placed on other nodes at boot time.

The node category defines which software image is provisioned to its member nodes and other management attributes.

Table 3 lists Cray DMP node categories.

Table 3. DMP Slave Node Categories

Category	Description
`esLogin-XE`	Default category configured by ESL installation software for Cray XE platform CDL nodes.
`esLogin-XC`	Default category configured by ESL installation software for Cray XC30 platform CDL nodes.

Category	Description
`esFS-MDS`	Default category configured by ESF installation software CLFS nodes.
`esFS-OSS`	Default category configured by ESF installation software CLFS nodes.
`esfs-odd-`*filesystem*	Used by esfsmon 2.0.0 to configure an odd numbered CLFS node for the file system named by *filesystem*.
`esfs-even-`*filesystem*	Used by esfsmon 2.0.0 to configure even-numbered CLFS node for the file system named by *filesystem*.
`esfs-failed-` *filesystem*	Used by esfsmon 2.0.0 to configure failed CLFS node for the file system named by *filesystem*.

Most importantly, the node category determines which software image a node runs. Node categories also provide control over several other parameters such as:

`revision` Object revision.

`bmcpassword`

Password used to send `ipmi`/`ilo` commands to nodes. The baseboard management controller (BMC or iDRAC) password is inherited from the `base` partition and typically not set for the node category.

`bmcusername`

User name used to send `ipmi`/`ilo` commands to nodes. Inherited from the `base` partition, and typically not set for the category.

`defaultgateway`

Default gateway for the category.

`filesystemexports`

Configure the entries placed in `/etc/exports`.

`filesystemmounts`

Configure the entries placed in `/etc/fstab`.

`installbootrecord`

Install boot record on slave node local disk to enable booting without a CIMS node.

`installmode`

`installmode` to be used by default, if none is specified in the node.

`ipmipowerresetdelay`

Delay used for `ipmi`/`ilo` power reset, default is `0`.

`managementnetwork`

Determines network used for management traffic. If not set, `partition` mode setting is used.

`name` Name of category.

`nameservers`

List of name servers the category uses.

`newnodeinstallmode`

Default install mode for new nodes.

`roles` Assign the roles the node should play.

`searchdomain`

Search domains for the category.

`services` Manage operating system services.

`softwareimage`

Software image the category uses.

`timeservers`

List of time servers the category uses.

`usernodelogin`

Set to always or never to control user log in to the node.

`disksetup` Disk setup for nodes.

`excludelistfullinstall`

Exclude list for full install. See Set Up Exclude Lists on page 115.

`excludelistgrab`

Exclude list for grabbing the image running on the node to an existing image.

`excludelistgrabnew`

Exclude list for grabbing to a new image.

`excludelistsyncinstall`

Exclude list for a sync install. Specifies what files and directories to exclude from consideration when copying parts of the file system from a known good software image to the node.

`excludelistupdate`

Exclude list for updating a running node.

`finalizescript`

Finalize script to be used for category.

`initializescript`

Initialize script to be used for category.

`notes` Administrator notes.

Node groups simplify management and control activities and enable administrators to perform commands on a group of nodes simultaneously. Typical node groups are listed below:

Table 4. Node Groups

Node Group	Description
`login`	All CDL nodes
`oss`	All OSS nodes
`esfs-even-`*filesystem*	All even CLFS nodes for *filesystem*

2.1.5 Software Image Management

A software image is a blueprint for the contents of the local file systems on slave nodes. Software images reside in the `/cm/images` directory on the CIMS and contain a full Linux™ file system and other customizations. When a slave node boots, the node provisioning system configures the node with a copy of its assigned software image determined by the node category. After the node boots, it is possible to instruct the node to resynchronize its local file systems with the software image. This procedure can be used to distribute changes to the software image without rebooting nodes. It is also possible to lock a software image so that no node is able to pick up the image until the software image is unlocked. Software images can be changed using Linux tools and commands such as `rpm` and `chroot`.

Important: The software images in `/cm/images` should be managed carefully and backed-up regularly, as they are needed to reinstall the CIMS software and reload the system configuration if needed. Refer to Back Up System Configuration Settings to an XML File on page 147 for more information about saving the system configuration.

Software images are typically prefixed with ESM, ESL, or ESF and include the release version and date. Use `cmsh softwareimage` mode, and enter `list` to display a list of images on the CIMS. Other image properties such as `notes` can help administrators manage software images.

Example 1. List software images

```
esms1# cmsh
[esms1]% softwareimage
[esms1->softwareimage]% list
Name (key)                Path                                    Kernel version
------------------------- --------------------------------------- --------------------------
ESF-XX-2.2.0-201401151643  /cm/images/ESF-XX-2.2.0-201401151643     2.6.32-279.14.1.el6.x86_64
ESF-XX-2.0.0-201304181540  /cm/images/ESF-XX-2.0.0-201304181530     2.6.32-279.14.1.el6.x86_64
ESL-XC-2.2.0-201401160637  /cm/images/ESL-XC-2.2.0-201401160637     3.0.80-0.5-default
ESL-XE-2.2.0-201401090705  /cm/images/ESL-XE-2.2.0-201401090705     2.6.32.59-0.7-default
ESL-XE-1.1.1-kdump         /cm/images/ESL-XE-1.1.1-kdump           2.6.32.59-0.7-default
ESL-XE-1.1.1_CLE4.1        /cm/images/ESL-XE-1.1.1_CLE4.1          2.6.32.59-0.7-default
default-image              /cm/images/default-image                3.0.80-0.5-default
default-image.previous     /cm/images/default-image.previous
```

Note that `default-image` and `default-image.previous` are images created by the Cray installer software, `ESMinstall`. The `ESLinstall` and `ESFinstall` installers configure software images for Cray Development and Login (CDL) and Lustre File System by Cray (CLFS) nodes. The Cray installer software adds the latest released software and updates to the `/cm/images` directory on the CIMS when the installation completes, and also configures the image in the Bright database. Installed images use the naming conventions in Distribution Media on page 23.

Important: The default image (`default-image`) should never be modified or deleted. Always clone `default-image` or a production image to create a new software image when making modifications. Always keep a functioning image as a backup. All software images in `/cm/images` on the CIMS should be managed carefully and backed up regularly.

When a node boots, the node provisioning system sets up the node with a copy of the software image that is configured for the node. Software images are assigned to a `category` in Bright. Nodes are also assigned to a `category` in Bright. This enables the administrator more control over what software image is used by a node.

After an image is installed and configured, use the Bright `clone` command to create a copy of the image for testing.

2.1.6 Node Provisioning

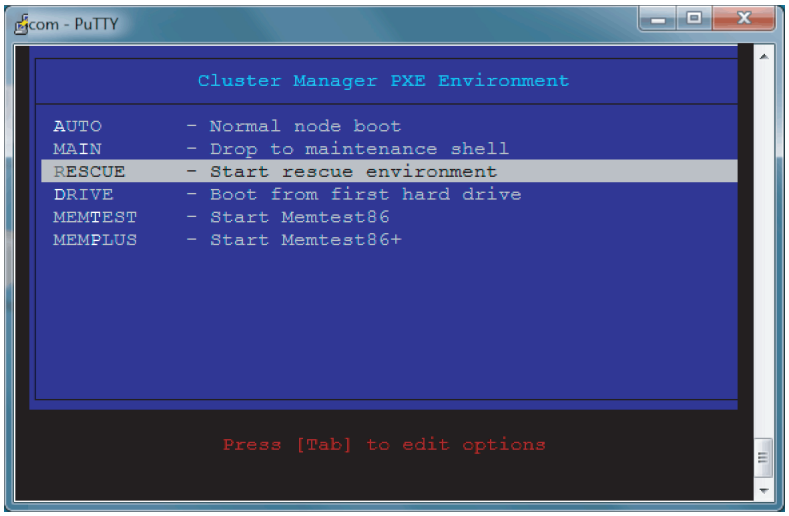
Node provisioning is the process of how nodes obtain a software image and occurs during power-up or when updating a running node.

2.1.6.1 Preboot Execution Environment (PXE) Booting

By default, slave nodes boot from the `esmaint-net` network over the `BOOTIF` interface. Bright controls this network boot or Preboot Execution Environment (PXE) boot. PXE booting is configured in the slave node BIOS settings. The CIMS runs a `tftpd` server process from within `xinetd`, which supplies the boot loader from within the default software image offered to nodes.

The boot loader runs on the node and displays a menu based on loading a menu module within a configuration file. The configuration file is located in the `default-image` software image in the `/cm/images/default-image/boot/pxelinux.cfg/` directory, `default` file.

Figure 10. PXE Boot Menu



The `MENU DEFAULT` value in the software image `/cm/images/default-image/boot/pxelinux.cfg/default` file is loaded for every node using the software image. To override its application on a per-node basis, the value of `PXE Label` can be set for each node. For example, to set a node to use the `MEMTEST` PXE label using `cmsh`:

Example 2. Set PXE label for a node

```
esms1# cmsh
[esms1]% device use eslogin001
[esms1->device[eslogin001]]% set pxelabel MEMTEST ; commit
```

To use the MEMTEST label for all nodes the esLogin-XE category use:

```
[esms1->device]% foreach -c esLogin-XE (set pxelabel MEMTEST)
[esms1->device*]% commit
```

Refer to the *Bright Cluster Manager 6.1 Administrator Manual* PDF file on the CIMS node for more information.

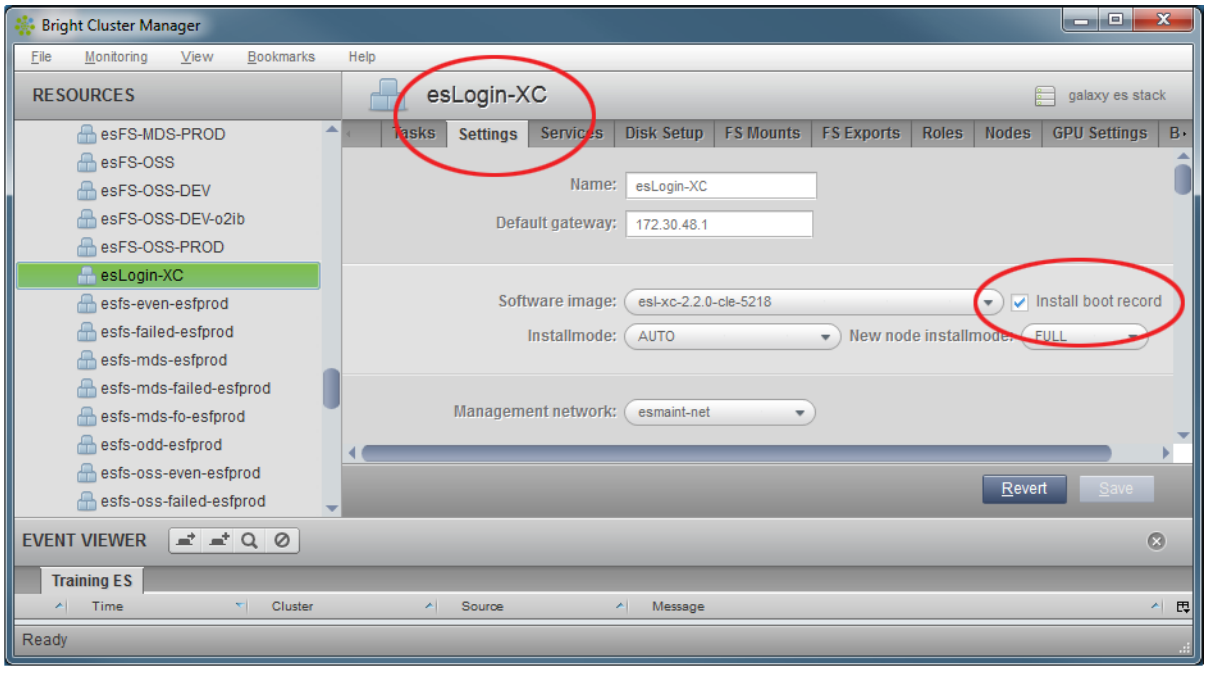
2.1.6.2 Install Slave Node Software Image Boot Record

Each slave node image configured in Bright should be set so the node-installer installs a boot record on the local drive. Set the `installbootrecord` property for the node settings and node category to `on` so that the node can boot from the local drive. Hard drive booting must be set to a higher priority than network booting in the BIOS settings for the node. Otherwise PXE booting occurs, despite the value set for `installbootrecord`.

Set the GRUB bootloader `Install boot record` checkbox in `cmgui` and save the setting in the node configuration or in the node category as shown in the figure.

Unplug the cable for the `esmaint-net` network (the provisioning network or BOOTIF network) on the slave node and reboot the node to determine if the node can boot without a CIMS node.

Figure 11. Install Boot Record



The `cmsh` command to enable `installbootrecord` for the node is:

```
esms1 # cmsh -c "device use node001; set
installbootrecord yes; commit"
```

The `cmsh` command to enable `installbootrecord` for a category is:

```
esms1 # cmsh -c "category use esLogin-XC; set
installbootrecord yes; commit"
```

2.1.6.3 The Boot Role

The action of providing a PXE boot image via DHCP and TFTP is known as providing *node booting*.

2.2 Bright License Management

Cray DMP systems are configured with a Bright Cluster Manager license file installed on the CIMS node. The license file includes:

- A `licensee` attribute or the name of the organization, the condition in which the specified organization may use the software; a licensed nodes attribute specifies the maximum number of nodes that the cluster manager may manage. CIMS nodes are also regarded as nodes too for this attribute.
- `licensed nodes` attribute specifies the maximum number of nodes that the cluster manager may manage. CIMS nodes are included.
- An `expiration date` for the license.

A license file can only be used on the machine for which it has been generated and cannot be changed once it has been issued. The license file is the X509v3 certificate for the CIMS node and is used throughout system operations.

2.2.1 Display License Attributes

The license file is installed in the following location on the CIMS:

`/cm/local/apps/cmd/etc/cert.pem`

The associated private key file is in:

`/cm/local/apps/cmd/etc/cert.key`

To verify that the attributes of the license have been assigned the correct values, use the GUI to select the CIMS node, **License** tab, to display license details.

Alternatively the `cmsh main` mode `licenseinfo` command displays:

Example 3. Display license attributes in cmsh

```
esms1:~ # cmsh
[esms1]% main licenseinfo
License Information
---------------------------- ------------------------------------------
Licensee                     /C=US/ST=Wisconsin/L=Chippewa Falls/O=Cray
                              Training/OU=Training and Doc/CN=Training
Serial Number                5510
Start Time                   Tue May  1 00:00:00 2012
End Time                     Fri Feb  8 23:59:00 2013
Version                      6.0
Edition                      Advanced
Pre-paid Nodes               512
Max Pay-as-you-go Nodes      1000
Node Count                   6
MAC Address / Cloud ID       78:2B:CB:40:CE:CA
```

The license in the example above enables 512 nodes. It is tied to a specific MAC address, and the `Node Count` field in the output of `licenseinfo` shows the current number of nodes used.

2.2.2 Verify a License

The `verify-license` utility determines whether a license is valid even when the cluster management daemon is not running.

Example 4. Verify a license

```
esms1# /etc/init.d/cmd start
Waiting for CMDaemon to start...
CMDaemon failed to start please see log file.
esms1# tail -1 /var/log/cmdaemon
Dec 30 15:57:02 esms-1 CMDaemon: Fatal: License has expired
```

2.2.3 Install the Bright License

The initial installation of Bright is licensed only for three nodes, including the CIMS. A permanent license must be installed before configuring the system.

- Use Procedure 1 on page 52 to install the license.
- Contact a Cray representative to purchase another node license for the second CIMS if needed for an HA configuration.
- If your existing license has expired and must reinstate your license.

Certificate Sign Request (CSR) data is displayed and saved in the file `/cm/local/apps/cmd/etc/cert.csr.new`. The `cert.csr.new` file may be used with an internet-connected browser.

After using a product key with `request-license`, reboot the system using the **`pexec reboot`** command from the CIMS. A reboot is not required when relicensing an existing system.

Important: If licensing a secondary CIMS for an HA configuration, get the MAC address for `eth0` of the secondary node. The MAC address for the secondary node is typically located on a label on the front panel of the node.

Procedure 1. Install the Bright license on a CIMS

Obtain a product key from Cray which enables the administrator to obtain and activate a license. The product key looks like the following string:

XXXXXX-XXXXXX-XXXXXX-XXXXXX-XXXXXX

1. Log in to the CIMS as `root`.

```
remote% ssh root@esms1
```

2. Get the MAC address (`HWaddr`) for `eth0` (`BOOTIF`) interface.

```
esms1# /sbin/ifconfig eth0
eth0      Link encap:Ethernet  HWaddr 78:2B:CB:40:CE:CA
          inet addr:10.141.255.254  Bcast:10.141.255.255  Mask:255.255.0.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:11944661 errors:0 dropped:0 overruns:0 frame:0
          TX packets:11018308 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:2589377379 (2469.4 Mb)  TX bytes:2060383201 (1964.9 Mb)
          Interrupt:36 Memory:d6000000-d6012800
```

3. Run the `request-license` command on the CIMS node. A prompt to reuse the private key and settings from the existing license displays if the existing license is valid.

   Important: When configuring an HA system, enter license information for the primary CIMS only. Do not enter the MAC address for the secondary CIMS. Run this procedure again when configuring the secondary CIMS and enter **`yes`** in step 4.c.

```
esms1# request-license
Product Key (XXXXXX-XXXXXX-XXXXXX-XXXXXX-XXXXXX):ProductKey
```

4. Enter the product key, then press `Enter`.

```
Re-use private key and settings from existing license? [Y/n] yes
```

   - If licensing a stand-alone CIMS node or primary CIMS node for an HA configuration, enter **`no`** at the prompt.
   - If configuring a secondary CIMS node an HA system, a prompt displays that asks to reuse the private key settings from the existing license. Enter **`yes`** at the prompt and then proceed to step 4.b.

a. Enter the site information for the Bright license when prompted:

```
Country Name (2 letter code): CountryName
State or Province Name (full name): StateProvince
Locality Name (e.g. city): City
Organization Name (e.g. company): Company
Organizational Unit Name (e.g. department): Department
Cluster Name: ClusterName
```

b. Enter the MAC address of the primary CIMS for `eth0` on (`esmaint-net`). If you are licensing a secondary CIMS in an HA configuration, enter the MAC address for `eth0` for the secondary CIMS.

```
MAC Address of primary head node (esms1) for eth0  []: FF:AE:FF:B5:E2:64
MAC Address of secondary head node (esms2) for eth0 []: FF:AE:FF:B5:E2:65
```

If configuring a stand-alone CIMS node, enter `no` at the following prompt. If configuring the primary CIMS node in an HA configuration, enter `yes` at the following prompt. If the primary HA CIMS node is configured, and are configuring the secondary HA CIMS node, enter `yes` and provide the required information to obtain an HA license certificate.

```
Will this cluster use a high-availability setup with 2 head nodes? [y/N]: no
```

c. Answer **no** when you are prompted to submit the certificate request:

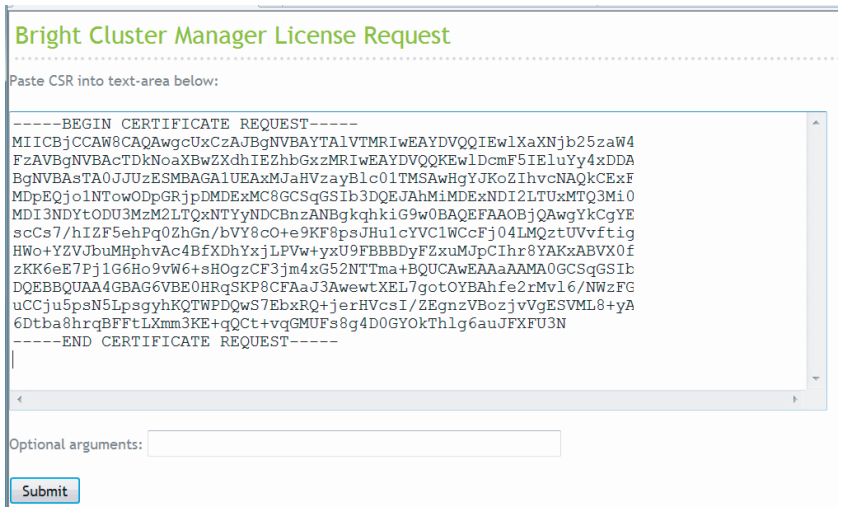
```
Certificate request data saved to /cm/local/apps/cmd/etc/cert.csr.new
Submit certificate request to http://support.brightcomputing.com/licensing/index.cgi
? [Y/n] no
```

The CSR (Certificate Sign Request) data is displayed and saved in the file `/cm/local/apps/cmd/etc/cert.csr.new` on the CIMS. The `cert.csr.new` file may be used to obtain a license with an Internet-connected browser: The license strings used below are fictitious.

```
-----BEGIN CERTIFICATE REQUEST-----
MIICBjCCAW8CAQAwgcUxCzAJBgNVBAYTAlVTMRIwEAYDVQQIEwlXaXNjb25zaW4x
FzAVBgNVBAcTDkNoaXBwZXdhIEZhbGxzMRIwEAYDVQQKEwlDcmF5IEluYy4xDDAK
BgNVBAsTA0JJUzESMBAGA1UEAxMJaHVzayBlc01TMSAwHgYJKoZIhvcNAQkCExFF
MDpEQjo1NTowODpGRjpDMDExMC8GCSqGSIb3DQEJBhMiMDExNDI2LTUxMTQ3Mi0w
MDI3NDYtODU3MzM2LTQxNTYyNDCBnzANBgkqhkiG9w0BAQEFAAOBjQAwgYkCgYEA
scCs7/hIZF5ehPq0ZhGn/bVY8cO+e9KF8psJHu1cYVC1WCcFj04LMQztUVvftigI
HWo+YZVJbuMHphvAc4BfXDhYxjLPVw+yxU9FBBBDyFZxuMJpCIhr8YAKxABVX0fS
zKK6eE7Pj1G6Ho9vW6+sHOgzCF3jm4xG52NTTma+BQUCAwEAAaAAMA0GCSqGSIb3
DQEBBQUAA4GBAG6VBE0HRqSKP8CFAaJ3AwewtXEL7gotOYBAhfe2rMvl6/NWzFGD
uCCju5psN5LpsgyhKQTWPDQwS7EbxRQ+jerHVcsI/ZEgnzVBozjvVgESVML8+yA0
6Dtba8hrqBFFtLXmm3KE+qQCt+vqGMUFs8g4D0GYOkThlg6auJFXFU3N
-----END CERTIFICATE REQUEST-----
```

5. On a system with Internet access, use a web browser to open http://support.brightcomputing.com/licensing/.
6. Copy CSR data obtained in step 4.a to obtain a license certificate.
7. Paste the CSR data (contents of `cert.csr.new`) into the web form, then select **Submit**.

Figure 12. Bright License Request Form



A signed license certificate is displayed (the following example is fictitious).

```
-----BEGIN CERTIFICATE-----
MIIDNzCCAh+gAwIBAgICFkwwDQYJKoZIhvcNAQEFBQAwbDELMAkGA1UEBhMCVVM
CzAJBgNVBAgTAkNBMREwDwYDVQQHEwhTYW4gSm9zZTEfMB0GA1UEChMWQnJpZ2h
IENvbXB1dGluZywgSW5jLjEcMBoGA1UEAxMTQnJpZ2h0IENvbXB1dGluZyBDQTA
Fw0xMjA5MTIwMDAwMDBaFw0zODEyMzEyMzU5MDBaMGYxCzAJBgNVBAYTAlVTMRQ
EgYDVQQIEwtNaXNzaXNzaXBwaTESMBAGA1UEBxMJVmlja3NidXJnMREwDwYDVQQ
EwhDcmF5IEluYzENMAsGA1UECxMEc3RjbzELMAkGA1UEAxMCbmEwgZ8wDQYJKoZ
hvcNAQEBBQADgY0AMIGJAoGBAN8lCM52TEnZ63yvxPvpe4WbBTPsFKUWOpIOHT8
tbctYf54E4K4A1A0ahX48OYdhafhTb7AO0gGSv/Vp+QxZQrkrSi6A8zwlNkrz4j
yDDYFNb4sBkJUbexHmEdR5Bhp/xfEx4X7EkFb5vdnhIyiwvn6Zs5tZ/Sgt+CJcH
KJFtAgMBAAGjbTBrMA8GA1UdEwEB/wQFMAMBAf8wHgYHKwYBBKFkIQQTFhE4NDo
Rjo2OTpFMzo1Qjo1ODAPBgcrBgEEoWQiBAQWAjE2MBAGBysGAQShZCMEBRYDNS4
MBUGBysGAQShZCQEChYIQWR2YW5jZWQwDQYJKoZIhvcNAQEFBQADggEBAF1pRtg
vjMr9TlihmEcO23raTLp308zkVFpW7vJ0T8KqEirwkzzrD83igtJNd2q6jtrpRL
3kxQ2sB0gGmuptHkrYecwZtVm5FhGOuPeDUG3ww4W+GyCkczCRtPkVTXul250Z9
LZqrK0zzPRKhMNEraTXPDEgHSEgEeykro30EGHpcuCoGCBJNvhi0bCIxgJoW5DV
ykGbeE4DKBlyW0r4NRqMBR+0BH2d1gCXBJtYhHMfDUWw6JOMmsCcoAY7Yhp4N9k
AJb++bi/pO2fQeDJopfxvlU2WsEEMcEItNklknaHlOYfQ3IcloH9w464MtGFakt
xGjtfqxcU
-----END CERTIFICATE-----
```

8. Copy the license text received and save it to a plain text file named `signedlicensefile` on the CIMS.

9. Enter the following command to install the license and answer each prompt.

```
esms1 # install-license signedlicensefile
========= Certificate Information ========
Version:                 6.0
Edition:                 Advanced
Common name:             na
Organization:            ACME
Organizational unit:     Training
Locality:                City
State:                   State
Country:                 US
Serial:                  3728
Starting date:           12 Sep 2012
Expiration date:         31 Dec 2038
MAC address:             84:8F:E4:E3:5B:64
Pre-paid nodes:          16
Max Pay-as-you-go Nodes: N/A
==========================================

Is the license information correct ? [Y/n] Y
Backup directory of old license:
/var/spool/cmd/backup/certificates/2013-01-28_08.56.22Installed new license

Waiting for CMDaemon to stop: OK
Installing admin certificates

Waiting for CMDaemon to start: OK

New license was installed. In order to allow nodes to obtain a new
node certificate, all nodes must be rebooted.

Please issue the following command to reboot all nodes:
        pexec reboot
```

Protect the `admin.pfx` file in the `/root` directory that contains the administrator certificate. If the license process fails, check `/var/log/cmdaemon` for failure information. Refer to Manage Bright `admin.pfx` Certificates on page 85 for important information about how to manage the Bright `admin.pfx` file holding the administrator certificate. This file is copied to the remote system that runs the `cmgui` and grants access to the system.

```
Please provide a password that will be used to password-protect the PFX file
holding the administrator certificate (/root/.cm/cmgui/admin.pfx).

Password:
Verify password:
```

2.2.4 Reinstate an Expired License

Procedure 2. Reinstating an expired license

1. Log in to the CIMS as `root`.

```
# ssh root@cray-esms1
```

2. Run the `request-license` command on the CIMS.

```
cray-esms1:~ # request-license
Product Key (XXXXXX-XXXXXX-XXXXXX-XXXXXX-XXXXXX):
```

3. Enter the product key, then press `Enter` (the example is not a valid key).

```
714354-916786-132324-207440-186713
```

4. Answer each prompt to reinstall the license.

```
Existing license was found:
  ...
Re-use private key and settings from existing license? [Y/n] y

Will this cluster use a high-availability setup with 2 head nodes? [y/N] n
MAC Address of primary head node for eth0 []: 78:2B:CB:40:CE:CA

Certificate request data saved to /cm/local/apps/cmd/etc/cert.csr.new
Submit certificate request to http://support.brightcomputing.com/licensing/index.cgi ? [Y/n] y

Contacting http://support.brightcomputing.com/licensing/index.cgi...

License granted.
License data was saved to /cm/local/apps/cmd/etc/cert.pem.new
Install license? [Y/n] y
========= Certificate Information ========
Version:                6.0
Edition:                Advanced
Common name:            Training
Organization:           ACME Training
Organizational unit:    Training and Doc
Locality:               Chippewa Falls
State:                  Wisconsin
Country:                US
Serial:                 5846
Starting date:          01 May 2012
Expiration date:        13 Mar 2013
MAC address:            78:2B:CB:40:CE:CA
Pre-paid nodes:         512
Max Pay-as-you-go Nodes: 1000
==========================================

Is the license information correct ? [Y/n] y
 directory of old license: /var/spool/cmd/backup/certificates/2013-02-10_11.34.53
Is this host the cluster's head node? [Y/n] y
Installed new license

Restarting Cluster Manager Daemon to use new license: OK
```

2.2.5 Reboot After Installing License

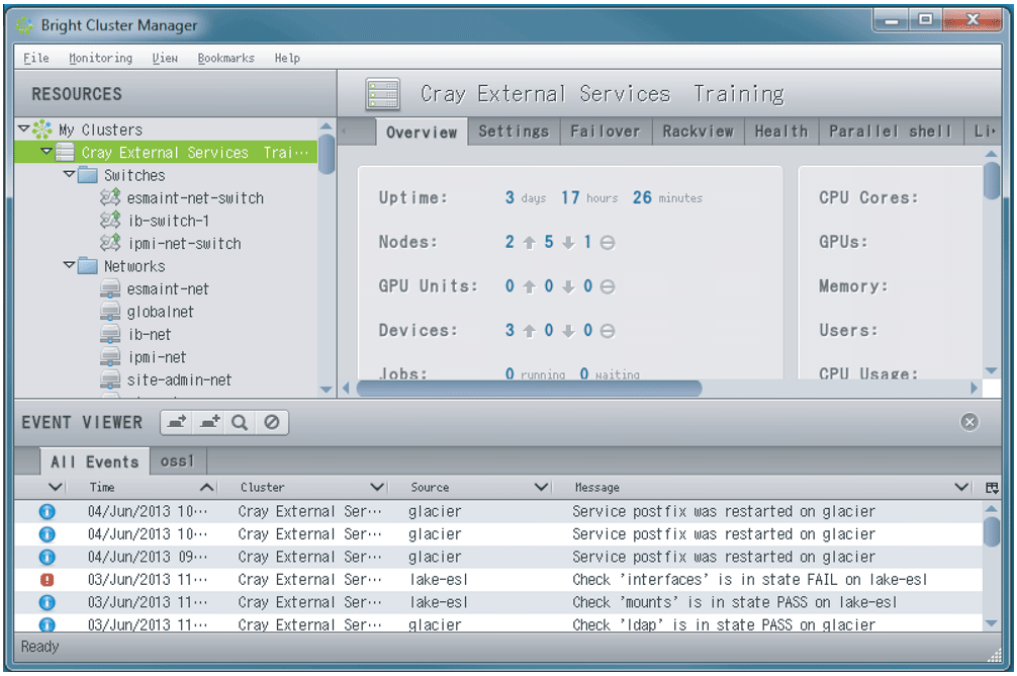
After using a product key with `request-license`, reboot the system using the **`pexec reboot`** command from the CIMS.

2.3 The Bright GUI

Refer to Install and Run `cmgui` on a Remote System on page 57 for procedures to install the `cmgui`. To connect to a system, use the procedures in Run `cmgui` and Connect to a DMP System on page 58.

The Bright `cmgui` window provides a resource tree down the left side that lists all of the components in a system. Selecting a resource opens an associated tabbed pane on the right side of the window that allows tab-related parameters to be viewed and managed. The number of tabs displayed and their contents depend on the resource selected. When learning to use Bright software, the `cmgui` window may be easier to learn and understand as opposed to the shell (`cmsh`), conversely, the `cmsh` shell environment may be easier and more efficient to use for some tasks. Both user interfaces (`cmsh` and `cmgui`) provide the same administrative capabilities. Refer to the *Bright Cluster Manager 6.1 Administrator Manual* PDF, stored on the CIMS in the `/cm/shared/docs/cm` directory.

Figure 13. Bright `cmgui` Window



2.4 Install and Run `cmgui` on a Remote System

The Linux™, Windows®, or Mac OS® installation software for the cluster management GUI (`cmgui`) are located in `/cm/shared/apps/cmgui/dist` on the CIMS. The Bright 6.1 is a FireFox® plugin.

Procedure 3. Install and run `cmgui` on a remote system

Important: Whenever you update a CIMS with the latest ESM software, always reinstall the `cmgui` software on the remote systems so that software updates are applied.

1. Copy the Windows `.exe` file, (`install.cmgui.6.1.`*revision*`.exe`), Linux compressed TAR file (`cmgui.6.1.`*revision*`.tar.bz2`), or Mac OS

package file (`install.cmgui.macosx.6.1.`*revision*`.pkg`) from the `/cm/shared/apps/cmgui/dist` directory on the CIMS node to a `tmp` directory on the remote system.

```
remote% scp root@esms1:/cm/shared/apps/cmgui/dist/* /tmp
```

2. Copy the PFX certificate file from the `root` directory of the CIMS to a secure location on the remote system so that it can be used for authentication purposes. Rename the file so that you can identify which system it authorizes (`DMP_System-admin.pfx` for example).

   Important: Refer to Manage Bright `admin.pfx` Certificates on page 85 for important information about how to manage the Bright `admin.pfx` file holding the administrator certificate.

```
remote%  scp root@esms1:/root/admin.pfx /securelocation/DMP_System-admin.pfx
```

3. Install the software.

   a. On Windows systems, execute the installer `.exe` file and follow the instructions.

   b. On Linux systems, extract the files using the `tar` command:

```
Remote%  tar -xvjf cmgui.6.1-revision.tar.bz2
```

   c. On Mac OS systems, click on the `.pkg` file and following the instructions.

4. Start `cmgui` and select the power plug icon and enter the PFX certificate password to connect to the DMP system. See Figure 15.

5. Connect to the DMP system using the procedure in Run `cmgui` and Connect to a DMP System on page 58.

2.5 Run `cmgui` and Connect to a DMP System

Before making the initial connection from a desktop computer running `cmgui`, a PFX file containing both the certificate and private key must be copied from the CIMS (`/root/.cm/cmgui/admin.pfx`) on the DMP system and stored in a secure location on the remote system.

Important: Refer to Manage Bright `admin.pfx` Certificates on page 85 for important information about how to manage the Bright `admin.pfx` file holding the administrator certificate.

If you need to manage more than one DMP system, rename the `admin.pfx` file on your local system appropriately. You only need to select and validate the `admin.pdx` file once.

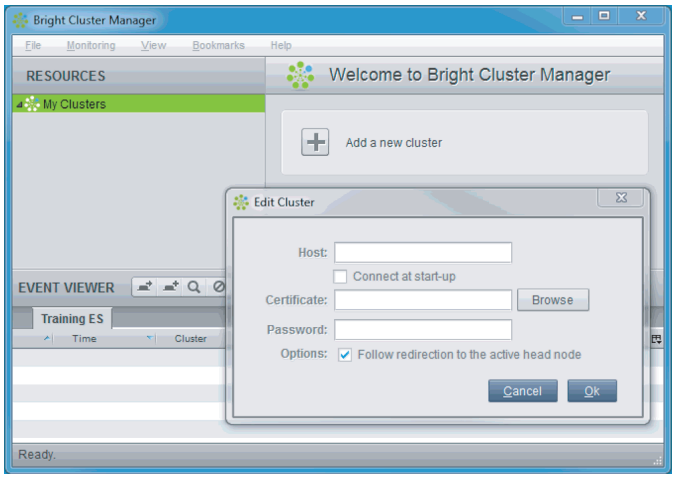
Procedure 4. Start cmgui and connect to a system

1. Copy the `admin.pfx` file to the remote computer. Rename the file to something specific, such as `cray_es_admin.pfx`.

```
remote# mkdir ~/cmgui-keys
remote# chmod 700 ~/cmgui-keys
remote# scp root@esms1:/root/.cm/cmgui/admin.pfx ~/cmgui-keys/cray_es_admin.pfx
```

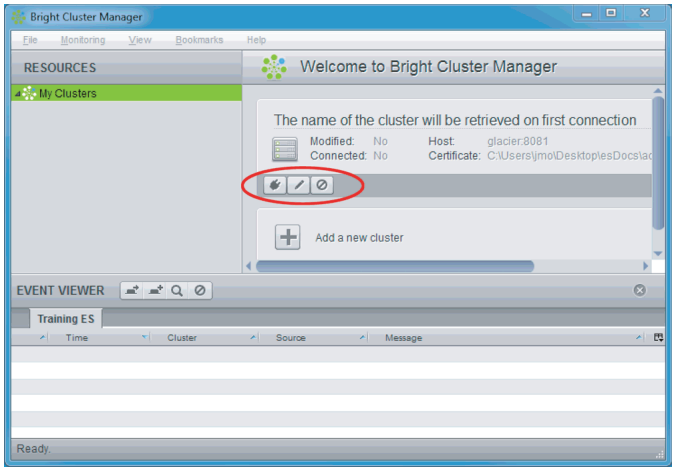
2. Run the `cmgui` executable.
3. To connect `cmgui` to a DMP system, select the + button. When you run `cmgui` and add a new system, you must enter the hostname of the CIMS, the location of the certificate file, and system password you configured during ESM installation and click **OK**.

Figure 14. Connect cmgui to a DMP System



4. The GUI window power plug icon enables you to connect to the system.

Figure 15. Connect `cmgui` to a DMP System



2.6 Run `cmgui` on the CIMS

The cluster management GUI (`cmgui`) is used to manage the system after you have installed and licensed the Bright software. The `cmgui` program may be run on the CIMS node and displayed to a remote X Window System™ running on a Linux™ desktop or other platform. The `cmgui` program may also be installed on a Linux, Windows®, or Mac OS® platform and supports a virtual network computing (VNC®) server for remote connections.

Important: Communication between the remote computer and the CIMS node should be encrypted. Cray recommends SSH port forwarding or SSH tunneling. When running the `cmgui` program from the remote computer, `cmgui` connects to the CIMS node using SSL. Cray recommends using SSH port forwarding when using VNC.

Procedure 5. Run `cmgui` on the CIMS

1. On a remote system such as a Linux desktop or PC, start an X-server application such as Xming or Cygwin/X.
2. Enter the following command to log in to the CIMS (in this example, `esms1`) with SSH X-forwarding.

```
remote% ssh -X root@esms1
esms1 #
```

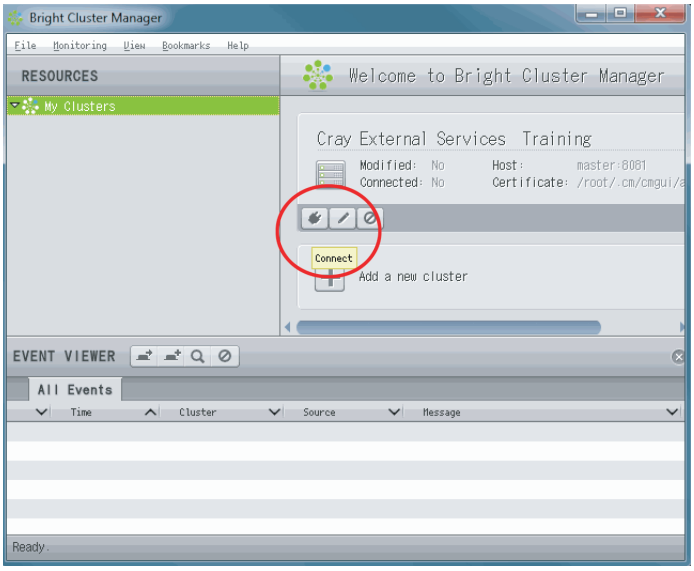
3. Start the `cmgui` program.

```
esms1# /cm/shared/apps/cmgui/cmgui &
```

Figure 16. `cmgui` Splash Screen

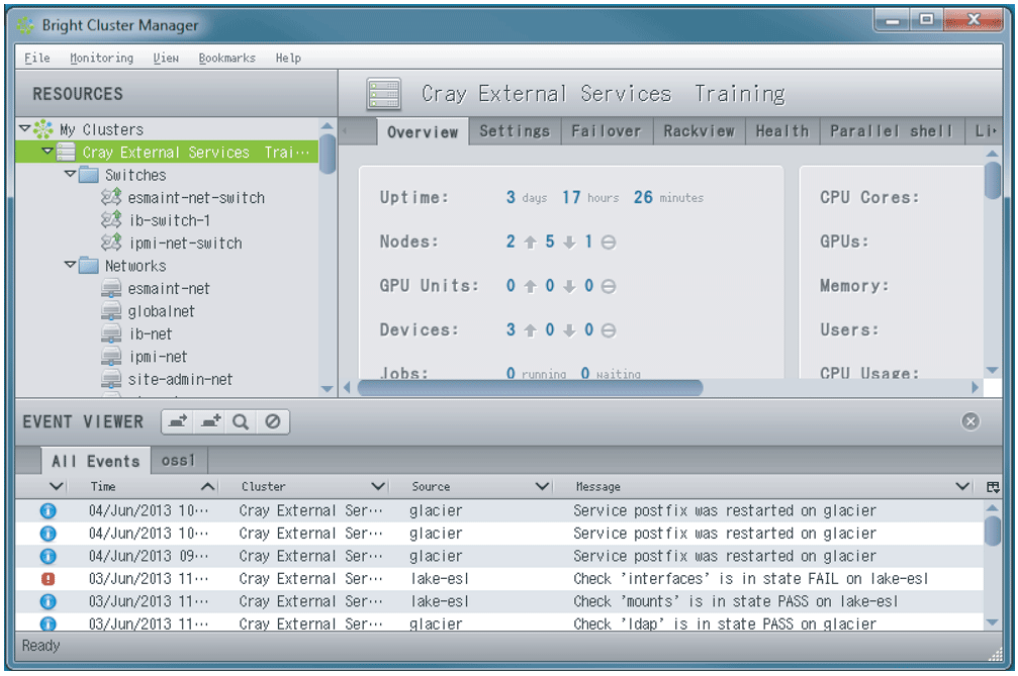
4. Select **Add a new cluster**. Enter the CIMS node hostname, username, and password and click **OK**.
5. Select the power plug icon and enter the system password to connect to the system.

Figure 17. `cmgui` Window



The `cmgui` window displays the DMP system configuration.

Figure 18. `cmgui` Connect-to-Cluster



2.7 Configure Virtual Network Computing (VNC®) on the CIMS

Virtual Network Computing (VNC®) software enables you to view and interact with the CIMS from another computer. The VNC software requires a TCP/IP connection between the server and the viewer. Firewalls or/and site security may restrict this connection. Refer to TightVNC (http://www.tightvnc.com/) for more information obtaining and installing VNC software.

You must configure a VNC account named `cray-vnc` before connecting to the VNC server.

Procedure 6. Start the VNC server

1. Log in to the CIMS as `root`.
2. Use the `chkconfig` command to check the current status of the server:

   ```
   esms1# chkconfig vnc
   vnc   off
   ```

3. Disable `xinetd` startup of `Xvnc`.

   If the `chkconfig` command you executed in step 2 reports that `Xvnc` was started by INET services (`xinetd`):

   ```
   esms1# chkconfig vnc
   vnc xinetd
   ```

Execute the following commands to disable `xinetd` startup of `Xvnc` (`xinetd` startup of `Xvnc` is the SLES 11 default, but it usually is disabled by `chkconfig`):

```
esms1# chkconfig vnc off
esm1# /etc/init.d/xinetd reload
Reload INET services (xinetd).                          done
```

If no other `xinetd` services have been enabled, the `reload` command will return `failed` instead of `done`. If the `reload` command returns `failed`, this is normal and you can ignore the `failed` notification.

4. Use the `chkconfig` command to start `Xvnc` at boot time:

   ```
   esms1# chkconfig vnc on
   ```

5. Start the `Xvnc` server immediately:

   ```
   esms1# /etc/init.d/vnc start
   ```

   If the password for `cray-vnc` has not already been established, the system prompts you for one. You must enter a password to access the server.

   ```
   Password: ********
   Verify:
   Would you like to enter a view-only password (y/n)? n
   xauth:  creating new authority file /home/cray-vnc/.Xauthority

   New 'X' desktop is esms1:1

   Creating default startup script /home/cray-vnc/.vnc/xstartup
   Starting applications specified in /home/cray-vnc/.vnc/xstartup
   Log file is /home/cray-vnc/.vnc/esms1:1.log

   esms1# ps -eda | grep vnc
   1839 pts/0    00:00:00 Xvnc
   ```

   The startup script starts the `Xvnc` server for display `:1`.

   To access the `Xvnc` server, use a VNC client, such as `vncviewer`, `tight_VNC`, `vnc4`, or a web browser. Direct it to the CIMS that is running `Xvnc`. Many clients allow you to specify whether you want to connect in view-only or in an active mode. If you choose active participation, every mouse movement and keystroke made in your client is sent to the server. If more than one client is active at the same time, your typing and mouse movements are intermixed.

   Commands entered through the VNC client affect the system as if they were entered from the CIMS. However, the main CIMS window and the VNC clients cannot detect each other. It is a good idea for the administrator who is sitting at the CIMS to access the system through a VNC client.

Procedure 7. Connect to VNC server through an SSH tunnel, using the `vncviewer`

Important: This procedure is for use with the TightVNC client program.

Verify that you have the `vncviewer -via` option available. If you do not, use Procedure 8 on page 64.

- If you are connecting from a workstation or laptop running Linux™, enter the `vncviewer` command shown below.

  The first password you enter is for `cray-vnc` on the CIMS. The second password you enter is for the VNC server on the CIMS, which was set when the VNC server was started for the first time using `/etc/init.d/vnc start` on the CIMS.

```
> vncviewer -via cray-vnc@esms1 localhost:1
Password: ********
VNC server supports protocol version 3.130 (viewer 3.3)
Password: ********
VNC authentication succeeded
Desktop name "cray-vnc's X desktop (esms:1)"
Connected to VNC server, using protocol version 3.3
```

Procedure 8. Connect to the VNC server through an SSH tunnel

This procedure assumes that the VNC server on the CIMS is running with the default port of `5901`.

1. This `ssh` command starts an `ssh` session between the local Linux computer and the CIMS, and it also creates an SSH tunnel so that port `5902` on the local host is forwarded through the encrypted SSH tunnel to port `5901` on the CIMS. You will be prompted for the `cray-vnc` password on the CIMS.

```
local_linux_prompt> ssh -L 5902:localhost:5901 esms1 -l cray-vnc
Password:
cray-vnc@esms1>
```

2. Now `vncviewer` can be started using the local side of the SSH tunnel, which is port `5902`. You will be prompted for the password of the VNC server on the CIMS. This password was set when the VNC server was started for the first time using `/etc/init.d/vnc start` on the CIMS.

```
remote% vncviewer localhost:2
Connected to RFB server, using protocol version 3.7
Performing standard VNC authentication
Password:
```

   The VNC window from the CIMS appears. All traffic between the `vncviewer` on the local Linux computer and the VNC server on the CIMS is now encrypted through the SSH tunnel.

Procedure 9. Connect an Apple® Mac® OS X system to the VNC server through an SSH tunnel

This procedure assumes that the VNC server on the CIMS is running with the default port of `5901`.

1. The following `ssh` command starts an `ssh` session between the local Mac OS X® computer and the CIMS, and it also creates an SSH tunnel so that port `5902` on the localhost is forwarded through the encrypted SSH tunnel to port `5901` on the CIMS. You will be prompted for the `cray-vnc` password on the CIMS.

   ```
   local_mac_prompt> ssh -L 5902:localhost:5901 esms1 -l cray-vnc
   Password:
   cray-vnc@esms1>
   ```

2. The `vncviewer` can now be started using the local side of the SSH tunnel, which is port `5902`. You will be prompted for the password of the VNC server on the CIMS. This password was set when the VNC server was started for the first time using `/etc/init.d/vnc start` on the CIMS.

   If you type this on the Mac OS X command line after having prepared the SSH tunnel, the `vncviewer` window displays.

   ```
   local_mac_prompt% open vnc://localhost:5902
   ```

   The VNC window from the CIMS appears. All traffic between the `vncviewer` on the local Mac OS X computer and the VNC server on the CIMS is now encrypted through the SSH tunnel.

Procedure 10. Connect to the VNC server through an SSH tunnel with Windows®

If you are connecting from a computer running Windows®, then both a VNC client program, such as TightVNC and an SSH program, such as PuTTY, SecureCRT®, or OpenSSH are recommended.

1. The same method described in Procedure 8 can be used for computers running the Windows operating system.

   Although TightVNC encrypts VNC passwords sent over the network, the rest of the traffic is sent unencrypted. To avoid a security risk, install and configure an SSH program that creates an SSH tunnel between TightVNC on the local computer (localhost port `5902`) and the remote VNC server (localhost port `5901`).

   Details about how to create the SSH tunnel vary amongst the different SSH programs for Windows computers.

2. After installing TightVNC, start the VNC viewer program by double-clicking on the **TightVNC** icon. Enter the hostname and VNC screen number, `localhost:`*number* (such as, `localhost:2` or `localhost:5902`), and then click on the **Connect** button.

2.8 The Command Shell (cmsh)

The cluster management shell (cmsh) provides a command-line interface to the system. The cmsh and the cmgui each provide the same capability. The command-line shell (cmsh) is invoked from an interactive session (through ssh) on the CIMS node, but cmsh can also be used to manage a cluster remotely. This section introduces the cmsh and provides examples of common tasks.

Refer to the *Bright Cluster Manager 6.1 Administrator Manual* PDF file on the CIMS node for more information about using pexec, foreach loops, and specifying ranges in cmsh.

Enter the following command as the root user to start the cmsh on the CIMS node:

```
esms1# cmsh
[esms1]%
```

When you run the cmsh from a UNIX® shell without arguments, it starts an interactive session. To return to the UNIX shell, enter the quit command:

```
[esms1]% quit
esms1#
```

The cmsh can be used in batch mode by specifying a command using the -c flag. Commands can be separated using semi-colons (;).

```
esms1# cmsh -c "main showprofile; device status eslogin01"
admin
eslogin-01 ............... [ UP ]
esms1#
```

The syntax for the cmsh is listed below:

```
cmsh [options] ................ Connect to localhost using default port
cmsh [options] <--certificate|-i certfile> <--key|-k keyfile> <host[:port]>
     Connect to a cluster using certificate and key in PEM format
cmsh [options] <--certificate|-i certfile>
     [-password|-p password] <uri[:port]>
     Connect to a cluster using certificate in PFX format

Valid options:
--help|-h ..................... Display this help
--noconnect|-u ................ Start unconnected
--controlflag|-z .............. ETX in non-interactive mode
--nossl|-s .................... Do not use SSL
--norc|-n ..................... Do not load cmshrc file on start-up
--command|-c <"c1; c2; ..."> .. Execute commands and exit
--file|-f <filename> .......... Execute commands in file and exit
--echo|-x ..................... Echo all commands
--quit|-q ..................... Exit immediately after error
```

Alternatively, commands can be piped to the `cmsh` from the UNIX® command line as `root` user:

```
esms1# echo device status | cmsh
eslogin01 ..............  [ UP ]
mycluster ..............  [ UP ]
oss001 .................  [ UP ]
oss002 .................  [ UP ]
switch01 ...............  [ UP ]
esms1#
```

The cluster management functions are grouped in separate `cmsh` modes. The first thing you must do when performing a cluster management operation is switch to the appropriate mode. The `cmsh` modes are listed below:

```
disconnect ................... Disconnect from cluster
connect ...................... Connect to cluster
quit ......................... Quit shell
exit ......................... Exit from current object or mode
help ......................... Display this help
run .......................... Execute cmsh commands from specified file
alias ........................ Set aliases
unalias ...................... Unset aliases
modified ..................... List modified objects
export ....................... Display list of aliases current list formats
events ....................... Manage events
list ......................... List state for all modes
category ..................... Enter category mode
cert ......................... Enter cert mode
device ....................... Enter device mode
jobqueue ..................... Enter jobqueue mode
jobs ......................... Enter jobs mode
main ......................... Enter main mode
monitoring ................... Enter monitoring mode
network ...................... Enter network mode
nodegroup .................... Enter nodegroup mode
partition .................... Enter partition mode
process ...................... Enter process mode
profile ...................... Enter profile mode
session ...................... Enter session mode
softwareimage ................ Enter softwareimage mode
test ......................... Enter test mode
user ......................... Enter user mode
```

Type **`device`** at the `cmsh` prompt to enter device mode.

```
[esms1]% device
[esms1->device]% list
Type             Hostname             MAC             Ip
---------------- -------------- ------------------ ---------------
EthernetSwitch     switch01    00:00:00:00:00:00     10.142.253.1
MasterNode         mycluster   00:E0:81:34:9B:48     10.142.255.254
ossNode            oss0        00:E0:81:2E:F7:96     10.142.0.1
ossNode            oss2        00:30:48:5D:8B:C6     10.142.0.2
[esms1->device]% exit
[esms1]%
```

Most modes in `cmsh` require that you specify an object, for instance, `device` mode requires that you specify *device objects* such as `esfs-mds1`, or `ib-switch-1`, and `network` mode requires you to specify *network objects* such as `esmaint-net` or `site-user-net`. The commands that can be used for controlling objects are the same in all modes. Table 5 lists the commands that may be used to act on objects in a particular mode.

Table 5. Command Shell Object Descriptions

Command	Description
`use`	Make the specified object the current object
`add`	Create an object and make it the current object
`clone`	Clone an object and make it the current object
`remove`	Remove an object
`commit`	Commit local changes to an object to the cluster management infrastructure
`refresh`	Undo local changes to an object
`list`	List all objects
`format`	Set formatting preferences for `list` output
`show`	Display all properties of an object
`get`	Display a particular property of an object
`set`	Set a particular property of an object
`clear`	Set empty value for a particular property of an object
`append`	Append a value to a particular list-property of an object
`removefrom`	Remove a given value from a particular list-property of an object
`modified`	Lists objects with uncommitted local changes
`usedby`	Lists objects that depend on a particular object
`validate`	Perform validation-check on the properties of an object

2.8.1 Mix `cmsh` and UNIX Shell Commands

You can execute UNIX commands while you perform cluster management. The `cmsh` enables users to execute UNIX commands by prefixing the command with a `!` character.

Example 5. Mixing cmsh and UNIX commands

```
esms1# cmsh
[esms1]% !hostname -f
esms1.cm.cluster
[esms1]%
```

The cmsh enables users to execute UNIX commands by prefixing the command with a ! character. When you exit the sub-shell, you return to the cmsh prompt. It is also possible to use the output of UNIX shell commands as part of a cmsh command by using the "backtick" syntax that is available in most UNIX shells as shown in Example 6.

Example 6. Using UNIX output in cmsh commands

```
[esms1->device]% device use `hostname`; status
cf-esms01 ..... [   UP   ]
[esms1->device]%
```

Similar to UNIX shells, cmsh also supports output redirection through common operators such as >, >> and |.

While looping over objects it may be helpful to execute a cmsh command for several objects simultaneously. The foreach can be used in several cmsh modes which enables you to loop over a list of objects. A foreach command takes a list of object names separated by spaces, and a list of commands that must be enclosed by (and) characters. The foreach command iterates over the specified objects and executes commands for each loop iteration. Example 7 shows an example of the foreach command syntax:

Example 7. Use a foreach loop to invoke commands

```
[esms1->device]%  foreach Object...Object ( Command; Command; )
[esms1->device]% foreach oss001 oss002 (get hostname; status)
oss001
oss001 ............. [ UP ]
oss002
oss002 ............. [ UP ]
[esms1->device]%
```

2.8.2 Specify a Range of Nodes in cmsh

You can act on a range of nodes in cmsh as shown in the example.

Example 8. Specify a range of objects in cmsh

```
[esms1->device]% check -n eslogin001..eslogin005 ib
Device Health Check Value Age (sec.)Info Message
------------------------------------------------------------------------
eslogin001 ib no data 0 Node down
eslogin002 ib PASS 0
eslogin003 ib PASS 0
eslogin004 ib PASS 0
eslogin005 ib PASS 0

[esms1->device]% reboot esfs-mds[1,2],esfs-oss[1,2,3,4]
```

2.8.3 Parallel Shell Execution

The parallel shell execution command, `pexec`, can be run from within the OS shell (bash by default), or from within CMDaemon (`cmsh` or `cmgui`). The OS shell `pexec` commands run on the nodes sequentially by default, and wait for the output from one node before continuing. If the OS shell `pexec` is run with the background execution option (`-b`), then the `bash` commands are executed in parallel. Running in parallel is not done by default, because it could be risky for some commands, such as power-cycling nodes with a reboot, which may put unacceptable surge demands on the power supplies. For example: Within `cmsh` or `cmgui`, the execution of a `power reset` command from `device` mode to power cycle a properly-configured group of nodes is safe, due to safeguards in CMDaemon to prevent nodes powering up too soon after each other.

CIMS Configuration Tasks [3]

3.1 Software Installation

Refer to the *Installing Cray Integrated Management Services (CIMS) Software* (S–2522) for CIMS software installation procedures.

3.1.1 Update Slave Node RPMs From ESM Media

If you update ESM software without updating slave node software, or if you are updating Bright software, run the `ESMupdateimage` command to update the RPMs delivered via the ESM media for all slave node production images.

> **Important:** Run `ESMupdateimage` command on software images after the ESM software is updated within the same version of Bright. Run `CIMSupgradeImages` on software images after upgrading to a new bright version.

Always reboot the slave nodes that are running the updated image, or push the updated image to the running slave node using `cmgui` **Update Node** button or `cmsh imageupdate` command from `device` mode.

The CIMS node and all CLFS nodes must run the same version of `lustre_control`. This RPM is also provided on the ESM media and is updated using the `ESMupdateimage` command.

Procedure 11. Update slave node RPMs from ESM media using `ESMupdateimage`

1. Log in to the CIMS as `root`.
2. Run `ESMupdateimage` for each software image. The `ESMupdateimage` command verifies that the software image exists in `/cm/images` and in the Bright database (via `cmsh`).

```
esms1# ESMupdateimage -s softwareimage
```

If it is not a valid software image, the command aborts. If the software image is valid, the command determines whether the software image has ESF or ESL software installed (specified in `/etc/opt/cray/release/ESFrelease` or `/etc/opt/cray/release/ESLrelease`) and then installs the proper RPMs for CLFS or CDL nodes. If neither of these files is present, then a generic set of slave node RPMs is installed.

3. If possible, reboot the node. When the node reboots it will get all of the changes to the updated software image. If that is not preferable, update the slave node from the software image using the `imageupdate` command from `cmsh` or **Update Node** button from `cmgui`.

   To reboot the node:

```
esms1# cmsh
[esms1]% device
[esms1->device]% reboot -n slavenode
```

   To push the updated software image to the node.

```
esms1# cmsh
[esms1]% device
[esms1->device]% imageupdate -w -n slavenode
```

   Use `synclog` to display the provisioning sync log for the node.

```
[esms1->device]% synclog -p slavenode
```

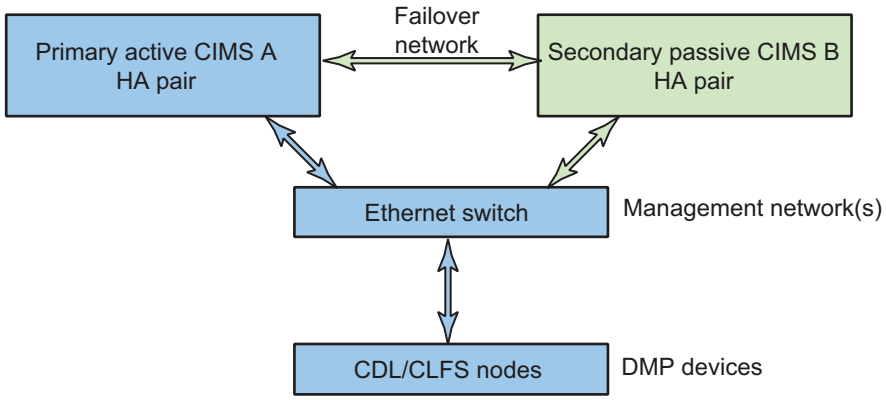
3.2 Clone a Replacement or Secondary CIMS Node

This section describes the procedure to replace CIMS node server hardware or update a secondary CIMS node in an HA configuration by cloning the primary CIMS node. The replacement procedure also supports single HA CIMS configurations. A simplified HA configuration is shown in the following figure. For non-HA configurations, ignore CIMS B and the failover network.

- CIMS A is the primary CIMS in the `Active` state.
- CIMS B is the secondary CIMS node (or replacement CIMS node if replacing CIMS node hardware) in the `Passive` state, if in an HA configuration.

For non-HA configurations, the replacement CIMS node may or may not be identical to the original CIMS node. If replacing a CIMS node in a high availability configuration, both CIMS nodes must have identical hardware configurations, most importantly, they must have an identical storage configuration.

Figure 19. CIMS HA Configuration



The process sequence to replace CIMS node hardware or update CIMS node HA hardware is as follows:

1. Clone the primary CIMS node (CIMS A in the figure) to the secondary CIMS node hardware.
2. Configure the secondary (or replacement) CIMS node RAID.
3. Configure the primary CIMS node to accept DHCP requests from unknown hosts.
4. PXE boot the secondary CIMS node from the primary CIMS node.
5. Enter the Bright Rescue Environment.
6. Edit the disk layout XML file. (When replacing a CIMS node in an HA configuration, do not change the disk setup.)
7. Edit the `excludelistnormal` file.
8. Run the `cm-clone-install` utility.
9. Reboot to a maintenance shell.
10. Mount the file system.
11. Run `mkinitrd`.
12. Reboot the secondary CIMS node.
13. A CIMS node in an HA configuration and DRBD management commands.

3.2.1 Clone Primary CIMS to New CIMS Hardware or Secondary CIMS

This section supports both HA and non-HA configurations.

3.2.1.1 Prerequisites

The following prerequisites are required before you begin:

- An open IP address on the `site-admin-net` for the secondary CIMS node.
- A keyboard, monitor, and mouse (or KVM) to configure the secondary CIMS node BIOS and RAID virtual disks.
- An unlocked Bright License Product Key and a temporary product key to use during the cloning operation.
- Password-less SSH access as `root` on both CIMS nodes (the primary CIMS and the CIMS being cloned).

3.2.1.2 Hardware Setup

Important: Use this hardware setup procedure only if you are replacing the CIMS node hardware. If you are upgrading software or if the disk on the secondary node is configured identically to the primary CIMS, then skip this section.

1. Connect a KVM to the secondary CIMS node.
2. Configure the secondary CIMS node RAID virtual disks and BIOS.
3. Configure the secondary CIMS node BIOS settings to PXE boot from `eth0`.
4. Connect `eth0` on the secondary CIMS node to the `esmaint-net` network of the primary CIMS. The CIMS node clone procedure uses this network.
5. Connect `eth1` interface on the secondary CIMS node to the `site-admin-net` network.

3.2.1.3 Clone the CIMS Node

Procedure 12. Clone the CIMS node

1. Login to the primary CIMS node as `root` and configure `cmd.conf` file to enable the primary CIMS node to accept DHCP requests. See Configure DHCP to Allow Requests from Unknown Nodes on page 156.
   a. Edit `/cm/local/apps/cmd/etc/cmd.conf`.
   b. Set `LockDownDhcpd = false` and save the file.
   c. Restart the `cmdaemon`.

```
esms1# /etc/init.d/cmd restart
```

2. From the primary CIMS node, use the temporary product key to license Bright for HA configuration. Make sure you have the MAC address of the secondary CIMS node. See Install the Bright License on page 51.

Example 9. Retrieve the MAC address from a node

```
esms1# cmsh -c “device use esms2; get mac”
```

- License for use in a HA configuration when prompted.

- Provide the `eth0` MAC address of the secondary CIMS node when prompted.
- Reuse primary keys from the previous license when prompted to avoid the requirement to reboot all slave nodes after installing the license.

3. Reboot the secondary CIMS node and configure the BIOS settings to enable the node to PXE boot from `eth0`.
4. Reboot the secondary CIMS node again and monitor the console. Select `Start Rescue Environment` when the menu displays on the console.
5. Login as `root`; no password is required.
6. Edit the `/cm/excludelistnormal` file. Add the following entries to the `excludelistnormal` file to prevent these directories from being cloned to the secondary CIMS node.

```
- /var/lib/named/proc
- /var/lib/ntp/proc
```

7. From the secondary CIMS node, run the `cm-clone-install` utility. Choose the correct command for either non-HA or HA configuration.

   For non-HA configurations, enter:

```
esms# cm/cm-clone-install --clone --hostname=new-hostname
```

   For HA configurations, enter the following command and reboot when prompted:

```
esms# /cm/cm-clone-install --failover
```

8. You can choose to edit the `disksetup.xml` file when prompted, or continue. Typically, this is required only for non-HA configurations if the disk size is different from the primary CIMS node or if the system will be converted to a HA configuration. When replacing a CIMS node in an HA configuration, do not change the disk setup. This step enables you to verify that the disk setup is correct. If so, continue.

The non-HA `disksetup.xml` file follows:

```
<diskSetup xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
     <device>
       <blockdev>/dev/sda</blockdev>
       <blockdev>/dev/hda</blockdev>
       <blockdev>/dev/vda</blockdev>
       <blockdev>/dev/xvda</blockdev>
       <partition id="a1">
         <size>512M</size>
         <type>linux</type>
         <filesystem>ext2</filesystem>
         <mountPoint>/boot</mountPoint>
         <mountOptions>defaults,noatime,nodiratime</mountOptions>
       </partition>
       <partition id="a2">
         <size>16G</size>
         <type>linux swap</type>
       </partition>
       <partition id="a3">
         <size>64G</size>
         <type>linux</type>
         <filesystem>ext3</filesystem>
         <mountPoint>/tmp</mountPoint>
       <mountOptions>defaults,noatime,nodiratime,nosuid,nodev</mountOptions>
       </partition>
       <partition id="a4">
         <size>max</size>
         <type>linux</type>
         <filesystem>ext3</filesystem>
         <mountPoint>/</mountPoint>
         <mountOptions>defaults,noatime,nodiratime</mountOptions>
       </partition>
     </device>
     <device>
       <blockdev>/dev/sdb</blockdev>
       <blockdev>/dev/hdb</blockdev>
       <blockdev>/dev/vdb</blockdev>
       <blockdev>/dev/xvdb</blockdev>
       <partition id="b1">
         <size>1024G</size>
         <type>linux</type>
         <filesystem>ext3</filesystem>
         <mountPoint>/var</mountPoint>
         <mountOptions>defaults,noatime,nodiratime</mountOptions>
       </partition>
       <partition id="b2">
         <size>10G</size>
         <type>linux</type>
         <filesystem>ext3</filesystem>
         <mountPoint>/var/lib/mysql/cmdaemon_mon</mountPoint>
         <mountOptions>defaults,noatime,nodiratime</mountOptions>
       </partition>
       <partition id="b3">
         <size>max</size>
         <type>linux</type>
         <filesystem>ext3</filesystem>
         <mountPoint>/cm</mountPoint>
         <mountOptions>defaults,noatime,nodiratime</mountOptions>
       </partition>
     </device>
   </diskSetup>
```

The HA `disksetup.xml` file follows:

```
<diskSetup xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
     <device>
       <blockdev>/dev/sda</blockdev>
       <blockdev>/dev/hda</blockdev>
       <blockdev>/dev/vda</blockdev>
       <blockdev>/dev/xvda</blockdev>
       <partition id="a1">
         <size>512M</size>
         <type>linux</type>
         <filesystem>ext2</filesystem>
         <mountPoint>/boot</mountPoint>
         <mountOptions>defaults,noatime,nodiratime</mountOptions>
       </partition>
       <partition id="a2">
         <size>16G</size>
         <type>linux swap</type>
       </partition>
       <partition id="a3">
         <size>64G</size>
         <type>linux</type>
         <filesystem>ext3</filesystem>
         <mountPoint>/tmp</mountPoint>
       <mountOptions>defaults,noatime,nodiratime,nosuid,nodev</mountOptions>
       </partition>
       <partition id="a4">
         <size>max</size>
         <type>linux</type>
         <filesystem>ext3</filesystem>
         <mountPoint>/</mountPoint>
         <mountOptions>defaults,noatime,nodiratime</mountOptions>
       </partition>
     </device>
     <device>
       <blockdev>/dev/sdb</blockdev>
       <blockdev>/dev/hdb</blockdev>
       <blockdev>/dev/vdb</blockdev>
       <blockdev>/dev/xvdb</blockdev>
       <partition id="b1">
         <size>1024G</size>
         <type>linux</type>
         <filesystem>ext3</filesystem>
         <mountPoint>/var</mountPoint>
         <mountOptions>defaults,noatime,nodiratime</mountOptions>
       </partition>
       <partition id="b2">
         <size>10G</size>
         <type>linux</type>
         <filesystem>ext3</filesystem>
         <mountPoint>/var/lib/mysql/cmdaemon_mon</mountPoint>
         <mountOptions>defaults,noatime,nodiratime</mountOptions>
       </partition>
       <partition id="b3">
         <size>3150G</size>
         <type>linux</type>
         <filesystem>ext3</filesystem>
         <mountPoint>/cm</mountPoint>
         <mountOptions>defaults,noatime,nodiratime</mountOptions>
       </partition>
       <partition id="b4">
         <size>20G</size>
         <type>linux</type>
         <filesystem>ext3</filesystem>
```

```
      <mountPoint>/drbd1</mountPoint>
      <mountOptions>defaults,noatime,nodiratime</mountOptions>
    </partition>
    <partition id="b5">
      <size>1G</size>
      <type>linux</type>
      <filesystem>ext3</filesystem>
      <mountPoint>/drbd2</mountPoint>
      <mountOptions>defaults,noatime,nodiratime</mountOptions>
    </partition>
    <partition id="b6">
      <size>1G</size>
      <type>linux</type>
      <filesystem>ext3</filesystem>
      <mountPoint>/drbd3</mountPoint>
      <mountOptions>defaults,noatime,nodiratime</mountOptions>
    </partition>
    <partition id="b7">
      <size>16G</size>
      <type>linux</type>
      <filesystem>ext3</filesystem>
      <mountPoint>/drbd4</mountPoint>
      <mountOptions>defaults,noatime,nodiratime</mountOptions>
    </partition>
    <partition id="b8">
      <size>max</size>
      <type>linux</type>
      <filesystem>ext3</filesystem>
      <mountPoint>/drbd5</mountPoint>
      <mountOptions>defaults,noatime,nodiratime</mountOptions>
    </partition>
  </device>
</diskSetup>
```

9. If configuring a non-HA CIMS node, proceed to Post Clone Procedures for Non-HA CIMS Configuration on page 78. If configuring an HA CIMS node, proceed to Post Clone Procedure for HA CIMS Configuration on page 80.

3.2.2 Post Clone Procedures for Non-HA CIMS Configuration

The following procedure is required only for non-HA CIMS configurations.

Procedure 13. Post clone procedures for non-HA CIMS configuration

1. After the clone completes, reboot the secondary (or replacement) CIMS node and configure the BIOS to boot from the optical drive first and then the hard disk.
2. Boot the replacement CIMS node from the Bright Installation DVD.
3. Login to the replacement CIMS node as `root`.
4. Remove the `/drbd*` mount points from the `/etc/fstab` file, and reboot the replacement CIMS node.
5. When the replacement CIMS node reboots, log in as `root`.

6. Mount the partitions for the target (`/dev/sda` and `/dev/sdb`).

```
esms# mkdir /localdisk
esms# mount /dev/sda5 /localdisk
esms# mount /dev/sda1 /localdisk/boot
esms# mount /dev/sda3 /localdisk/tmp
esms# mount /dev/sdb1 /localdisk/var
esms# mount /dev/sdb2 /localdisk/var/lib/mysql/cmdaemon_mon
esms# mount /dev/sdb3 /localdisk/cm
```

7. Bind mount `/dev`, `/proc` and `/sys` to their target partitions.

```
esms# mount --bind /dev /localdisk/dev
esms# mount --bind /proc /localdisk/proc
esms# mount --bind /sys /localdisk/sys
```

8. Use the `chroot` shell to run `mkinitrd_setup` and `mkinitrd` in the `/localdisk` image.

```
esms# chroot /localdisk
esms:/> mkinitrd_setup
Scanning scripts ...
Resolve dependencies ...
Install symlinks in /lib/mkinitrd/setup ...
Install symlinks in /lib/mkinitrd/boot ...
...
esms:/>  mkinitrd
Kernel image:   /boot/vmlinuz-3.0.74-0.6.8-default
Initrd image:   /boot/initrd-3.0.74-0.6.8-default
Root device:    /dev/sda5 (mounted on / as ext3)
...
```

The following error may be ignored:

```
WARNING: no dependencies for kernel module 'acpi_power_meter' found.
modprobe: Module acpi_pad not found.
WARNING: no dependencies for kernel module 'acpi_pad' found.
Kernel Modules: scsi_mod libata libahci ahci cdrom sr_mod sg dm-mod ata_piix crc-t10dif sd_mod st edd ipv6_lib ipv6
Features:       acpi block usb resume.userspace resume.kernel
34747 blocks
sh: -c: line 0: syntax error near unexpected token `('
sh: -c: line 0: `udevadm info -q name -n (hd) 2> /dev/null'
perl-bootloader: 2014-02-12 16:22:23 ERROR: GRUB::GrubDev2UnixDev: did not find a match for hd in the device map
Scanning scripts ...
Resolve dependencies ...
Install symlinks in /lib/mkinitrd/setup ...
Install symlinks in /lib/mkinitrd/boot ...
```

9. Edit the `/localdisk/etc/HOSTNAME` file.

```
esms:/> vi /localdisk/etc/HOSTNAME
```

10. Set the correct hostname for `eth0`.

11. Exit the `choot` shell and unmount `/localdisk`.

```
esms:/> umount /localdisk/dev
esms# umount /localdisk/sys
```

12. Reboot the replacement CIMS node to complete the configuration.

3.2.3 Post Clone Procedure for HA CIMS Configuration

The following procedure is required only for HA CIMS configurations.

Procedure 14. Post clone procedures for HA CIMS configuration

1. After the clone completes, reboot the secondary (or replacement) CIMS node and configure the BIOS to boot from the optical drive first, and then the hard disk.
2. Make sure that passwordless SSH as `root` works in both directions between the two CIMS nodes. Run `ssh-keygen -R` *CIMS* on each CIMS and then use SSH to connect to each CIMS from the other, to recapture the new SSH keys if necessary. This must work in both directions between the two CIMS before proceeding.
3. Finalize the primary CIMS node. From the primary CIMS server enter the following command and provide the MySQL root password, `initial0` when prompted.

```
esms# cmha-setup -x -c /cm/local/apps/cluster-tools/ha/conf/crayfailoverconf.xml -f -r
Please enter the mysql root password: mysqlpassword
   Updating secondary master mac address ..... [ OK ]
  Initializing failover setup on master2 ..... [ OK ]
                       Cloning database ..... [ OK ]
                  Update DB permissions ..... [ OK ]
 Checking for dedicated failover network ..... [ OK ]
 A reboot has been issued on esms2
```

4. On the secondary (or replacement) CIMS node, recover the DRBD devices.

   Important: Perform these steps on the **secondary CIMS node only.**

   a. Clear the first 1MB of each DRBD device (`/dev/sdb4` through `/dev/sdb8`)

```
# dd if=/dev/zero of=/dev/sdb4 bs=1M count=1
# dd if=/dev/zero of=/dev/sdb5 bs=1M count=1
# dd if=/dev/zero of=/dev/sdb6 bs=1M count=1
# dd if=/dev/zero of=/dev/sdb7 bs=1M count=1
# dd if=/dev/zero of=/dev/sdb8 bs=1M count=1
```

   b. Unmount the `/drbd*` partitions on both CIMS nodes.

Caution: Failure to unmount the `drbd` partitions from both CIMS nodes will cause a failure that requires the re-installation of both CIMS nodes.

```
esms1# umount /drbd1
esms1# umount /drbd2
esms1# umount /drbd3>
esms1# umount /drbd4
esms1# umount /drbd5
esms1# ssh esms2 umount /drbd1
esms1# ssh esms2 umount /drbd2
esms1# ssh esms2 umount /drbd3
esms1# ssh esms2 umount /drbd4
esms1# ssh esms2 umount /drbd5
```

c. Remove references to the `drbd` partitions from `/etc/fstab` on both CIMS nodes. The lines for `/drbd*` file systems similar to these should be removed from `/etc/fstab`.

```
esms1# vi /etc/fstab

Remove the lines that are similar to the lines below:

UUID=61d757e9-0d9a-46d1-a300-78297ee29d01 /drbd1   ext3  defaults,noatime,nodiratime    0 2
UUID=b7292424-07d4-4f6a-90d9-0792c5350e30 /drbd2   ext3  defaults,noatime,nodiratime    0 2
UUID=aca44a4d-ed9f-4589-9d4f-d9e55275b676 /drbd3   ext3  defaults,noatime,nodiratime    0 2
UUID=343a0ba1-3cb8-4354-902a-266f02a5f117 /drbd4   ext3  defaults,noatime,nodiratime    0 2
UUID=89035459-bc48-415c-8226-60d440d0ff0c /drbd5   ext3
defaults,noatime,nodiratime    0 2
```

d. Create DRBD metadata on the DRBD devices.

```
# drbdadm create-md all
```

e. Reattach all DRBD devices. They will begin syncing data automatically.

```
# drbdadm attach all
```

5. Reboot the secondary (or replacement) CIMS node. The secondary (or replacement) CIMS node configuration is complete. Verify the cloned CIMS node is configured correctly.

3.2.4 Verify the Cloned CIMS Node is Configured Correctly

Procedure 15. Verify the cloned CIMS node is configured correctly

1. Log in to both CIMS nodes as `root` and configure the `cmd.conf` file to block PXE boot requests. See Configure DHCP to Allow Requests from Unknown Nodes on page 156.

   a. Edit `/cm/local/apps/cmd/etc/cmd.conf`.

   b. Set `LockDownDhcpd = true` and save the file.

c. Restart the `cmdaemon`.

```
esms1# /etc/init.d/cmd restart
```

2. Enter the following commands on both CIMS nodes and compare the output from each command. The output should be identical, with exception of the CIMS node hostname and MAC addresses in the `device` listing.

```
esms1# cmsh –c “device list”
esms1# cmsh –c “network list”
esms1# cmsh –c “softwareimage list”
esms1# cmsh –c “network list”
esms1# diff /cm/node-installer/scripts/disks /cm/node-installer/scripts/disks.cray
```

There should be no differences between the `disks` file and the `disks.cray` file on both nodes. If there are differences copy the `disks.cray` file to `disks`.

3. Transfer the primary CIMS node cables to the CIMS replacement node.

4. Verify that power status can be read from the cloned CIMS node.

```
esms1# cmsh –c “device; power status”
```

5. Verify the device status from the cloned CIMS node.

```
esms1# cmsh –c “device status”
```

6. Reboot a slave node to verify the secondary (or replacement) CIMS node can provision and boot a node.

```
esms1# cmsh –c “device status”
```

7. License the secondary CIMS node using the permanent Bright site license product key. Reuse the product key to avoid rebooting all slave nodes. License the system as HA or single CIMS node.

3.3 Administrative Passwords

There are several administrative passwords for a Cray Data Management Platform (DMP) system. Each password is described below:

- CIMS password. The `root` password for the primary and secondary CIMS nodes (the same).
- Software images. The `root` password for software images: This allows a `root` log in to a slave node, and is stored in the software image.
- The node installer. The `root` password for the node-installer allows a `root` log in to the node when the node-installer minimal operating system is running. The node-installer stage prepares the node for the final operating system when the node boots.
- MySQL® password. The `root` password for MySQL allows a `root` log in to the MySQL server.

- The administrator certificate password: This password decrypts the `/root/admin.pfx` file on the CIMS node so that the administrator certificate can be submitted to CMDaemon for administrative tasks. See Manage Bright `admin.pfx` Certificates on page 85.
- The baseboard management controller (iDRAC) password. The iDRAC password for the CIMS node allows a root log in to the iDRAC to manage the system and start a remote console. The iDRAC password is changed using the `cmsh` shell, and not the `cm-change-passwd` script. See Change the Password for the Baseboard Management Controller (BMC or iDRAC) on page 87.
- Network switch administrative passwords should be managed by connecting a console to the switch and enter the switch configuration commands.

Procedure 16. Change DMP system passwords

1. Log in to the CIMS as `root`.

2. Enter `cm-change-passwd` and follow the prompts to change each password on the system.

```
esms1# cm-change-passwd
With this utility you can easily change the following passwords:
* root password of head node
* root password of slave images
* root password of node-installer
* root password of mysql
* administrator certificate for use with cmgui (/root/admin.pfx)

Note: if this cluster has a high-availability setup with 2 head
nodes, be sure to run this script on both head nodes.

Change password for root on head node? [y/N]: y
Changing password for root on head node.
Changing password for user root.
New UNIX password: newrootpassword
Retype new UNIX password: newrootpassword
passwd: all authentication tokens updated successfully.
Change password for root in default-image [y/N]: y
Changing password for root in default-image.
Changing password for user root.
New UNIX password: newdefaultimagepassword
Retype new UNIX password: newdefaultimagepassword
passwd: all authentication tokens updated successfully.
Change password for root in node-installer? [y/N]: y
Changing password for root in node-installer.
Changing password for user root.
New UNIX password: newnode-installerpassword
Retype new UNIX password: newnode-installerpassword
passwd: all authentication tokens updated successfully.
Change password for MYSQL root user? [y/N]: y
Changing password for MYSQL root user.
Old password: oldMYSQLpassword
New password: newMYSQLpassword
Re-enter new password: newMYSQLpassword
Change password for admin certificate file? [y/N]: y
Enter old password: oldcertificatepassword
Enter new password: newcertificatepassword
Verify new password: newcertificatepassword
Password updated
```

Important: See Manage Bright `admin.pfx` Certificates on page 85 for more information about changing passwords for the `admin.pfx` certificate.

3. Use `cmsh` to change the CIMS BMC (iDRAC port) password.

```
esms1# cmsh
[esms1]% partition use base
[esms1->partition[base]]% show
Parameter                      Value
------------------------------ --------------------------------
Administrator e-mail
BMC Password                   *********
BMC User ID                    2
BMC User name                  root
Burn configs                   <2 in submode>
Cluster name                   Training
Default burn configuration
Default category               default
Default software image         default-image
External network               site-admin-net
Externally visible IP
Failover                       not defined
Management network             esmaint-net
Masternode                     esms1
Name                           base
Name servers                   aaa.bbb.ccc.ddd  aaa.bbb.ccc.ddd
Node basename                  node
Node digits                    3
Notes                          <0 bytes>
Revision
Search domains                 your.domain.com
Time servers                   timeserver1.com timerserver2.com
Time zone                      America/Chicago
[esms1->partition[base]]% set bmcpassword newbmcpassword
[esms1->partition*[base*]]% commit
```

3.3.1 Manage Bright `admin.pfx` Certificates

Important: When an administrator changes the password on the system certificate, the certificate is re-encrypted with the new password. If an administrator has an old copy of a valid certificate that is encrypted with the old password, this administrator can continue to access the CMDaemon unless you revoke the old certificate. Old certificates must be revoked by using the `cmsh cert` mode `revokecertificate` command (refer to Procedure 17 on page 87), or by using the **Authentication** resource from the `cmgui` resource tree (see Figure 20).

The Bright Cluster Manager® (Bright) infrastructure (CMDaemon or `cmd`) requires public key authentication using X.509v3. X.509 is an ITU-T standard for a public key infrastructure (PKI) for single sign-on (SSO) and Privilege Management Infrastructure (PMI). The X.509 standard specifies, amongst other things, standard formats for public key certificates, certificate revocation lists, attribute certificates, and a certification path validation algorithm. This means in practice, a person authenticating to the cluster management infrastructure must present his/her certificate (i.e. the public key) and in addition must have access to the private key that corresponds to the certificate. A certificate includes a profile that determines which cluster management operations the holder of the certificate may perform.

The administrator password provided during Bright installation encrypts the `admin.pfx` file generated as part of the installation. The same password is also used as the initial `root` password for all nodes.

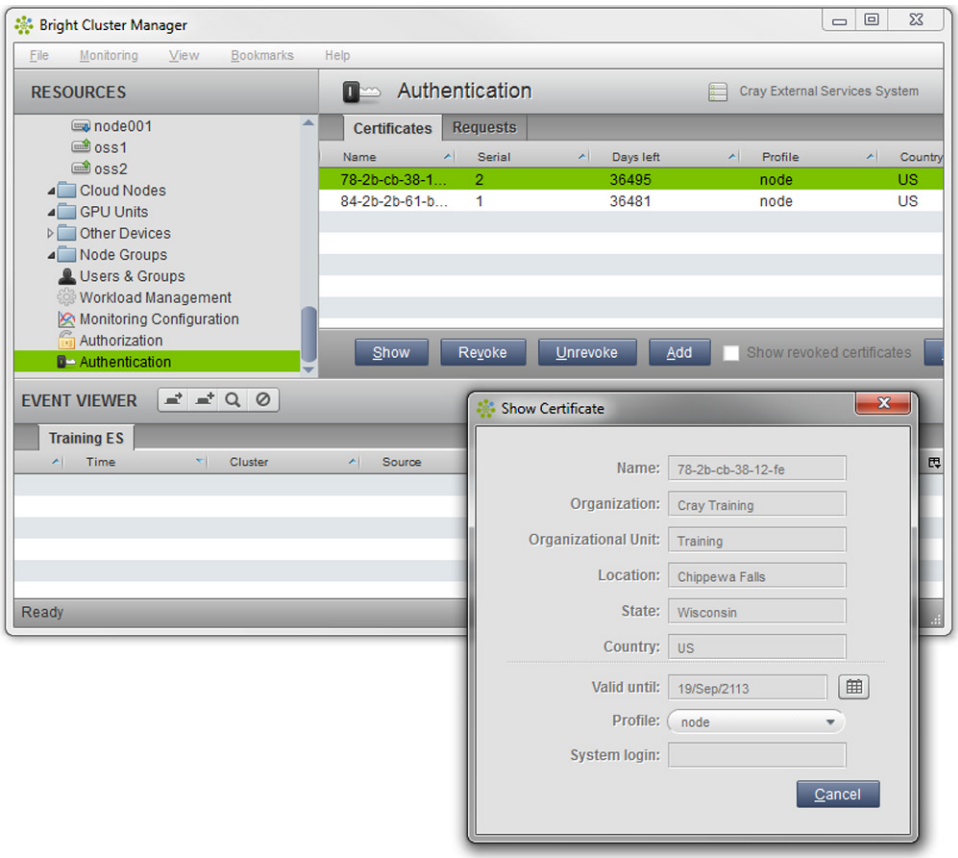
The administrator certificate is required to enable the CMDaemon and `cmsh` shell. Typically, administrators copy the `admin.pfx` file to their local laptop or workstation and use the Bright GUI (`cmgui`) to manage the system.

When the `/root/admin.pfx` file is updated with a new licence or password, the previous copy of the `admin.pfx` file continues to enable administrators to access the CMDaemon.

The password defined for the administrator certificate is used to decrypt the `admin.pfx` file, so that the administrator certificate can be presented to CMDaemon.

When the password for the `admin.pfx` file changes, the administrator must distribute the `admin.pfx` file and password to other administrators, and revoke older certificates to prevent administrators access with the old system certificate.

Figure 20. `cmgui` Authentication Menu



Procedure 17. Revoke administration certificates

1. Log in to the CIMS as `root` and start `cmsh`.

```
esms1# cmsh
```

2. Switch to `cert` mode.

```
[esms1]% cert
[esms1->cert]% listcertificates
```

3. List the certificates. The Name column shows the MAC address of the node's `esmaint-net` network adapter.

```
[esms1->cert]% listcertificates
Serial num   Days left  Profile          Country          Name               Revoked
------------ ---------- ---------------- ---------------- ------------------ --------
1            36481      node             US               84-2b-2b-61-b0-04  No
2            36495      node             US               78-2b-cb-38-12-fe  No
```

4. To revoke a certificate, specify the Serial number.

```
[esms1->cert]% revokecertificate 1
Certificate revoked.

[esms1->cert]% listcertificates
Serial num   Days left  Profile          Country          Name               Revoked
------------ ---------- ---------------- ---------------- ------------------ --------
1            36481      node             US               84-2b-2b-61-b0-04  Yes
2            36495      node             US               78-2b-cb-38-12-fe  No
```

3.3.2 Change the Password for the Baseboard Management Controller (BMC or iDRAC)

Procedure 18. Change the password on the BMC (iDRAC)

1. Log in to the CIMS as `root` and start `cmsh`.

```
esms1# cmsh
```

2. Switch to `partition` mode. Use the `base` partition to change the BMC password.

```
[esms1]% partition use base
[esms1->partition[base]]%
```

3. Get the BMC user name.

```
[esms1->partition[base]]% get bmcusername
root
```

4. Get the BMC password (set during installation).

```
[esms1->partition[base]]% get bmcpassword
bmcpassword
```

5. Change and commit the BMC password.

```
[esms1->partition[base]]% set bmcpassword
enter new password: NewPassWord
retype new password: NewPassWord
[esms1->partition[base*]]% commit
[esms1->partition[base]]%
```

3.4 Change CIMS Configuration Settings

You can modify the CIMS configuration settings such as baseboard management controller (BMC) password, name servers, search domains, and time servers using `cmsh partition` mode commands. The following example lists the CIMS configuration settings for the `base` partition. Use the `set` command from `partition` mode to set specific properties for the CIMS.

Example 10. CIMS configuration settings

```
esms1# cmsh
[esms1]% partition use base
[esms1->partition[base]]% show
Parameter                      Value
------------------------------ ------------------------------
Administrator e-mail
BMC Password                   *********
BMC User ID                    2
BMC User name                  root
Burn configs                   <2 in submode>
Cluster name                   Cray Training
Default burn configuration
Default category               default
Default software image         default-image
External network               site-admin-net
Externally visible IP
Failover                       not defined
Management network             esmaint-net
Masternode                     esms1
Name                           base
Name servers                   aaa.bbb.ccc.ddd aaa.bbb.ccc.ddd
Node basename                  node
Node digits                    3
Notes                          <0 bytes>
Revision
Search domains                 your.domain.com
Time servers                   timeserver1.com timerserver2.com
Time zone                      America/Chicago
```

3.5 Configure the RAID Virtual Disks

A CIMS node has six physical disks. You must reconfigure the CIMS node disks into two RAID-5 virtual disks, `/dev/sda` and `/dev/sdb`. The Bright software creates the required disk partitions during installation.

If you configure partitions for a single CIMS, then later add a second CIMS, you must resize the `/cm` partition for the HA configuration.

Procedure 19. Set up RAID virtual disks

This procedure includes detailed steps for the DELL™ R720 server using the PERC H710P Mini BIOS Configuration Utility 4.00-0014. Depending on your server model and version of RAID configuration utility, there could be minor differences in the steps to configure your system. For more information, refer to the documentation for your DELL™ PERC controller or server RAID controller software.

1. Connect a keyboard, monitor, and mouse to the front panel USB and monitor connectors on the CIMS.
2. Power up the CIMS. As the CIMS node reboots, enter the RAID controller configuration utility by pressing `Ctrl-R` when prompted.

   Cray recommends using the RAID configuration utility (via `Ctrl-R`) to configure the RAID virtual disks instead of the System Setup **Device Settings** menu.

   In this utility, use the `up-arrow` or `down-arrow` key to select (highlight) an item in a list. Press `Enter` to select items. To display a menu of options for an item, press the `F2` key. Use the `right-arrow`, `left-arrow`, or `Tab` key to change between the **Yes** and **No** buttons in a confirmation window.
3. Clear the default disk configuration, if necessary.
   a. If any disk groups are currently defined, select **Disk Group 0**, then press the `F2` key.
   b. Select **Delete Disk Group**, then press `Enter`.
   c. In the pop-up confirmation window, select **Yes** to confirm your changes.
4. Create a new virtual disk for `/dev/sda`. In this step, you will configure `/dev/sda` as a RAID-5 virtual disk with a capacity of 256 GB.
   a. Select **No Configuration Present**, then press the `F2` key.
   b. Select **Create New VD**, then press `Enter`. The **Create New VD** screen opens.
   c. Change the RAID level to RAID-5.
      1) Select **RAID Level**, then press `Enter` to display the available options.
      2) Select **RAID-5**.
      3) Press `Enter` to return to the main screen.
   d. Select all physical disks for this RAID-5 disk group.
      1) Press `Tab` to move to the **Physical Disks** area.

2) Press `Enter` to check the box for a physical disk. This action also advances the selection to the next disk.

3) Repeat the previous step for each physical disk.

e. Press `Tab` to move to **VD Size**, then enter **`256`**. Be sure to specify GB and not MB. The PERC controller software automatically adjusts this value to 255.9.

f. Press `Tab` to move to **VD Name**, then enter **`sda`**.

g. Enable disk initialization.

1) Press `Tab` to move to the **Advanced Settings** area.

2) Press `Enter` to check the **Advanced Settings** box so that you can make changes.

3) Select **Initialize**, then press `Enter` to check the box.

h. To confirm your changes, press the `Tab` key to select **OK**, then press `Enter`.

i. A message appears to let you know that initialization will destroy data on the virtual disk. Select **OK** to continue, then press `Enter`.

j. An `"Initialization complete"` message appears. Select **OK** to continue, then press `Enter`.

5. Add a new virtual disk for `/dev/sdb`. In this step, you will configure `/dev/sdb` as a RAID-5 virtual disk with the remainder of the available space.

a. Select **Disk Group: 0, RAID-5**, then press the `F2` key.

b. Select **Add New VD**, then press `Enter`. The **Add VD in Disk Group 0** screen opens.

c. Keep **VD Size** as presented (the remainder of the disks).

d. Press `Tab` to move to **VD Name**, then enter `sdb`.

e. Enable disk initialization.

1) Press `Tab` to move to the **Advanced Settings** area.

2) Press `Enter` to check the **Advanced Settings** box so that you can make changes.

3) Select **Initialize**, then press `Enter` to check the box.

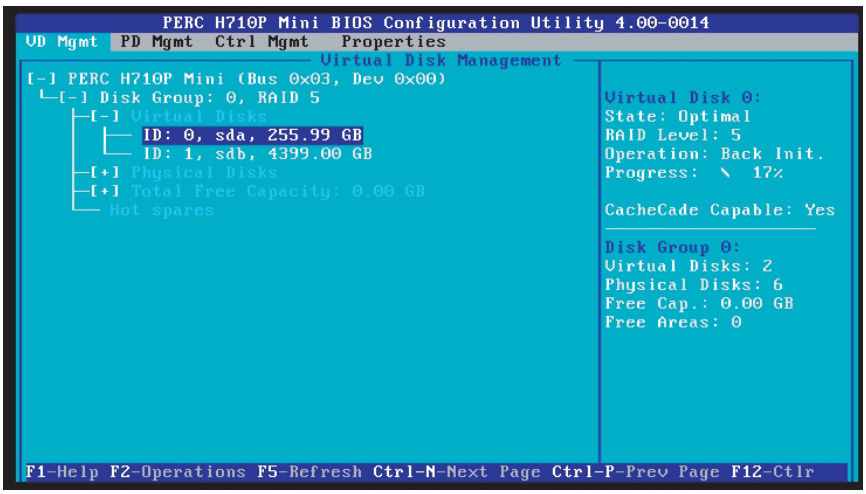
f. To confirm your changes, press the `Tab` key to select **OK**, then press `Enter`.

g. A message appears to let you know that initialization will destroy data on the virtual disk. Select **OK** to continue.

h. An "Initialization complete" message appears. Select **OK** to continue, then press Enter.

6. Verify the virtual disk changes. Compare your settings with those shown in Figure 21.

Figure 21. Final RAID Configuration Settings



7. To exit the RAID configuration utility, press the Escape key.

8. To confirm, press **OK**, then press Enter. Disk initialization is performed in the background, and takes about 2 hours to complete.

9. A message appears that prompts you to reboot. Press Ctrl-Alt-Delete. The server will restart the boot process and will not interrupt RAID initialization. During the system reboot, be prepared to type F2, when prompted, to change the system setup.

Refer to Configure the LSI® MegaCLI™ RAID Utility on page 91 for a procedure to configure the Bright megaraid health check for local RAID devices.

3.6 Configure the LSI® MegaCLI™ RAID Utility

CIMS nodes that use LSI® Corporation MegaRAID™ controllers, also the megaraid_sas kernel module with PERC710P, PERC 6/i RAID or other hardware modules. To manage, monitor, and configure the local RAID systems, install the LSI MegaCLI RAID utility on the CIMS. The MegaCLI utility also enables you to configure monitoring metrics, healthchecks, and administrator alerts in Bright that monitor the CIMS local RAID systems.

Use Procedure 20 on page 92 to install and run the MegaCLI utility. The utility installs in `/opt/MegaRAID/MegaCli`. Verify the utility is installed and running correctly, then use Procedure 22 on page 95 to configure the Bright `healthcheck` feature to monitor CIMS node RAID devices. Use Procedure 21 on page 93 to configure the MegaCLI utility for a slave node.

Procedure 20. Install the MegaCLI utility on the CIMS

1. Open a web browser and access the license agreement at http://www.lsi.com/Pages/user/eula.aspx?file=http://www.lsi.com/. Click **Accept** to accept the software license agreement.
2. Navigate to the storage downloads area of the LSI website and search for "MegaCLI".
3. Download the latest MegaCLI archive for Linux® (for example, `MegaCli_Linux.zip`) from the Downloads area of the `www.lsi.com` website.
   a. Log in to CIMS as `root`, copy the downloaded MegaCLI archive to the CIMS, and decompress the archive.

```
esms1# mkdir /root/MegaCLI
esms1# cd /root/MegaCLI
esms1# scp user@remotesystem:/user/MegaCli_Linux.zip /root/MegaCLI
esms1# unzip MegaCli_Linux.zip
Archive:  ./MegaCli_Linux.zip
   creating: MegaCli_Linux/
  inflating: MegaCli_Linux/MegaCli-8.07.08-1.i386.rpm
  inflating: MegaCli_Linux/megacli_8.07.08-1_all.deb
```

   b. Extract the Linux MegaCLI RPM from the archive.

```
esms1# pwd
/root/MegaCLI
esms1# cd MegaCli_Linux
esms1# rpm -ivh  MegaCli-Revision.noarch.rpm
```

4. Run the utility to verify it is functioning.

```
esms1# cd /opt/MegaRAID/MegaCli
esms1# ./MegaCli64 -AdpAllInfo -aAll |more
Adapter #0
==============================================================
                    Versions
                ================
Product Name    : PERC 6/i Integrated
Serial No       : 1122334455667788
FW Package Build: 6.0.2-0002

                    Mfg. Data
                ================
Mfg. Date       : 06/08/07
Rework Date     : 06/08/07
Revision No     :
Battery FRU     : N/A

                Image Versions in Flash:
                ================
FW Version         : 1.11.52-0396
BIOS Version       : NT13-2
WebBIOS Version    : 1.1-32-e_11-Rel
Ctrl-R Version     : 1.01-010B
Boot Block Version : 1.00.00.01-0008
...
```

5. Proceed to Procedure 22 on page 95 to configure the MegaCLI healthcheck in Bright.

Procedure 21. Install the MegaCLI utility on slave node

1. Open a web browser and access the license agreement at http://www.lsi.com/Pages/user/eula.aspx?file=http://www.lsi.com/. Click **Accept** to accept the software license agreement.
2. Navigate to the storage downloads area of the LSI website and search for "MegaCLI".
3. Download the latest MegaCLI archive for Linux® (for example, `MegaCli_Linux.zip`) from the Downloads area of the `www.lsi.com` website.
4. Log in to CIMS as `root`, copy the downloaded MegaCLI archive to the CIMS, and decompress the archive.

```
esms1# mkdir /root/MegaCLI
esms1# cd /root/MegaCLI
esms1# scp user@remotesystem:/user/MegaCli_Linux.zip /root/MegaCLI
esms1# unzip MegaCli_Linux.zip
Archive:  ./MegaCli_Linux.zip
   creating: MegaCli_Linux/
  inflating: MegaCli_Linux/MegaCli-8.07.08-1.i386.rpm
  inflating: MegaCli_Linux/megacli_8.07.08-1_all.deb
```

5. Clone the current working slave node software image and choose a unique name that identifies the MegaCLI utility image. Copy the current slave node image

from the UNIX™ prompt and wait for the copy to complete before cloning the image in Bright. This ensures the clone operation is complete before you continue.

```
esms1# cp -pr /cm/images/softwareimage /cm/images/megacli-image
```

6. Start `cmsh`, clone the slave node software image to an image named `megacli-image`.

```
esms1# cmsh
[esms1]% softwareimage
[esms1->softwareimage]% clone softwareimage megacli-image
[esms1->softwareimage*]% commit
```

7. Switch to `category` mode, and clone a slave node category to a new category named `megacli-category` and assign the software image (`megacli-image`) to the category (`megacli-category`).

```
[esms1->softwareimage*]% category
[esms1->category]% clone category megacli-category
[esms1->category*[megacli-category*]% commit
[esms1->category[megacli-category]% set softwareimage megacli-image
[esms1->category*[megacli-category*]% commit
[esms1->category[megacli-category]% quit
esms1#
```

8. Bind mount `/root/MegaCLI_Linux` to `/cm/images/megacli-image/tmp/MegaCLI` or copy the RPM to a directory in the `megacli-image` software image. This example shows the bind mount method.

```
esms1# mkdir -p /cm/images/megacli-image/tmp/MegaCLI
esms1# mount --bind /root/MegaCLI_Linux /cm/images/megacli-image/tmp/MegaCLI
```

9. Install the MegaCLI software on the slave node software image (`megacli-image`).

```
esms1# chroot /cm/images/megacli-image/
esms1:> rpm -ivh  MegaCli-Revision.noarch.rpm
reparing...                ########################################### [100%]
   1:MegaCli                ########################################### [100%]
esms1:> exit
```

Important: You must remove the bind mount.

10. Remove the bind mount.

```
esms1# umount /cm/images/megacli-image/tmp/MegaCLI
esms1# rm -f /cm/images/megacli-image/tmp/MegaCLI
```

11. Use `cmsh` to assign a slave node (in this example, `esfs-mds1`) to the `megacli-category`.

```
esms1# cmsh
[esms1]% device
[esms1->device]% use esfs-mds1
[esms1->device[esfs-mds1]]% set category megacli-category
[esms1->device*[esfs-mds1*]]% commit
```

12. Reboot the `esfs-mds1` node (or from `cmsh` use `imageupdate`) to test the `megacli-image` and exit `cmsh`.

```
[esms1->device[esfs-mds1]]% reboot esfs-mds1
[esms1->device[esfs-mds1]]% quit
```

Open a remote console to the `esfs-mds1` node and verify that the `megacli-image` software image boots without errors.

13. SSH to `esfs-mds1` and run the utility to verify that it is functioning on the slave node.

```
esms1# ssh esfs-mds01
Last login: Thu Nov  7 12:56:51 2013 from esms1.cm.cluster
[root@esfs-mds1 ~]# cd /opt/MegaRAID/MegaCli
[root@esfs-mds1 ~]# ./MegaCli64 -AdpAllInfo -aAll |more
Adapter #0
==============================================================
                    Versions
                ================
Product Name    : PERC 6/i Integrated
Serial No       : 1122334455667788
FW Package Build: 6.0.2-0002

                    Mfg. Data
                ================
Mfg. Date       : 06/08/07
Rework Date     : 06/08/07
Revision No     :
Battery FRU     : N/A

                Image Versions in Flash:
                ================
FW Version         : 1.11.52-0396
BIOS Version       : NT13-2
WebBIOS Version    : 1.1-32-e_11-Rel
Ctrl-R Version     : 1.01-010B
Boot Block Version : 1.00.00.01-0008
...
```

14. Assign other slave nodes to the `megacli-category`.

15. Proceed to Procedure 22 on page 95 to configure the `megaraid` health check in Bright.

Procedure 22. Configure the `megaraid` healthcheck in Bright

1. Log in to the CIMS as `root` and start `cmsh`.

```
esms1# cmsh
[esms1]%
```

2. Switch to `monitoring healthchecks` mode.

```
[esms1]% monitoring healthchecks
[esms1->monitoring->healthchecks]% list
Name (key)                  Command
--------------------------- ------------------------------------------------
DeviceIsUp                  <built-in>
ManagedServicesOk           <built-in>
chrootprocess               /cm/local/apps/cmd/scripts/healthchecks/chrootp+
cmsh                        /cm/local/apps/cmd/scripts/healthchecks/cmsh
diskspace                   /cm/local/apps/cmd/scripts/healthchecks/diskspa+
dmesg                       /cm/local/apps/cmd/scripts/healthchecks/dmesg
exports                     /cm/local/apps/cmd/scripts/healthchecks/exports
failedprejob                /cm/local/apps/cmd/scripts/healthchecks/failedp+
failover                    /cm/local/apps/cmd/scripts/healthchecks/failover
gpuhealth_quick             /cm/local/apps/cmd/scripts/healthchecks/gpuheal+
hardware-profile            /cm/local/apps/cmd/scripts/healthchecks/node-ha+
hpraid                      /cm/local/apps/cmd/scripts/healthchecks/hpraid
ib                          /cm/local/apps/cmd/scripts/healthchecks/ib
interfaces                  /cm/local/apps/cmd/scripts/healthchecks/interfa+
ldap                        /cm/local/apps/cmd/scripts/healthchecks/ldap
lustre                      /cm/local/apps/cmd/scripts/healthchecks/lustre
megaraid                    /cm/local/apps/cmd/scripts/healthchecks/megaraid
mounts                      /cm/local/apps/cmd/scripts/healthchecks/mounts
mysql                       /cm/local/apps/cmd/scripts/healthchecks/mysql
ntp                         /cm/local/apps/cmd/scripts/healthchecks/ntp
oomkiller                   /cm/local/apps/cmd/scripts/healthchecks/oomkill+
portchecker                 /cm/local/apps/cmd/scripts/healthchecks/portche+
rogueprocess                /cm/local/apps/cmd/scripts/healthchecks/roguepr+
schedulers                  /cm/local/apps/cmd/scripts/healthchecks/schedul+
smart                       /cm/local/apps/cmd/scripts/healthchecks/smart
ssh2node                    /cm/local/apps/cmd/scripts/healthchecks/ssh2node
swraid                      /cm/local/apps/cmd/scripts/healthchecks/swraid
testhealthcheck             /cm/local/apps/cmd/scripts/healthchecks/testhea+
```

3. If `megaraid` is present in the list, proceed at step 5. If `megaraid` is not present, add the `megaraid` healthcheck.

```
[esms1->monitoring->healthchecks]% add megaraid
[esms1->monitoring->healthchecks*[megaraid*]]% show
Parameter                      Value
------------------------------ ------------------------------------------------
Class of healthcheck           misc
Command
Description
Disabled                       no
Extended environment           no
Name                           megaraid
Notes                          <0 bytes>
Only when idle                 no
Parameter permissions          optional
Revision
Sampling method                samplingonnode
State flapping count           7
Timeout                        5
Valid for                      node,headnode
```

4. Configure the `megaraid` healthcheck and commit the settings.

```
[esms1->monitoring->healthchecks*[megaraid*]]% set classofhealthcheck disk
[esms1->monitoring->healthchecks*[megaraid*]]% set command /cm/local/apps/cmd/scripts/healthchecks/megaraid
[esms1->monitoring->healthchecks*[megaraid*]]% commit
[esms1->monitoring->healthchecks[megaraid]]%
```

5. Configure the `megaraid` healthcheck for the CIMS node. To configure the health check on a slave node, specify its category (in this example, `megacli-category`) using the `monitoring setup healthconf megacli-category` command.

```
[esms1->monitoring->healthchecks[megaraid]]% monitoring setup healthconf headnode
[esms1->monitoring->setup[HeadNode]->healthconf]% add megaraid
[esms1->monitoring->setup*[HeadNode*]->healthconf*[megaraid*]]% show
Parameter                      Value
------------------------------ ------------------------------------------------
Check Interval                 120
Disabled                       no
Fail Actions
Fail severity                  10
GapThreshold                   2
HealthCheck                    megaraid
HealthCheckParam
LogLength                      3000
Only when idle                 no
Pass Actions
Revision
Stateflapping Actions
Store                          yes
ThresholdDuration              1
Unknown Actions
Unknown severity               10
```

6. Commit the changes.

```
[esms1->monitoring->setup*[HeadNode*]->healthconf*[megaraid*]]% commit
[esms1->monitoring->setup[HeadNode]->healthconf[megaraid]]%
```

7. Wait a few minutes, and switch to `device` mode to show the health data for the CIMS or slave node. Verify that the `megaraid` health data shows `PASS`. To show the health data for a slave node (`esfs-mds1` for example), enter the command `device use esfs-mds1`.

```
[esms1->monitoring->setup[HeadNode]->healthconf[megaraid]]% device use esms1
[esms1->device[esms1]]% latesthealthdata
Health Check                 Severity Value           Age (sec.) Info Message
---------------------------- -------- --------------- ---------- ------------------------
DeviceIsUp                   0        PASS            1
ManagedServicesOk            0        PASS            19
mounts                       0        PASS            19
exports                      0        PASS            19
smart                        0        PASS            19         sda: Smart command failed
ldap                         0        PASS            19
failover                     0        PASS            19
interfaces                   40       FAIL            19         eth4 not up
oomkiller                    0        PASS            19
cmsh                         0        PASS            19
mysql                        0        PASS            19
failedprejob                 0        PASS            19
diskspace:2% 10% 20%         0        PASS            19
ntp                          0        PASS            19
schedulers                   0        PASS            19
chrootprocess                0        PASS            19
megaraid                     0        PASS            19
```

Use the following command to run the `megaraid` healthcheck from a command line.

```
[esms1->device[esms1]]% quit
esms1# /cm/local/apps/cmd/scripts/healthchecks/megaraid
PASS

esms1# /cm/local/apps/cmd/scripts/healthchecks/megaraid -d 3>&1
megacli command path: /opt/MegaRAID/MegaCli/MegaCli64
cmd:     /opt/MegaRAID/MegaCli/MegaCli64 -LdPdInfo -aALL -NoLog
line:    Adapter #0
adapter: 0
line:    Virtual Drive: 0 (Target Id: 0)
vdrive:  0
line:    State               : Optimal
vstate:  Optimal
line:    Span: 0 - Number of PDs: 3
span:    0
line:    PD: 0 Information
pdisk:   0
line:    Enclosure Device ID: 32
enc:     32
line:    Slot Number: 0
encslot: 0
line:    Firmware state: Online, Spun Up
pstate:  Online, Spun Up
line:    Drive has flagged a S.M.A.R.T alert : No
psmart:  No
line:    PD: 1 Information
...
```

3.7 Add a New or Modified Disk Setup XML File to the Bright Database

Important: If the default disk setup XML files are updated in a ESM release and the site disk setup XML files have been customized, system administrators must compare the newly released disk setup XML files with the current production disk setup XML files, and merge the changes manually. After the changes have been merged, you must load the updated disk setup file into the Bright database for the node category and reboot the node.

An ESM software update may modify the default disk setup XML files for the default `esFS-MDS` and `esFS-OSS` categories or default CDL categories `esLogin-XC` and `esLogin-XE`. CLFS categories for esfsmon 2.0.0 are `esfs-odd-`*filesystem*, `esfs-even-`*filesystem*, and `esfs-failed-`*filesystem*. New disk setup files may also be added to the `/opt/cray/esms/cray-es-diskpartitions-XX/default/` directory, which then must be added to the Bright database manually. Use this procedure to add a new disk setup XML file to an existing node category (in this example, the `esFS-MDS` category).

Procedure 23. Changing the disk setup XML file for a category

1. Log in to the CIMS as `root` and start `cmsh`.

```
esms1# cmsh
[esms1]%
```

2. Switch to `category` mode and select the `esFS-MDS` category.

```
[esms1]% category
[esms1->category]% use esFS-MDS
```

3. Get the current disk setup for the `esFS-MDS` category.

```
[esms1->category[esFS-MDS]]% get disksetup
```

4. Set your disk setup to either the full disk setup using 1TB capacity disks (`esfs-diskfull.xml`), or if the system contains smaller capacity disks, choose (`esfs-small-diskfull.xml`), depending on the hardware configuration.

```
[esms1->category[esFS-MDS]]% set disksetup
/opt/cray/esms/cray-es-diskpartitions-XX/default/esfs-diskfull.xml
```

or

```
[esms1->category[esFS-MDS]]% set disksetup
/opt/cray/esms/cray-es-diskpartitions-XX/default/esfs-small-diskfull.xml
```

5. Commit the change.

```
[esms1->category[esFS-MDS*]]% commit
```

6. Reboot the nodes in the `esFS-MDS` category.

```
[esms1->category[esFS-MDS]]% device
[esms1->device]% reboot -c esFS-MDS
esfs-mds1: Reboot in progress ...
esfs-mds2: Reboot in progress ...
```

7. Repeat this procedure for each CLFS category, (`esfs-odd-`*filesystem*, `esfs-even-`*filesystem*, and `esfs-failed-`*filesystem*) or other default node categories that use the new disk setup files.

3.8 Power Control

The Bright software, IPMI, and the DELL™ Remote Access Controller (iDRAC) enable you to monitor and control power remotely. (If the system includes intelligent PDUs, these too can be controlled and monitored from Bright.) Refer to Use the iDRAC Remote Console on page 135 for more information about the iDRAC.

The Bright `cmgui` **Overview** tab of a device can be used to check its power status information. Right-clicking a node in the resource tree also displays power control commands. The **Task** tab, enables you to select:

- Power on
- Power off
- Reset — powers off a device and powers it on again after a brief delay

When doing a power operation on multiple devices, CMDaemon inserts a 1 second delay between successive devices, to avoid power surges on the infrastructure. The delay period may be altered using `cmsh -d | --delay` option.

The following power control examples can be used from `cmsh device` mode. Log in to the CIMS as `root`, start `cmsh`, and switch to `device` mode.

```
esms1# cmsh
[esms1]% device
[esms1-<device]%
```

`power -n eslogin01 on`

Powers up an individual node such as, `eslogin01`

`power -n eslogin01..eslogin04,eslogin06 off`

Powers off a list of nodes, such as `eslogin01` to `eslogin04` and `eslogin06`

`power -c eslogin -d 10 reset`

Power cycles all nodes in the `eslogin` category, with a 10 second delay between each node power reset

`power -g DM_nodes`

Power on all nodes in the `DM_nodes` node group

`power -g esfs-oss status`

Check power status of all nodes in the `esfs-oss` node group.

`ON`	Power is on
`OFF`	Power is off
`RESET`	Displays during the short time the power is off during a power reset. The reset is a hard power off for PDUs, but can be a soft or hard reset for other power control devices.
`FAILED`	Power status communication failure
`FAILED`	Power status communication failure
`UNKNOWN`	Power status script timeout

`pexec power off`

Powers off all nodes

Bright software also supports power saving features through resource managers such as Simple Linux Utility for Resource Management (SLURM) or other workload management software. Refer to the *Bright Cluster Manager 6.1 Administrator Manual* PDF file on the CIMS node for more information.

3.9 Reboot Slave Nodes

Procedure 24. Rebooting slave nodes

1. Log in to the CIMS as `root` and start `cmsh`.

```
esms1# cmsh
[esms1]%
```

2. Switch to `device` mode and launch a remote console (`rconsole`) on the slave node (in this example `eslogin1`).

```
[esms1]% device use eslogin1
[esms1->device[eslogin1]]% rconsole
```

3. In a separate CIMS window, login as `root` and reboot the slave node using `cmsh` or use the **Reboot** button from the `cmgui`.

```
esms1# cmsh
[esms1]% device use eslogin1
[esms1->device[eslogin1]]% reboot
```

The following reboot examples can be used from `cmsh device` mode.

Log in to CIMS as `root`, start `cmsh`, and switch to `device` mode.

```
esms1# cmsh
[esms1]% device
[esms1->device]%
```

`reboot -n eslogin01`

Reboot an individual node

`reboot -n eslogin01..eslogin04,eslogin06`

To reboot a list of nodes

`reboot -c esLogin-XC`

Reboot all nodes in the `eslogin` category

`reboot -g Login`

Reboot all nodes in the `Login` node group

`pexec reboot`

Reboot all nodes

`reboot esfs-mds[1,2],esfs-oss[1,2,3,4]`

Specifies a range of nodes

See Parallel Shell Execution on page 70.

3.10 Shut Down Slave Nodes

These examples you how to perform an orderly shutdown and power off a node or nodes. You can shutdown nodes individually, by list, range, rack, and by category or node group.

The following reboot examples can be used from `cmsh device` mode. Log in to the CIMS as `root`, start `cmsh`, and switch to `device` mode.

```
esms1# cmsh
[esms1]% device
[esms1->device]%
```

`shutdown -n eslogin01`

Shutdown an individual node

`shutdown -n eslogin01..eslogin04,eslogin06`

Shutdown a list of nodes

`shutdown -c eslogin`

Shutdown all nodes in the `eslogin` category

`shutdown -g es-datamover`

Shutdown all nodes in the `es-datamover` node group

3.11 Network Settings

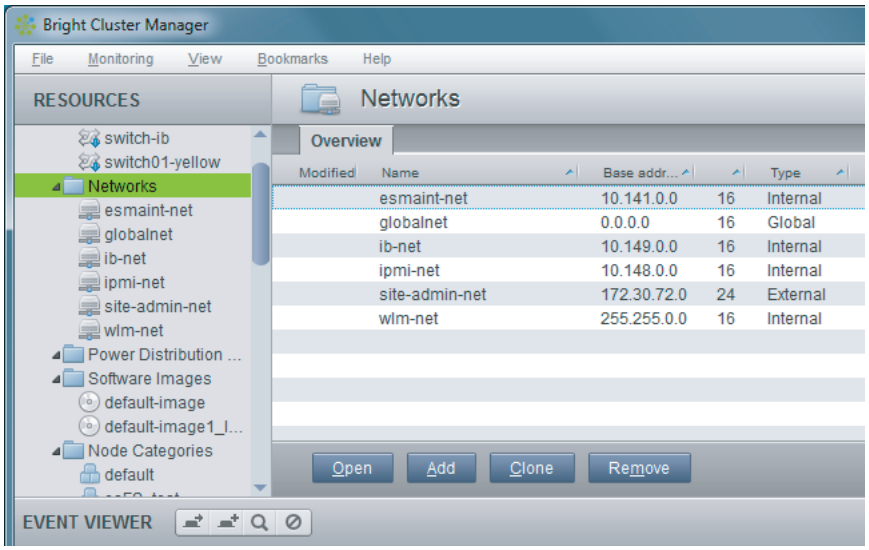
Bright Cluster Manager® (Bright) configures three default network objects. These are `internalnet`, `externalnet`, and `globalnet`. These are renamed for a DMP system, but used throughout the GUI menus as a way to classify the different networks.

Internal	The internal system network, or management network (these networks are renamed to `esmaint-net`, `ib-net`, `ipmi-net`, `wlm-net`).
External	The network connecting the DMP system to the outside world (`site-user-net`, `site-admin-net` typically a user or campus network).
Global	A special network used to set the domain name for nodes so that they can be resolved (not used in a Cray DMP system).

The figure shows an example of the networks in a Cray DMP system from the Bright GUI.

Select the `Networks` object in the **Resources** tree to view all the networks defined in the DMP system. You can sort on each of the columns,

Figure 22. Bright Network Configuration GUI



The `network` mode of the `cmsh` command can be used to modify network parameters for each of the networks defined in the system.

A DMP system requires the following networks:

`esmaint-net`

Internal management network that connects the CIMS server(s) with the slave nodes. This network enables Bright to manage and provision the slave nodes and other devices in the DMP system.

`ipmi-net` Internal IPMI/DRAC (Dell Remote Access Controller) network that provides remote console and power management of the slave nodes from the CIMS.

site-admin-net

External administration network used by site administrators to log in to the CIMS server (typically on the same network that the SMW is on). The name and IP address of this network are customized during installation. The CIMS IPMI interface (BMC or iDRAC) may also be on this network (instead of `ipmi-net`) to provide remote console and power management of the CIMS server.

`site-user-net`

External user network used by the slave nodes. On CDL nodes, this network provides user access and authentication services such as LDAP. On CLFS nodes, this network connects to the site LDAP for file ownership authentication. The name and IP address of this network are customized during installation.

`ib-net` InfiniBand® network used by the slave nodes for Lustre LNET traffic.

`failover-net`

Internal failover network used between two CIMS servers in an HA configuration for heartbeats between the active/passive CIMS nodes. This network does not connect to a managed switch.

The network parameters can be modified using Bright **Network→Settings** tab from the `cmgui`. See Figure 23. The table lists and describes the network parameters you can modify either from the `cmgui` or from the `cmsh`.

Figure 23. Network Settings GUI

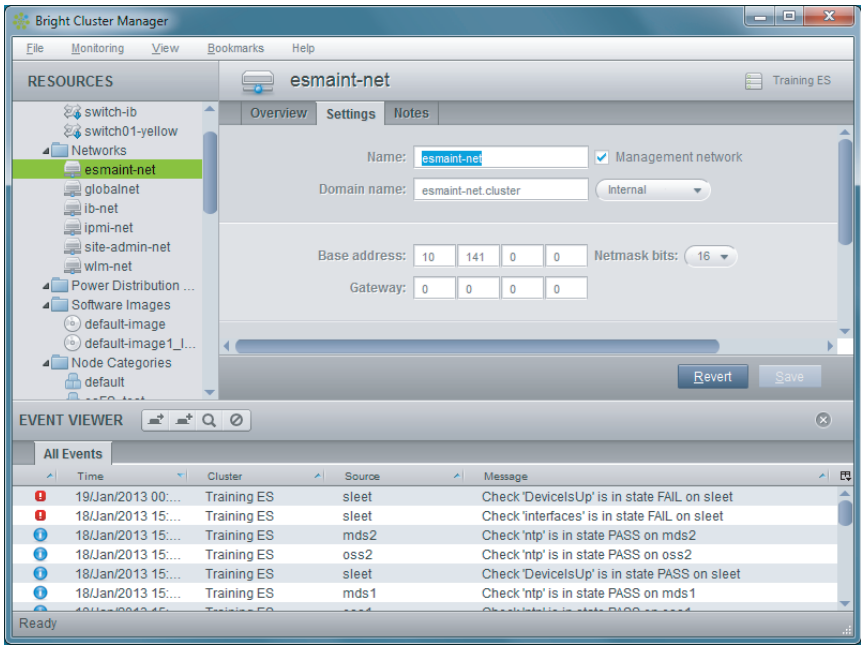


Table 6. Network Configuration Settings

Setting	Description
Name	Name of this network
Domain name	DNS domain associated with the network
Management network	Modify this setting if nodes are managed by the CIMS.
External network	Modify this setting if it is an external network.
Base address	Base address of the network (also known as the *network address*)
Gateway	Default route IP address

Setting	Description
Netmask bits	Prefix-length, or number of bits in netmask. The part after the `/` in classless inter-domain routing (CIDR) notation.
MTU	Maximum Transmission Unit. The maximum size of an IP packet transmitted without fragmenting. Typically set to 1500 for Ethernet networks. If configuring a 10GbE network, set the MTU to 9000 if using jumbo frames. Make sure to set all devices on the network to the same MTU value.
Dynamic range start/end	Start/end IP addresses of the DHCP range temporarily used by nodes during PXE boot on the internal network.
Allow node booting	Nodes set to boot from this network (useful in the case of nodes on multiple networks).
Do not allow nodes to boot from this network	New nodes are not offered a PXE DHCP IP address from this network (DHCPD is locked down by default in `/cm/local/apps/cmd/etc/cmd.conf`). The `lockdowndhcpd` setting is can also be configured in `cmsh network` mode for a specific network.

The example `cmsh` commands below show how to view or set the network parameters for a DMP system CIMS node (`esms1`) and `esmaint-net` network.

IP address

```
esms1# cmsh -c "device interfaces esms-1; get eth1 ip"
esms1# cmsh -c "device interfaces esms-1; set eth1 ip address;commit"
```

Base address

```
esms1# cmsh -c "network get esmaint-net baseaddress"
esms1# cmsh -c "network; set esmaint-net baseaddress address;commit"
```

Broadcast address

```
esms1# cmsh -c "network get esmaint-net broadcastaddress"
esms1# cmsh -c "network; set esmaint-net broadcastaddress address;commit"
```

Netmask bits

```
esms1# cmsh -c "network get esmaint-net netmaskbits"
esms1# cmsh -c "network; set esmaint-net netmaskbits bitsize;commit"
```

Gateway

```
esms1# cmsh -c "network get esmaint-net gateway"
esms1# cmsh -c "network; set esmaint-net gateway address; commit"
```

Name servers

```
esms1# cmsh -c "partition get base nameservers"
esms1# cmsh -c "partition; set base nameservers address; commit"
```

Search domains

```
esms1# cmsh -c "partition get base searchdomains"
esms1# cmsh -c "partition; set base searchdomains hostname;commit"
```

Time servers

```
esms1# cmsh -c "partition get base timeservers"
esms1# cmsh -c "partition; set base timeservers hostname; commit"
```

3.11.1 The `sipcalc` Utility

The `sipcalc`(1) utility installed on the CIMS node is a useful tool for calculating or checking such IP subnet values (see the man page on `sipcalc` or see `sipcalc -h` for help on this utility).

Example 11. The `sipcalc` utility

```
esms1# sipcalc 192.168.0.1/28
-[ipv4 : 192.168.0.1/28] - 0

[CIDR]
Host address                    - 192.168.0.1
Host address (decimal)          - 3232235521
Host address (hex)              - C0A80001
Network address                 - 192.168.0.0
Network mask                    - 255.255.255.240
Network mask (bits)             - 28
Network mask (hex)              - FFFFFFF0
Broadcast address               - 192.168.0.15
Cisco wildcard                  - 0.0.0.15
Addresses in network            - 16
Network range                   - 192.168.0.0 - 192.168.0.15
Usable range                    - 192.168.0.1 - 192.168.0.14
```

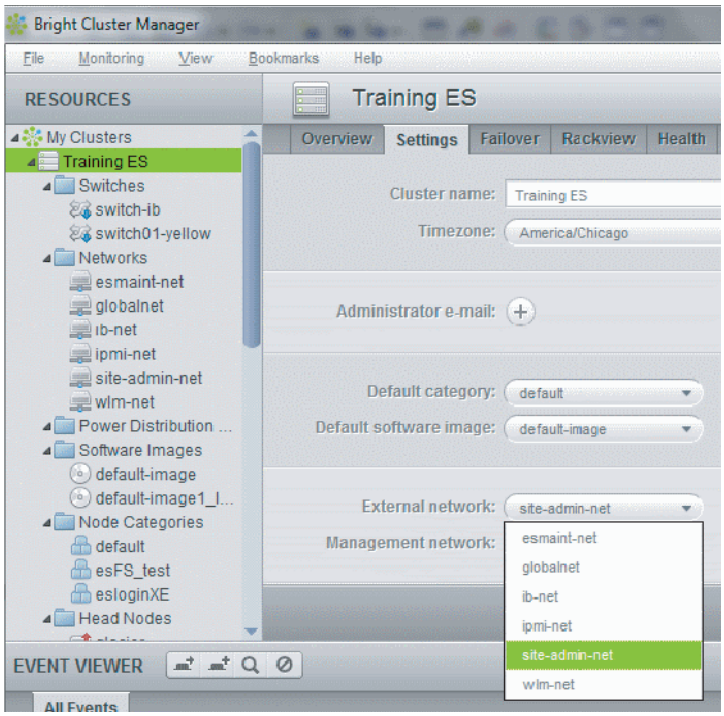
3.11.2 DNS Domains

Every network has an associated DNS domain which can be used to access a device through a particular network. For `esmaint-net`, the default DNS domain is set to `esmaint-net.cluster`, which means that the hostname `esms1.cm.cluster` can be used to access device `esms1` through the maintenance network. The InfiniBand® network domain is `ib-net.cm.cluster`. Internal DNS zones are generated automatically based on the network definitions and the defined nodes on these networks. For networks marked as external, no DNS zones are generated.

3.11.3 Add a Network

In `cmsh`, a new network can be added from the `network` mode using the `add` or `clone` commands. The default assignment of networks can be set from the GUI **Management network** and **External network** menus on the **Settings** tab of the top level DMP system object in the resource tree.

Figure 24. Add a Network



In `cmsh` the assignment to **Management network** and **External network** is set or modified from the base object in `partition` mode:

Example 12. Changing the default setting of a network

```
 esms1# cmsh
[esms1]% partition use base
[esms1->partition[base]]% set managementnetwork esmaint-net; commit
[esms1->partition[base]]% set externalnetwork site-user-net; commit
```

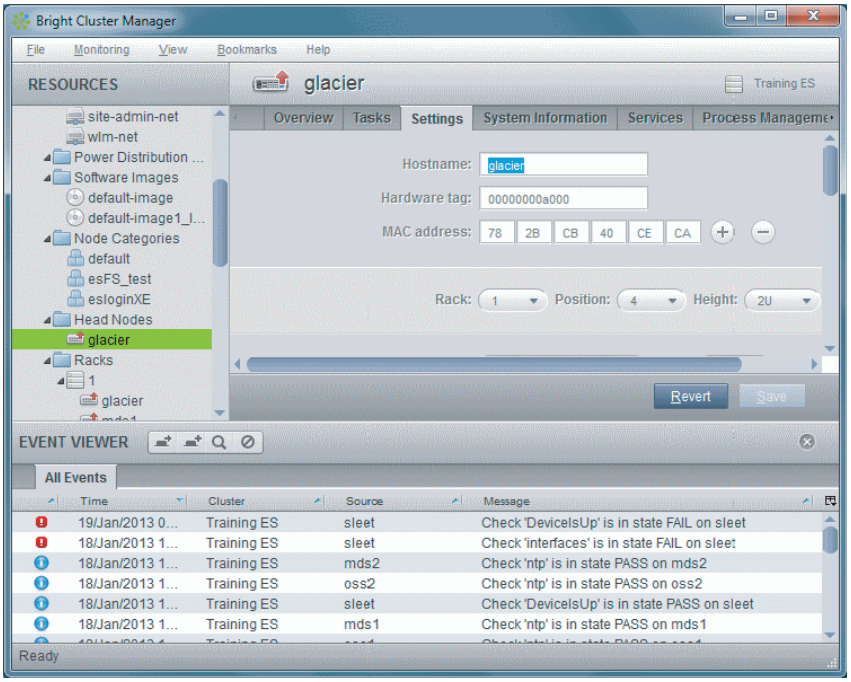
3.11.4 Change Node Hostnames

The alias `master` may be used to reach the head node. The name can be changed in a similar manner for each, following the guidelines in Device Names in Bright on page 42.

Procedure 25. Change node hostnames

1. To change the hostname of the head node (CIMS), the CIMS device object listed under **Head Nodes** must be modified.
2. Using `cmgui`, select the device listed under **Head Nodes** in the resource tree, then select the **Settings** tab.

Figure 25. Change Node Hostnames with `cmgui`



3. Modify the **Hostname** property (follow guides in Device Names in Bright on page 42), and click on the **Save** button.

In `cmsh`, the hostname of the head node is changed in `device` mode:

Example 13. Change node hostnames with `cmsh`

```
esms1# cmsh
[esms1]% device use esms1
[esms1->device[esms1]]% set hostname esms2
[esms2->device*[esms2*]]% commit
[esms2->device[esms2]]% quit
esms1# sleep 30; hostname -f esms2.cm.cluster esms2.cm.cluster
```

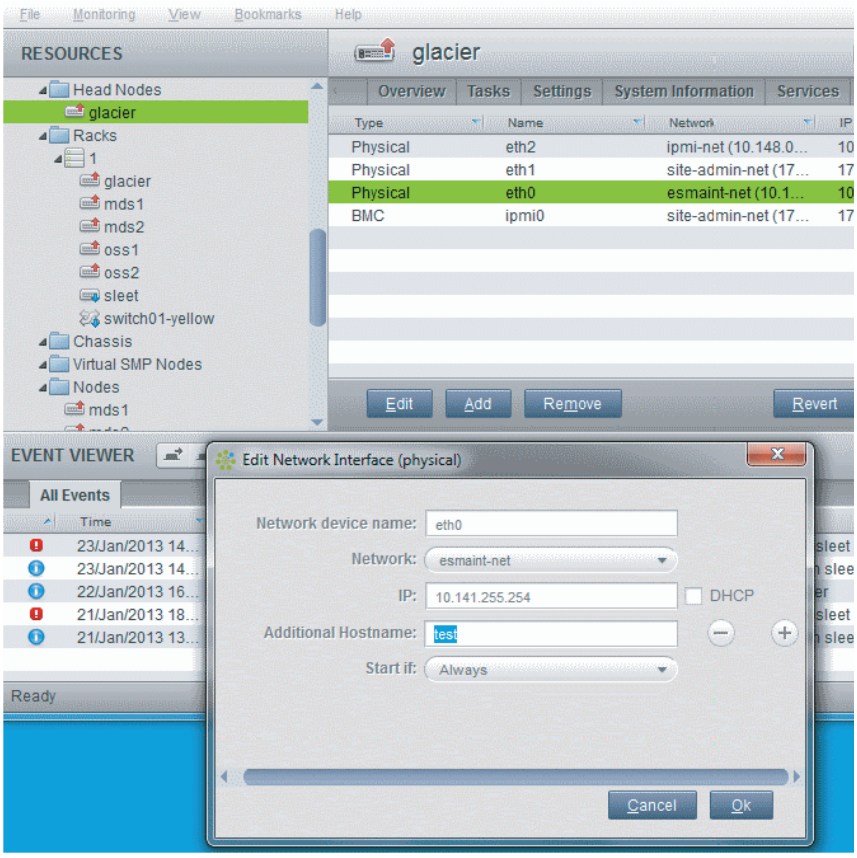
3.11.5 Add Hostname to an Internal Network

Hostname can be added as name/value pairs to the `/etc/hosts` file(s) within the system, but it is recommended to let Bright manage hostname resolution for devices on the `esmaint-net` through its DNS server on the `esmaint-net` interface.

Multiple hostnames can be added as space-separated entries. The `named` service automatically restarts within about 20 seconds after committal, implementing the configuration changes. The system restarts automatically when there are changes made to service configurations by `cmgui` or `cmsh`.

The `cmgui` can be used to add a hostname to a network by selecting the CIMS head node (`glacier`) in the resource tree, then the **Networks** tab, and physical device for `eth0` and the `esmaint-net`, and clicking **Edit**. The CIMS node in the following figure is named `glacier`.

Figure 26. Add a Hostname to an Internal Network



In `cmsh`, the hostnames can be added to the `additionalhostnames` object, from within `interfaces` submode for the CIMS node. The CIMS node is `esms1` in this example. The `interfaces` submode is accessible from the `device` mode. The CIMS `eth0` interface for `esmaint-net` is assigned a hostname `test` in this example. The `!` character can be used to invoke Linux commands such as `ping`, when in `cmsh`.

Example 14. Add hostnames to an internal network using cmsh

```
glacier# cmsh
[glacier]% device use glacier
[glacier->device[glacier]]% interfaces
[glacier->device[glacier]->interfaces]% list

Type         Network device name  IP               Network
------------ -------------------- ---------------- ----------------
bmc          ipmi0                aaa.bbb.ccc.ddd  site-admin-net
physical     eth0 [prov]          10.141.255.254   esmaint-net
physical     eth1                 aaa.bbb.ccc.ddd  site-admin-net
physical     eth2                 10.148.255.254   ipmi-net

[glacier->device[glacier]->interfaces]% use eth0
[glacier->device[glacier]->interfaces[eth0]]% set additionalhostnames test
[glacier->device*[glacier*]->interfaces*[eth0*]]% commit
[glacier->device[glacier]->interfaces[eth0]]%
Tue Jan 22 16:40:29 2013 [notice] glacier: Service named was restarted
[glacier->device[glacier]->interfaces[eth0]]% !ping test
PING test.cm.cluster (10.141.255.254) 56(84) bytes of data.
64 bytes from glacier.cm.cluster (10.141.255.254): icmp_seq=1 ttl=64 time=0.038 ms
64 bytes from glacier.cm.cluster (10.141.255.254): icmp_seq=2 ttl=64 time=0.033 ms
```

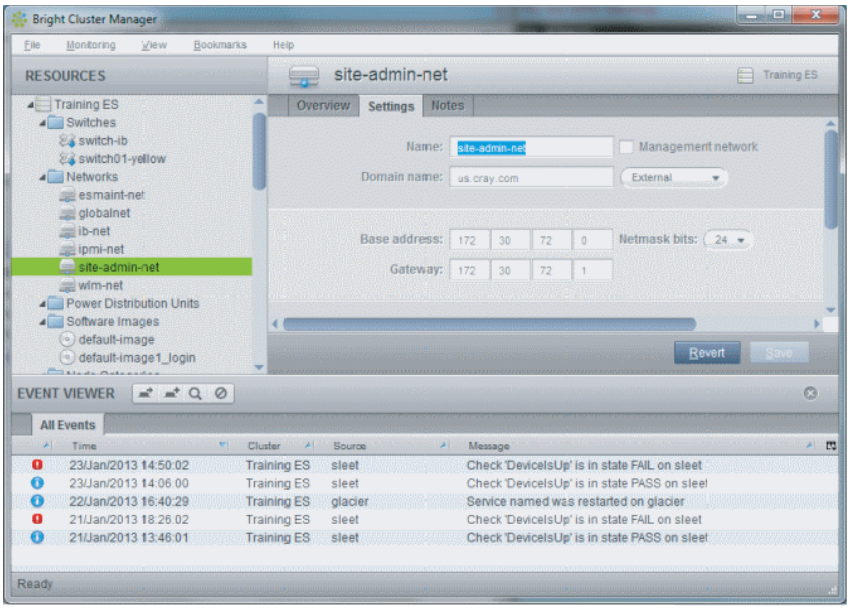
3.11.6 Change External Network Parameters for the System

Changing the network parameters of a DMP system (apart from the IP address of the system) requires making changes to the external network object (`site-admin-net`, `site-user-net`), and the system object network settings.

3.11.6.1 Change the External Network Object Settings

External network objects (`site-admin-net` or `site-user-net`) contain the network settings to enable connections to the external network, for example, a head node. Network settings are configured in the **Settings** tab of the **Networks** resource of `cmgui`.

Figure 27. Change External Network Object Settings



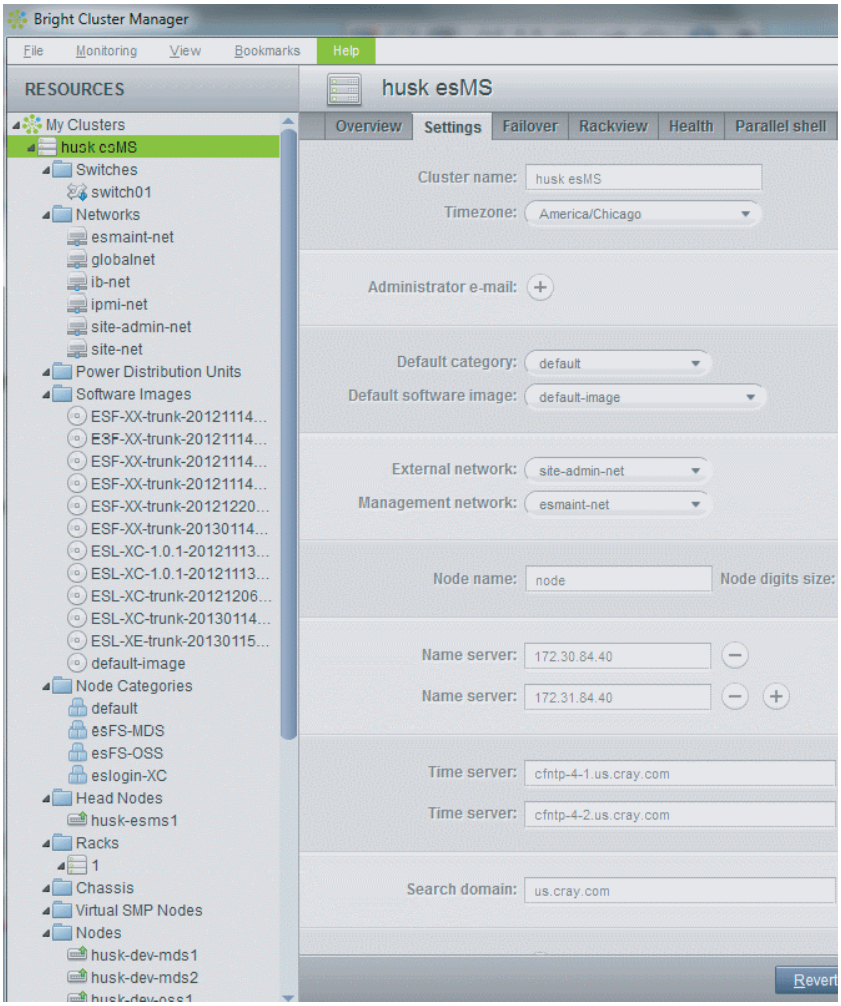
The following external network parameters can be configured:

- IP network parameters of the system (but not the IP address of the system):
  - **Base address**: the IP address of the external network. This is not to be confused with the IP address of the system.
  - **Netmask bits**: the netmask size, or prefix-length, of the external network, in bits.
  - **Gateway**: the default route for the external network.
  - **Dynamic range start** and **Dynamic range end**: Not used by the external network configuration.
- **Domain name**: the network domain (LAN domain, i.e. what domain machines on the external network use as their domain)
- **Name**: the network name such as, `site-admin-net`, `site-user-net`, `site-net`)
- The External network checkbox: this is checked for a Type 1 cluster (nodes are connected on a private internal network)
- **MTU**: size (the maximum value for a TCP/IP packet before it fragments on the external network the default value is 1500)

3.11.6.2 Change Network Settings for the CIMS

The CIMS (head node object) contains other network settings used to connect to the outside. These are configured in the **Settings** tab of the head node object resource in `cmgui`. These settings are e-mail address(es) for the administrator, the external name servers used by the system to resolve external hostnames, the DNS search domain (what the cluster uses as its domain), and NTP time servers (used to synchronize the time on the system with standard time) and time zone settings.

Figure 28. Change Network Settings for the CIMS



The static IP address of the head node can also be changed using `cmsh` in the `base` object under `partition` mode.

Example 15. Change the network settings for the CIMS

```
esms1# cmsh
[esms1]% network use site-admin-net
[esms1->network[site-admin-net]]% set baseaddress 192.168.1.0
[esms1->network*[site-admin-net*]]% set netmaskbits 24
[esms1->network*[site-admin-net*]]% set gateway 192.168.1.1
[esms1->network*[site-admin-net*]]% commit
[esms1->network[site-admin-net]]% partition use base
[esms1->partition[base]]% set nameservers 192.168.1.1
[esms1->partition*[base*]]% set searchdomains searchdomain1.com searchdomain2.com
[esms1->partition*[base*]]% append timeservers ntp.timeserver1.com ntp.timeserver2.com
[esms1->partition*[base*]]% commit
[esms1->partition[base]]% device use esms1
[esms1->device[esms1]]% interfaces
[esms1->device[esms1]->interfaces]% use eth1
[esms1->device[esms1]->interfaces[eth1]]% set ip 192.168.1.176
[esms1->device[esms1]->interfaces*[eth1*]]% commit
[esms1->device[esms1]->interfaces[eth1]]% exit; exit;
[esms1->device]% reboot
```

Reboot the CIMS node to activate the changes.

3.11.7 Use DHCP to Supply Network Values for the External Interface

Connecting the DMP system via DHCP on the external network is not recommended. This is because DHCP-related issues can complicate network troubleshooting when compared with using static assignments.

3.11.8 Change Ethernet Interface Speed Settings

The `interfaces` health check in Bright displays `FAIL` if an interface on the CIMS node connects to a device that operates at a slower speed. For example a switch that operates at 100Mb/s. To correct the problem, adjust the CIMS node interface speed setting.

Procedure 26. Change Ethernet interface speed settings

1. Log in to the CIMS node as `root` and start `cmsh`.

```
esms1# cmsh
[esms1]%
```

2. Switch to device mode and select the CIMS node. The switch to the `interfaces` sub mode and list the interfaces.

```
[esms1]% device use esms1
[esms1->device[esms1]]% interfaces
[esms1->device[esms1]->interfaces]% list
Type         Network device name  IP               Network
------------ -------------------- ---------------- ----------------
bmc          ipmi0                172.30.72.49     site-admin-net
physical     eth0 [prov]          10.141.255.254   esmaint-net
physical     eth1                 172.30.72.46     site-admin-net
physical     eth2                 10.148.255.254   ipmi-net
```

3. Select `eth0` and show its settings.

```
[esms1->device[esms1]->interfaces]% use eth0
[esms1->device[esms1]->interfaces[eth0]]% show
Parameter                     Value
----------------------------- --------------------
Additional Hostnames
Card Type                     Ethernet
DHCP                          no
IP                            10.141.255.254
MAC                           00:00:00:00:00:00
Network                       esmaint-net
Network device name           eth0 [prov]
Revision
Speed
Start if                      ALWAYS
Type                          physical
```

4. Set the `speed` setting for the `eth0` interface to `100Mb/s` and commit the change.

```
[esms1->device[esms1]->interfaces[eth0]]% set speed 100Mb/s
[esms1->device*[esms1*]->interfaces*[eth0*]]% commit
```

3.12 Set Up Exclude Lists

Exclude lists may be configured for the `esLogin-XC`, `esLogin-XE`, `esfs-odd-`*filesystem*, `esfs-even-`*filesystem*, and `esfs-failed-`*filesystem*categories. Software images are synchronized by either pushing files from software image on the CIMS node to the slave node, or pulling files from the slave node to the software image on the CIMS node.

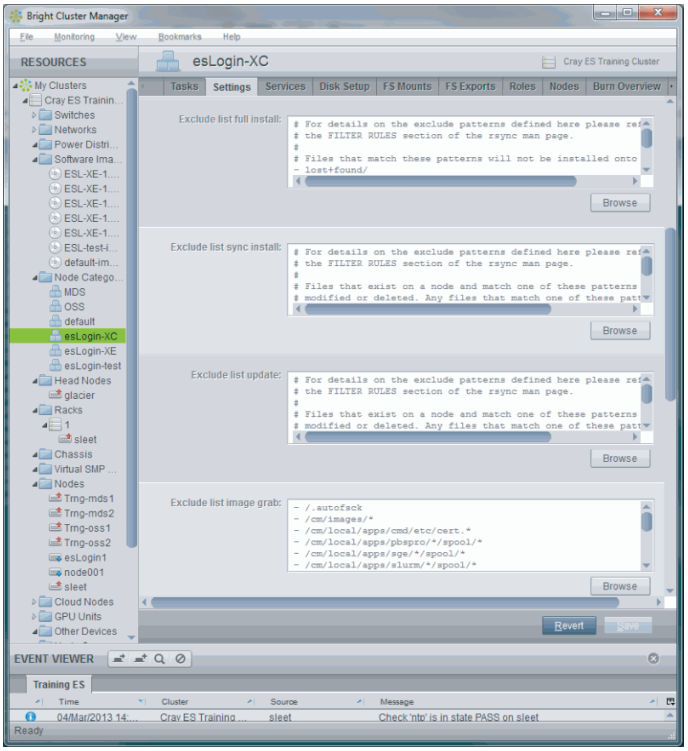
Three exclude lists control which files are pushed from the CIMS node to the slave node. These are: `excludelistfullinstall`, `excludelistsyncinstall`, and `excludelistupdate`. These lists contain files that are **not** pushed to the slave node during software image installation.

Two exclude lists control how files are pulled from the slave node to the software image on the CIMS which are `exludelistgrab`, `excludelistgrabnew`.

Files created or modified by a finalize script must be listed in the `excludelistupdate` exclude list for the category. Software updates will overwrite customized files if the files are not specified in an exclude list for the category. Customized files must also be specified in the `excludelistgrab`, and `excludelistgrabnew` exclude lists to prevent customized files from being copied to the CIMS. Be sure to configure the `excludelistgrab` and `excludelistgrabnew` exclude lists to exclude all network file systems such as NFS®, Lustre®, or GPFS™ file systems.

The figure shows the category exclude lists under the `cmgui` **Node Category->Settings** tab.

Figure 29. Setting up Exclude Lists in `cmgui`



Refer to the *Bright Cluster Manager 6.1 Administrator Manual* PDF file on the CIMS node for more information about exclude lists.

3.12.1 Check Exclude Lists

Each of the exclude lists has specific comments about the preconfigured exclusions. To check what is currently in one of the preconfigured exclude lists, run the following `cmsh` command or use the `cmgui` to select a node category from the resource tree, then select the **Settings** tab.

```
esms1# cmsh -c "category use esLogin-XC; get excludelistfullinstall"
# For details on the exclude patterns defined here please refer to
# the FILTER RULES section of the rsync man page.
#
# Files that match these patterns will not be installed onto the node.
- lost+found/
- /proc/*
- /sys/*
```

3.12.2 Changing Exclude Lists

To change an exclude list, run this command. It will open an editor so you can make changes to the list.

```
esms1# cmsh -c "category use esLogin-XC; set excludelistfullinstall; commit"
```

3.12.3 Exclude User Home Directories

If user home directories in `/home/users` are mounted from an NFS server, then add `/home/users/*` to `excludelistfullinstall`, `excludelistsyncinstall`, and `excludelistupdate`, to prevent those directories from being removed when synchronizing from a software image to a CDL node.

If user home directories in `/home/users` are mounted from an NFS server and also mount a Lustre file system in `/lus/scratch`, then add `/home/users/*` and `/lus/scratch` to the `excludelistgrab` and `excludelistgrabnew` exclude lists. This prevents a `grabimage` command from copying all of the files from a remote file server to the software image.

```
/home/users/*
/lus/scratch
```

3.12.4 Exclude List Defaults

Default exclude lists are configured for the `esLogin-XC`, `esLogin-XE`, `esFS-MDS` and `esFS-OSS` categories in Bright when ESL or ESF software is installed. During an `imageupdate` command, the synchronization process uses the `excludelistupdate` list, which is a list of files and directories. One of the cross checking actions that may run during the synchronization is that the items on the list are excluded when copying parts of the file system from a known good software image to the node.

Make sure the text `no-new-files:` is prepended to each entry in the `excludelistsyncinstall` and `excludelistupdate`.

The default exclude lists follow:

`excludelistfullinstall` — Push

When the full software image installation occurs at boot time, **all** files from the software image in the CIMS are pushed to the slave node unless they are included in the `excludelistfullinstall` exclude list.

Files that match these patterns are not installed onto the slave node.

```
- lost+found/
- /proc/*
- /sys/*
```

`excludelistsyncinstall` — Push

When software image is synchronized at boot time, **all** files from the software image on the CIMS are pushed to the slave node unless they are entered in the `excludelistsyncinstall` exclude list.

Any files that match one of these patterns and that exist in the image but are absent on the node, will be copied to the node. Files that exist on a node and match one of these default patterns are not modified or deleted.

If necessary, add `no-new-files:` to every line in `excludelistsyncinstall` to exclude files that are needed for a full install but are troublesome for an `imageupdate`.

```
- /cm/local/apps/pbspro/*/spool/aux/*
- /cm/local/apps/pbspro/*/spool/checkpoint/*
- /cm/local/apps/pbspro/*/spool/mom_logs/*
- /cm/local/apps/pbspro/*/spool/spool/*
- /cm/local/apps/pbspro/*/spool/undelivered/*
- /cm/local/apps/pbspro/*/spool/mom_priv/hooks/tmp/*
- /cm/local/apps/*/var/prologs
<- /home/*
no-new-files: - /dev/*
no-new-files: - /tftpboot/*
no-new-files: - /var/lock/*
no-new-files: - /var/run/*
no-new-files: - /.autorelabel
```

`excludelistupdate` — Push

If the node is already booted, running the `cmsh imageupdate` command pushes all files from the slave node software image on the CIMS to the software image **on the running node**, except those entered on the `excludelistupdate` exclude list.

The `exludelistupdate` list is in the form of two sublists. Both sublists are lists of paths, except that the second sublist is prefixed with the text `no new files:`. When a node is updated, all of its files are examined during `imageupdate` synchronization. The logic used is as follows:

Files that exist on a node and match one of the patterns below will not be modified or deleted. Any files that match one of these default patterns and that exist in the image but are absent on the node are copied to the node.

If an excluded path from `excludelistupdate` exists on the node, then no files from that path are copied over from the software image to the node.

If an excluded path from `excludelistupdate` does not exist on the node, then:

- if the path is on the first, non-prefixed list, then the path is copied over from the software image to the node.

- if the path is on the second, prefixed list, then the path is not copied over from the software image to the node. That is, no new files are copied over, like the prefix text implies.

To work around this logic, prepend `no-new-files:` to each line. This prevents new files and paths being created on the nodes.

If necessary, add `no-new-files:` to every line in `excludelistupdate` to exclude files that are needed for a full install but are troublesome for a imageupdate.

```
- /boot/boot
- /boot/initrd-*.orig
- /cm/local/apps/pbspro/*/spool/aux/*
- /cm/local/apps/pbspro/*/spool/checkpoint/*
- /cm/local/apps/pbspro/*/spool/mom_logs/*
- /cm/local/apps/pbspro/*/spool/spool/*
- /cm/local/apps/pbspro/*/spool/undelivered/*
- /cm/local/apps/pbspro/*/spool/mom_priv/hooks/tmp/*
- /cm/local/apps/pbspro/*/spool/*
- /cm/local/apps/*/var/prologs
- /etc/adjtime*
- /etc/blkid/*
- /etc/grub.conf
- /etc/lvm/.cache
- /etc/openvpn
- /etc/postfix/*.db
- /etc/rc.d/rc*.d/*nfsserver
- /etc/reader.conf.d/reader.conf
- /etc/reader.conf
- /etc/sysconfig/network/config
- /etc/sysconfig/network/ifcfg-*
- /etc/sysconfig/network/routes
- /var/adm/netconfig/*
- /var/cache/hald/*
- /var/lib/ntp/*
- /var/lib/misc/random-seed
- /var/lib/postfix/master.lock
- /var/lib/smartmontools/*
- /var/net-snmp/mib_indexes/*
- /cm/local/apps/intel-mic/*/filesystem
- /etc/ofed-mic.map
```

```
- /etc/libibverbs.d/ibscif.driver
- /etc/rc.d/init.d/ofed-mic
- /etc/rc.d/*/*ofed-mic
- /etc/sysconfig/mic/mic*.conf
- /etc/udev/rules.d/*-udev-scif.rules
- /lib/modules/*/extra/mic.ko
- /lib/modules/*/updates
- /sbin/sysctl_perf_tuning
- /sbin/connectx_port_config
- /usr/bin/dev_test
- /usr/lib64/libibscif-rdmav2.so
- /usr/sbin/ibscif-opt
- /usr/src/kernels/*/Module.symvers.mic
- /usr/src/kernels/*/include/scif.h
- /var/cache/sysctl_perf_tuning
- /usr/sbin/ibpd
- /etc/infiniband
- /usr/bin/ibdev2netdev
- /etc/modprobe.d/mlx4_en.conf
- /etc/modprobe.d/ib*.conf
- /etc/rc.d/*/*openibd
- /etc/udev/rules.d/*-ibpd.rules
- /etc/udev/rules.d/*-ib.rules
- /etc/udev/rules.d/*-persistent-net.rules
- /etc/udev/rules.d/*-persistent-cd.rules
no-new-files: - /cgroup/*
no-new-files: - /cm/node-installer-ebs
no-new-files: - /media/*
no-new-files: - /var/lib/ldap/*
no-new-files: - /var/lib/rpm/__db.*
no-new-files: - /var/run/*
no-new-files: - /.autorelabel
```

`excludelistgrab` — Pull

Using the `cmsh grabimage` command, or `cmgui` **Grab to Image** button synchronizes files **from** the slave node **to** its existing software image, unless the files or directories are entered in the `excludelistgrab` exclude list. The default list follows:

```
- /boot/grub/device.map
- /boot/grub/grub.conf
- /boot/grub/menu.lst
- /cgroup/*
- /cm/local/apps/pbspro/*/spool/aux/*
- /cm/local/apps/pbspro/*/spool/checkpoint/*
- /cm/local/apps/pbspro/*/spool/mom_logs/*
- /cm/local/apps/pbspro/*/spool/spool/*
- /cm/local/apps/pbspro/*/spool/undelivered/*
- /cm/local/apps/pbspro/*/spool/mom_priv/hooks/tmp/*
- /cm/local/apps/*/var/prologs
- /etc/exports
- /etc/fstab
- /etc/lvm/cache/*
- /etc/mtab
- /etc/postfix/*
- /etc/ntp.conf
- /etc/ntp/*
```

```
- /etc/openvpn/*
- /etc/sysconfig/network
- /etc/sysconfig/network/ifcfg-eth*
- /etc/sysconfig/network/ifcfg-ib*
- /etc/sysconfig/network/ifcfg-br*
- /etc/sysconfig/network/ifcfg-bond*
- /etc/sysconfig/routes
- /var/adm/netconfig/*
- /var/cache/zypp/*
- /var/lib/dhcp/*
- /var/lib/ldap/*
- /var/lib/nfs/*
- /var/lib/rpm/__db.*
- /var/run/*.pid
- /var/run/*/*.pid
```

`excludelistgrabnew` — Pull

Running the `cmsh grabimage -n` *newimage* command synchronizes files from the slave node to a new software image, unless the files or directories are entered in the `excludelistgrabnew` exclude list. The default list follows.

```
- /cgroup/*
- /cm/local/apps/pbspro/*/spool/aux/*
- /cm/local/apps/pbspro/*/spool/checkpoint/*
- /cm/local/apps/pbspro/*/spool/mom_logs/*
- /cm/local/apps/pbspro/*/spool/spool/*
- /cm/local/apps/pbspro/*/spool/undelivered/*
- /cm/local/apps/pbspro/*/spool/mom_priv/hooks/tmp/*
- /cm/local/apps/*/var/prologs
- /etc/mtab
- /etc/openvpn/*
- /etc/sysconfig/network
- /etc/sysconfig/network/ifcfg-eth*
- /etc/sysconfig/network/ifcfg-ib*
- /etc/sysconfig/network/ifcfg-br*
- /etc/sysconfig/network/ifcfg-bond*
- /etc/sysconfig/routes
- /media/*
- /var/lib/dhcp/*
```

3.13 Clone a Production Slave Node Software Image

Avoid corrupting production slave node software images. Always clone the production software image to a test image and make modifications or install updates on the test software image.

If a slave node has changes that have not been applied to the software image on the CIMS, use the `cmsh grabimage` command to save the image on the node to an existing image on the CIMS node. See *Data Management Platform (DMP) Administrator's Guide* (S–2327) or the *Bright Cluster Manager 6.1 Administrator Manual* for more information about how to grab an image from a running node.

Procedure 27. Clone a slave node software image

This procedure clones a slave node software image (`ESL-XC-2.0.0`) to a new software image named `ESL-XC-2.2.0test`.

1. Log into the CIMS as `root` and run the `cmsh` command.

```
remote% ssh root@esms1
esms1# cmsh
[esms1]%
```

2. Enter `softwareimage` mode.

```
[esms1]% softwareimage
[esms1->softwareimage]%
```

3. List the available images. This example shows three images: `ESL-XC-2.0.0`, `ESL-XE-2.0.0`, and `default-image`.

```
[esms1->softwareimage]% list
Name (key)                 Path                              Kernel version
-------------------------- --------------------------------  ------------------
ESL-XC-2.0.0               /cm/images/ESL-XC-2.0.0           3.0.93-0.8-default
ESL-XE-2.0.0               /cm/images/ESL-XE-2.0.0           2.6.32.59-0.7-default
default-image              /cm/images/default-image          3.0.80-0.5-default
```

4. Create a new software image named `ESL-XC-2.2.0test` by cloning `ESL-XC-2.0.0`.

   Important: The time to clone a software image using Bright depends on the image size. Cloning a minimal image (operating system only) completes in 5 to 10 minutes. A fully configured image with the Cray development environment (including CDT or CADE software) can require between 30 to 75 minutes to complete. The Bright clone operation spawns a background process that does not prevent you from rebooting a node or performing other configuration changes to software images before the image is fully cloned. Cray recommends that you copy images from the UNIX prompt, wait for the prompt to return, then clone the image within Bright before you proceed.

```
[esms1->softwareimage]% quit
esms1# cp -pr /cm/images/ESL-XC-2.0.0 /cm/images/ESL-XC-2.2.0test
esms1# cmsh
esms1% softwareimage
[esms1->softwareimage]% clone ESL-XC-2.0.0 ESL-XC-2.2.0test
[esms1->softwareimage*[ESL-XC-2.2.0test*]]% commit
```

5. Check the list of images.

```
[esms1->softwareimage[ESL-XC-2.2.0test]]% list
Name (key)                Path                              Kernel version
------------------------  --------------------------------  ----------------------
ESL-XC-2.0.0              /cm/images/ESL-XC-2.0.0           3.0.93-0.8-default
ESL-XC-2.2.0test          /cm/images/ESL-XC-2.2.0test       3.0.93-0.8-default
ESL-XE-2.0.0              /cm/images/ESL-XE-2.0.0           2.6.32.59-0.7-default
default-image             /cm/images/default-image          3.0.80-0.5-default
```

6. Exit `cmsh`.

```
[esms1->softwareimage[ESL-XC-2.2.0test]]% quit
esms1#
```

3.14 Isolate a Slave Node for Testing

This procedure describes how to isolate a slave node (in this example, `eslogin1`) for testing.

Procedure 28. Isolating slave node for testing

1. Log in to the CIMS node as `root`.
2. Use Bright `cmsh` to create a test image.
   a. Copy (clone) the current working software image. Choose a unique name to identify the new test image. This example clones the image name `ESL-XC-2.2.0` to `ESL-test-image`. Copy the image from the UNIX® prompt and wait for the copy to complete before cloning the image in Bright. This ensures the clone operation is complete before you continue.

```
esms1# cp -pr /cm/images/ESL-XC-2.2.0 /cm/images/ESL-test-image

esms1# cmsh
[esms1]% softwareimage
[esms1->softwareimage]% list
Name (key)            Path                               Kernel version
--------------------- ---------------------------------- -----------------
default-image           /cm/images/default-image           3.0.93-0.8-default
ESL-XC-2.2.0            /cm/images/ESL-XC-2.2.0            3.0.93-0.8-default

[esms1->softwareimage]% clone ESL-XC-2.2.0  ESL-test-image
[esms1->softwareimage*[ESL-test-image*]]% commit
```

   b. Create a test category from your default slave node category (in this example, `esLogin-XC`) and assign the cloned image to that category.

```
[esms1->softwareimage[ESL-test-image]]% category
[esms1->category]% clone esLogin-XC esLogin-test
[esms1->category*[esLogin-test*]]% set softwareimage ESL-test-image
[esms1->category*[esLogin-test*]]% commit
```

c. Temporarily assign a CDL node (in this example, `eslogin1`) to the `esLogin-test` category.

```
[esms1->category[esLogin-test]]% device
[esms1->device]% use eslogin1
[esms1->device[eslogin1]]% set category esLogin-test
[esms1->device*[eslogin1*]]% commit
[esms1->device[eslogin1]]% category
[esms1->category]% list
Name (key)               Software image
------------------------ ------------------------
default                  default-image
esLogin-XC               ESL-XC-2.2.0
esLogin-XE               ESL-XE-2.1.0
esLogin-test             ESL-test-image

[esms1->category]% usedby esLogin-test
Category used by the following:
Type             Name                     Parameter                Autochange
---------------- ------------------------ ------------------------ ------------
Device           eslogin1                 category                 no
```

d. Open a new shell window and log in to the CIMS node as `root`.

e. Start `cmsh`, and launch a remote console (`rconsole`) on the CDL node.

```
esms1# cmsh
[esms1]% device use eslogin1
[esms1->device[eslogin1]]% rconsole
```

f. In a separate CIMS window, login as `root` and reboot the slave node (`eslogin1` in the example) using `cmsh` or use the **Reboot** button from the `cmgui`.

```
esms1# cmsh
[esms1]% device use eslogin1
[esms1->device[eslogin1]]% reboot
```

3. Verify the node boots without errors before you begin your testing.

4. Install and configure the new software on the `ESL-test-image` and routinely test boot the node to verify proper operation.

5. After you have created the new software image, move the slave node out of the `esLogin-test` category, back into the default CDL category (in this example, `esLogin-XC`).

```
esms1# cmsh
[esms1]% device use eslogin1
[esms1->device[eslogin1]]% set category esLogin-XC
[esms1->device*[eslogin1*]]% commit
[esms1->device[eslogin1]]% list
Type           Hostname (key)   MAC                Category     Ip          Network
-------------- ---------------- ------------------ ------------ ----------- ---------
...
PhysicalNode    eslogin1         00:00:00:00:00:00  esLogin-XC   10.141.0.2  esmaint-net
```

6. Clone the new test image (`ESL-test-image`) into the new working CDL

software image (in this example, `ESL-XC-2.2.0_CLE 5.2`). Copy the image from the UNIX® prompt and wait for the copy to complete before cloning the image in Bright. This ensures the clone operation is complete before you continue.

```
[esms1->device[eslogin1]]% quit
esms1# cp -pr /cm/images/ESL-test-image /cm/images/ESL-XC-2.2.0_CLE 5.2

esms1# cmsh
[esms1]% softwareimage clone ESL-test-image ESL-XC-2.2.0_CLE 5.2
[esms1->softwareimage*[ESL-XC-2.2.0_CLE 5.2*]]% commit
[esms1->softwareimage[ESL-XC-2.2.0_CLE 5.2]]% list
Name (key)                    Path                                      Kernel version
----------------------------- ----------------------------------------- ----------------------
ESL-XE-2.1.0                  /cm/images/ESL-XE-2.1.0                   2.6.32.59-0.7-default
ESL-XC-2.2.0_CLE 5.2           /cm/images/ESL-XC-2.2.0_CLE 5.2           3.0.93-0.8-default
ESL-test-image                /cm/images/ESL-test-image                 3.0.93-0.8-default
default-image                 /cm/images/default-image                  3.0.80-0.5-default
```

7. Use Bright to assign the new default CDL software (`ESL-XC-2.2.0_CLE 5.2`) to the default CDL category (`esLogin-XC`).

```
[esms1->softwareimage[ESL-XC-2.2.0_CLE 5.2]]% category
[esms1->category]% use esLogin-XE
[esms1->category*[esLogin-XC*]]% set softwareimage ESL-XC-2.2.0_CLE 5.2
[esms1->category*[esLogin-XC*]]% commit
```

8. Reboot all of slave nodes in the `esLogin-XC` category with the new image.

```
[esms1->category[esLogin-XC]]% device reboot -c esLogin-XC
```

3.15 Create a CDL Node Group

Optional: Node groups can simplify and automate administration tasks by allowing management operations to be performed on groups of nodes. It is not necessary to configure node groups to manage the system.

Nodes may belong to several groups at the same time. There are no parameters associated with a node group other than the member nodes.

For sites with multiple CDL nodes, Cray recommends creating a node group for these nodes.

Procedure 29. Create a CDL node group

1. Log in to the CIMS as `root` and start `cmsh`.

```
esms1# cmsh
```

2. Switch to `nodegroup` mode:

```
[esms1]% nodegroup
[esms1->nodegroup]%
```

3. Use the `add` command to add a node group. This example creates a new node group called `Login`.

```
[esms1->nodegroup]% add Login
[esms1->nodegroup*[Login*]]%
```

4. Use the `append` command to add nodes to the group. Multiple nodes can be added as a list (*N*) or a range (`node1..node`*N*).

```
[esms1->nodegroup*[Login*]]% append nodes eslogin1..eslogin5
```

5. Commit your changes.

```
[esms1->nodegroup*[Login*]]% commit
[esms1->nodegroup[Login]]%
```

6. Exit `cmsh`.

```
[esms1->nodegroup[Login]]% quit
esms1#
```

3.16 Add a Managed Switch or Device to the Bright Configuration

You can include Ethernet, InfiniBand® (IB), Fibre Channel (FC), or serial-attached SCSI (SAS) switches, RAID controllers, or intelligent PDU to the Bright configuration. Refer to the device documentation supplied by the manufacturer for configuration and setup procedures. Bright uses SNMP community strings to communicate to devices. SNMP must be enabled for the device and the SNMP community strings should be configured correctly. By default, the SNMP community strings for switches and PDUs are typically set to public and private for respectively read and write access. This example shows how to configure SNMP community strings for an Ethernet switch using `cmsh`.

Example 16. Change SNMP community strings for devices

```
[esms1]% device use switch1-esmaint-net
[esms1->device[switch1-esmaint-net]]% get readstring
public
[esms1->device[switch1-esmaint-net]]% get writestring
private
[esms1->device[switch1-esmaint-net]]% set readstring public2
[esms1->device*[switch1-esmaint-net*]]% set writestring private2
[esms1->device*[switch1-esmaint-net*]]% commit
```

The following procedure describes how to setup a Mellanox IS50xx series IB switch and configure Bright to manage it. Refer to Figure 3 for the standard IP addressing scheme used on the `esmaint-net` network for switches or other devices.

Most device command-line interfaces (CLIs) have built-in help systems that can be displayed by entering **`?`** on the command line. Some switches also support a context-sensitive help system that displays valid commands or command options when pressing the `Tab` key.

Uplink ports (switch ports that are connected to other switches or to the `esmaint-net`) must be configured in Bright.

Procedure 30. Add a Mellanox IS50XX series switch to the Bright configuration

1. Connect the serial management port on the switch to a laptop or PC running terminal emulator software (e.g., minicom, Putty, etc.). Settings are typically 9600 Baud, 8N1, no flow control, and VT100 emulation.
2. Press return in emulator software console window to display the console prompt. It may be necessary to power cycle the switch to reset the console.
3. Do **not** use the setup wizard. Enter `Ctrl-z` to exit the wizard, if it starts.
4. Enter the login and password (refer to the switch documentation for default login and password).

```
Mellanox FabricIT Switch Management
switch-5e0120 login: admin
Password: admin
Last login: Mon Aug 20 12:55:57 on ttyS0
Mellanox Switch
```

5. Type a question mark **?** on the command line to display valid commands from the current mode.

```
switch-5e0120 [standalone: master] > ?
cli              Configure CLI shell options
enable           Enter enable mode
exit             Log out of the CLI
fabric           Manage fabric diagnostics
help             View description of the interactive help system
no               Negate or clear certain configuration options
ping             Send ICMP echo requests to a specified host
show             Display system configuration or statistics
slogin           Log into another system securely using ssh
telnet           Log into another system using telnet
terminal         Set terminal parameters
test             Diagnostics
traceroute       Trace the route packets take to a destination
ib-switch-1 [standalone: master] >
```

6. At the console prompt, run the following commands to enable switch configuration from a terminal:

```
switch-5e0120 [standalone: master]# enable
switch-5e0120 [standalone: master]# configure terminal
```

7. Set up simple network management protocol (SNMP).

```
conswitch-5e0120 [standalone: master] (config)# snmp-server community public
```

8. Set the IP address and netmask of the Ethernet port used to connect to the `esmaint-net` network (in this example `eth0`).

```
switch-5e0120 [standalone: master] (config)# interface eth0 ip address
10.141.200.1 255.255.255.0
```

9. Set a hostname for the switch.

```
switch-5e0120 [standalone: master] (config)# hostname ib-switch-1
ib-switch1 [standalone: master] (config) #
```

10. Write the configuration to memory and exit.

```
ib-switch1 [standalone: master] (config) # write memory
ib-switch1 [standalone: master] (config) #
ib-switch-1 [standalone: master] (config) # exit
ib-switch-1 [standalone: master] # exit

Mellanox FabricIT Switch Management

ib-switch-1 login:
```

11. Log in to the CIMS as `root` and start `cmsh`.

```
esms1# cmsh
[esms1]%
```

12. From `device` mode, add the IB switch to Bright. Your options for device type are: `cloudnode`, `physicalnode`, `virtualsmpnode`, `headnode`, `ethernetswitch`, `ibswitch`, `myrinetswitch`, `powerdistributionunit`, `genericdevice`, `racksensor`, `chassis`, `gpuunit`. Use the hostname that you configured for the switch in step 9.

```
[esms1]% device add ibswitch ib-switch-1
[esms1->device*[ib-switch-1*]]%
```

13. Set the management network to `esmaint-net`.

```
[esms1->device*[ib-switch-1*]]% set network esmaint-net
```

14. Set the IP address for interface configured in step 8.

```
[esms1->device*[ib-switch-1*]]% set ip 10.141.200.1
```

15. Configure the SNMP read string to public and write string to private.

```
[esms1->device*[ib-switch-1*]]% set readstring public
[esms1->device*[ib-switch-1*]]% set writestring private
```

16. (Optional) Set uplink ports.

```
[esms1->device*[ib-switch-1*]]% set uplinks uplinkport
```

17. (Optional) Use the `cmsh set` command to set other switch parameters such as `rack` ID, `deviceheight` (1U), `deviceposition` in rack, `mac` address, hardware `tag`, and administrator `notes`. All of these settings can be configured using the `cmgui` after the switch configured in Bright.

18. Commit the changes and list the devices managed by Bright.

```
[esms1->device*[ib-switch-1*]]% commit
[esms1->device[ib-switch-1]]% list
esms1#
```

```
[esms1->device[ib-switch-1]]% quit
esms1#Type                  Hostname (key)      MAC                Category        Ip              Network
--------------------- ------------------ ----------------- ---------------- --------------- ------------
EthernetSwitch         esmaint-net-switch  00:0F:8F:8E:9D:C0                   10.141.50.1     esmaint-net
EthernetSwitch         ipmi-net-switch     00:0B:5F:CE:2F:40                   10.148.50.1     ipmi-net
EthernetSwitch         wlm-net-switch      00:00:00:00:00:00                   10.128.100.1    wlm-net
HeadNode               esms1               78:2B:CB:40:CE:CA                   10.141.255.254  esmaint-net
IBSwitch               ib-switch-1         00:00:00:00:00:00                   10.141.200.1    esmaint-net
PhysicalNode           eslogin-001         84:2B:2B:61:B0:04  esLogin-XC       10.141.0.37     esmaint-net
PhysicalNode           mds001              78:2B:CB:50:E9:A3  -mds         10.141.0.10     esmaint-net
[esms1->device[ib-switch-1]]%
```

19. Exit `cmsh`.

```
[esms1->device*[ib-switch-1*]]% quit
esms1#
```

3.17 Configure the DELL™ 5548 1GbE Switch

The CIMS system should include configuration settings for Ethernet switches so that they can be monitored by Bright. Use this procedure to change switch settings for a Dell 5548 switch if your system is not preconfigured, or if you need to reconfigure another Ethernet switch.

For the VLAN port assignments, see Figure 2.

Procedure 31. Configure the Dell 5548 1GbE switch

This procedure shows the instructions for a 48-port Ethernet switch. For a 24-port Ethernet switch, use the VLAN port rules and example commands to adapt the configuration for a smaller switch.

1. Connect the serial port of the switch to a suitable VT100 emulator (minicom, Putty, etc.). Settings are 9600 Baud, 8N1, no flow control, VT100 emulation.
2. Power on the switch.
3. Do **not** use the setup wizard. Enter `Ctrl-z` to exit the wizard, if it starts.
4. At the console prompt, run the following commands:

```
console> enable
console# config
```

5. Set up SNMP.

```
console (config)# snmp-server community public
```

6. Set up the VLANs.

```
console (config)# vlan database
console (config-vlan)# vlan 2
console (config-vlan)# vlan 3
console (config-vlan)# vlan 4
console (config-vlan)# exit
console (config)# interface vlan 1
console (config-if)# name esmaint-net
console (config-if)# interface vlan 2
console (config-if)# name ipmi-net
console (config-if)# interface vlan 3
console (config-if)# name site-admin-net
console (config-if)# interface vlan 4
console (config-if)# name site-user-net
console (config-if)# exit
```

7. Configure management IP and the netmask. This is always on VLAN 1 (on `esmaint-net`).

```
console (config)# interface vlan 1
console (config-if)# ip address 10.141.0.100 255.255.0.0
console (config-if)# exit
```

8. Set the default gateway of VLAN 1 to be the IP address of `eth0` on the CIMS.

```
console (config)# ip default-gateway 10.141.255.254
```

9. Configure the ports to the VLANs using the scheme shown in Figure 2. In the commands below, replace *NN* with the appropriate VLAN port number.

   a. Configure the remaining VLAN 1 ports (on `esmaint-net`) by running the following two commands for each port on this VLAN.

```
console (config)# interface gigabitethernet 0/NN
console (config-if)# switchport access vlan 1
```

   b. Configure the VLAN 2 ports (on `ipmi-net`) by running the following two commands for each port on this VLAN.

```
console (config-if)# interface gigabitethernet 0/NN
console (config-if)# switchport access vlan 2
```

   c. Configure the VLAN 3 ports (on `site-admin-net`) by running the following two commands for each port on this VLAN.

```
console (config-if)# interface gigabitethernet 0/NN
console (config-if)# switchport access vlan 3
```

   d. Configure the VLAN 4 ports (on `site-user-net`) by running the following two commands for each port on this VLAN.

```
console (config-if)# interface gigabitethernet 0/NN
console (config-if)# switchport access vlan 4
```

10. Disable spanning tree protocol (STP) to disable loop-free, redundant bridging paths between daisy-chained switches.

```
console (config-if)# no spanning-tree
```

11. Set up the `admin` and `root` users:

```
console (config-if)# exit
console (config)# username admin privilege 15 password initial0
console (config)# username root privilege 15 password initial0
console (config)# exit
```

12. Save the configuration.

```
console# exit
console# write
Overwrite file [startup-config]? [yes/press any key for no] yes
console# exit
```

3.18 Zone the QLogic® FC Switch

If your system includes QLogic® Fibre Channel (FC) switch, follow Procedure 32 on page 131 to zone the LUNs on your QLogic SANBox™ switch by using a utility called *QuickTools*. If a LUN is to be shared between failover host pairs, each host must be given access to the LUN. The CIMS host port should be given access to all LUNs.

QuickTools is an application that is embedded in the QLogic switch and is accessible from a workstation browser with a compatible Java™ plug-in. You must have a Java browser plugin, version 1.4.2 or later.

These instructions assume that the disk device has four host ports connected to ports 0-3 for the QLogic SANbox switch.

Zoning is implemented by creating a *zone set*, adding one or more zones to the zone set, and selecting the ports to use in the zone.

This procedure presupposes that the SANBox is configured and on `esmaint-net` network.

Procedure 32. Configuring zoning for a QLogic SANbox switch using QuickTools utility

1. Start a web browser.
2. Enter the IP address of your switch on the `esmaint-net` network. The IP address of each RAID controller is preconfigured by Cray and is listed on a sticker on the back of the RAID controller.
3. Enter the login name and password when the **Add a New Fabric** window pops up and prompts for them. The default administrative login name is `admin`, and the default password is `password`.

4. The QuickTools utility displays in your browser. Click **Add Fabric**. If you receive a dialog box notification that the request failed to connect over a secured connection, click **Yes** and continue.
5. The switch is located and displayed in the window. Double-click the **switch** icon. Information about the switch displays in the right panel.
6. At the bottom of the panel, click the **Configured Zonesets** tab.
7. From the Tool Bar menu, select **Zoning** and then **Edit Zoning**. The **Edit Zoning** window displays.
8. Click the **Zone Set** button. The **Create a Zone Set** window displays. Create a new zone set. (In this example, assume that the zone set is named `XT0`.)
9. Right-click the **XT0 zone** and select **Create a Zone**.
10. Create a new zone name.
11. On the right panel, click the button in front of *zonename* to open a view of the domain members.
12. Define the ports in the zone to ensure that the discovery of LUNs is consistent among the CIMS, CLFS, and CDL nodes. Using the mouse, left-click on the desire port, and draft it to *zonename*.
13. Click **Apply**. The **error-checking** window displays.
14. When prompted, select **Perform Error Check**.
15. After confirming that no errors were found, click **Save Zoning**.
16. When prompted to activate a Zone Set, click **Yes** and then select the appropriate **XT3** zone set.
17. At this point, Cray recommends that you create a backup of your switch configuration (Procedure 33 on page 132) before you close and exit the application.

Procedure 33. Create a backup of your QLogic switch configuration

Create a backup of your QLogic switch configuration with the QuickTools utility. You must have a Java browser plugin, version 1.4.2 or later to use QuickTools.

If you need to start your web browser and open the QuickTools utility, complete steps 1 through 4. If you currently have the QuickTools utility open, skip to step 5.

1. Start a web browser.
2. Enter the IP address of your switch on `esmaint-net`. The IP address of each RAID controller is preconfigured by Cray and is listed on a sticker on the back of the RAID controller.

3. Enter the login name and password when the **Add a New Fabric** window pops up and prompts for them. The RAID default administrative login name is `admin`, and the default password is `password`.
4. The QuickTools utility appears. Click **Add Fabric**. If you receive a dialog box that states that the request failed to connect over a secured connection, click **Yes** and continue.
5. From within the QuickTools utility, complete the configuration backup.
   a. At the top bar, select **Switch** and then **Archive**. A **Save** window pops up with blanks for **Save in:** and **File Name:**.
   b. Enter the directory (for example, `crayadm`) and a file name (for example, `sanbox_archive`) for saving your QLogic switch configuration.
   c. Click the **Save** button.
6. Close and exit the application.

3.19 Configure the InfiniBand (`ib-net`) Network for Slave Nodes

The InfiniBand® network (`ib-net`) is used only by slave nodes in the DMP system, but is configured on the CIMS. Before adding any slave nodes to the system, you must configure the InfiniBand network (`ib-net`) in Bright.

Procedure 34. Configure the InfiniBand (`ib-net`) network in Bright

1. Open a window on the CIMS, log in as `root`, and enter the `cmsh` command:

```
esms1# cmsh
[esms1]%
```

2. Switch to `network` mode:

```
[esms1]% network
[esms1->network]%
```

3. Check the currently configured networks with the `list` command. These networks were defined in the customized Cray XML configuration file. The network `globalnet` is created by Bright but is not used in a DMP system.

```
[esms1->network]% list
Name (key)       Type         Netmask bits     Base address     Domain name          IPv6
---------------- ------------ ---------------- ---------------- -------------------- ----
esmaint-net      Internal     16               10.141.0.0       esmaint-net.cluster  no
globalnet        Global       16               0.0.0.0          cm.cluster           no
ipmi-net         Internal     16               10.148.0.0       ipmi-net.cluster     no
site-admin-net   External     20               aaa.bbb.ccc.ddd  your.domain.com       no
```

4. Use the `clone` subcommand to clone a similar network:

```
[esms1->network]% clone ipmi-net ib-net
```

5. Set the network parameters.

a. Set the base IP address to 10.149.0.0.

```
% set baseaddress 10.149.0.0
```

b. Set the domain name to `ib-net.cluster`.

```
% set domainname ib-net.cluster
```

c. Set the netmask bits to 16.

```
% set netmaskbits 16
```

d. Set the MTU:

```
% set mtu 2044
```

e. Set the broadcast address:

```
% set broadcastaddress 10.149.255.255
```

f. Use the `show` subcommand to view your changes.

```
% show
Parameter                      Value
------------------------------ ------------------------------------------------
Base address                   10.149.0.0
Broadcast address              10.149.255.255
Domain Name                    ib-net.cluster
Dynamic range end              0.0.0.0
Dynamic range start            0.0.0.0
Gateway                        0.0.0.0
IPv6                           no
Lock down dhcpd                no
MTU                            2044
Management allowed             no
Netmask bits                   16
Node booting                   no
Notes                          <0 bytes>
Revision
Type                           Internal
name                           ib-net
```

g. Save your changes.

```
% commit
```

6. Display the changed network list.

```
[esms1->network]% list
Name (key)       Type         Netmask bits     Base address     Domain name          IPv6
---------------- ------------ ---------------- ---------------- -------------------- ----
esmaint-net      Internal     16               10.141.0.0       esmaint-net.cluster  no
globalnet        Global       16               0.0.0.0          cm.cluster           no
ib-net           Internal     16               10.149.0.0       ib-net.cluster       no
ipmi-net         Internal     16               10.148.0.0       ipmi-net.cluster     no
site-admin-net   External     20               aaa.bbb.ccc.ddd  your.domain.com      no
```

7. Exit `cmsh`.

```
% quit
```

3.20 Use the iDRAC Remote Console

In addition to the remote console capabilities of Bright, DELL™ servers provide a remote console and administrative interface through the IP address of the iDRAC port on the CIMS that is accessible from a web browser. The iDRAC interface enables CIMS control through the baseboard management controller (BMC). Refer to Administrative Passwords on page 82 to change the iDRAC administrative password from the Bright `cmsh` shell.

The iDRAC port on each node in the system can also be accessed through a web browser (Firefox) by using secure shell (SSH) X11 forwarding from the CIMS.

Procedure 35. Use secure shell X11 forwarding and Firefox to open a remote console

1. Start X server software (such as Xming) on your Windows computer.
2. Connect to the CIMS node using secure shell X11 forwarding.

```
remote # ssh -X root@esms1
```

3. Start the Firefox web browser.

```
remote # firefox&
```

4. Enter the IP address of the node iDRAC port to connect and login to the iDRAC.

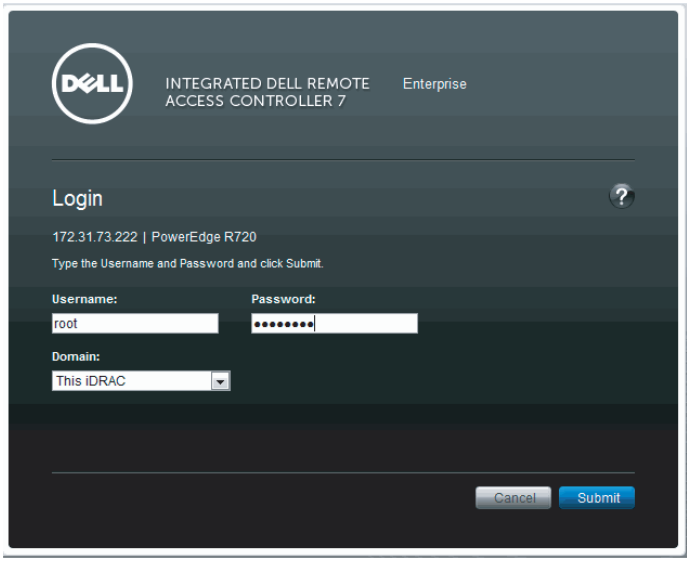
For detailed information, see the Dell iDRAC documentation:

`http://support.dell.com/support`

Procedure 36. Use the iDRAC web interface and remote console

1. From a web browser, open the IP address of the CIMS iDRAC port, `https://`*idrac_port_IP*. A login screen displays.

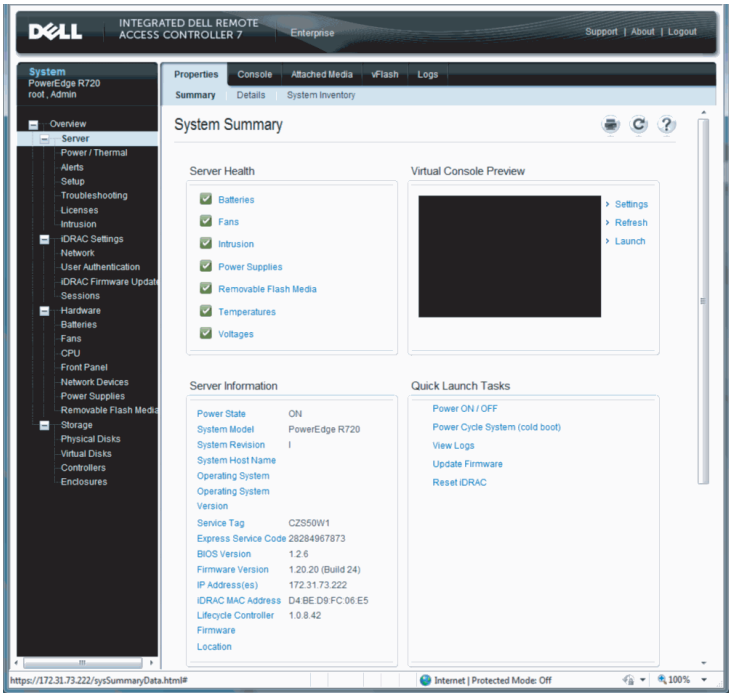
Figure 30. iDRAC Login Page



2. Log in as `root` and click **Submit**.

   The **System Summary** window displays. CIMS power, control, and status information is available from this interface.

Figure 31. iDRAC System Summary Page

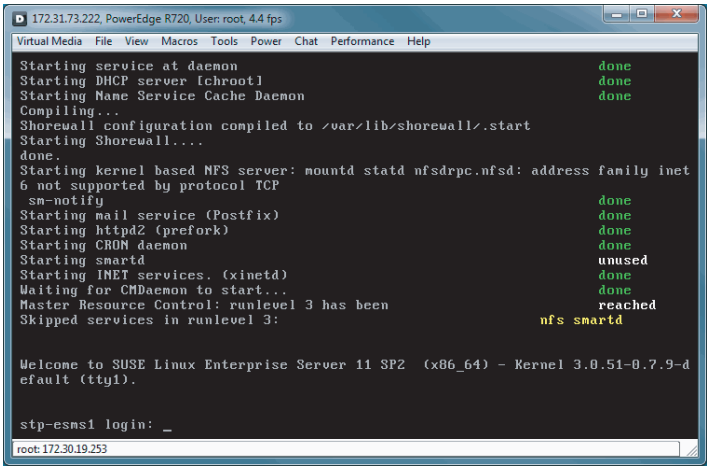


3. To access the CIMS console, click on the **Console** tab.

The **Virtual Console** window appears.

4. Click on **Launch Virtual Console**. The iDRAC web interface will install a Virtual Console Java™ application. Confirm the installation of this application to use the Virtual Console feature from an iDRAC port.

Figure 32. iDRAC Remote Console Window



Tip: Press the `F9` key to access the console window menu. To logout of the virtual console, close the window or select **File**, then **Exit** from the console menu. You will still be logged into the iDRAC port from your web browser.

3.21 Change iDRAC Settings After ESM Software Installation

The iDRAC port is configured during the ESM software installation procedure and normally does not need to be reconfigured after the ESM software is installed. This procedure makes configuration changes to the iDRAC (changes the gateway setting) using `cmsh partition` mode remote access controller administration (RACADM) commands. See the *Bright Cluster Manager 6.1 Administrator Manual* for more information about `cmsh partition` mode. See the *RACADM Command Line Reference Guide* from www.dell.com/support for more detailed iDRAC RACADM command information.

Procedure 37. Change iDRAC settings after ESM software installation

The procedure changes the gateway settings for the iDRAC, and can be used to maker other configuration changes to the iDRAC after the ESM software is installed.

1. Log into the CIMS as `root`.

```
remote% ssh root@esms1
esms1# cmsh
[esms1]%
```

2. Use the `netstat` command to display the default route for the CIMS iDRAC on the 10.148.0.0 network.

```
esms1# netstat -rn
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
0.0.0.0         0.0.0.0         0.0.0.0         UG        0 0          0 eth1
10.141.0.0      0.0.0.0         255.255.0.0     U         0 0          0 eth0
10.148.0.0      0.0.0.0         255.255.0.0     U         0 0          0 eth2
127.0.0.0       0.0.0.0         255.0.0.0       U         0 0          0 lo
0.0.0.0         0.0.0.0         255.255.0.0     U         0 0          0 eth0
0.0.0.0         0.0.0.0         255.255.255.0   U         0 0          0 eth1
esms1#
```

3. Start `cmsh` and enter `parition` mode, select the `base` partition, and display the BMC (iDRAC) administrator user name and password so that you can log in to the iDRAC.

```
esms1# cmsh
[esms1]% partition
[esms1->partition]% use base
[esms1->partition[base]]% get bmcusername
root
[esms1->partition[base]]% get bmcpassword
iDRACrootpassword
```

4. Exit `cmsh` and return to the CIMS node root prompt.

```
[esms1->partition[base]]% quit
esms1#
```

5. Log in to the iDRAC (typically named *CIMShostname*`-drac`) as `root`. Use the password obtained in step 3. The iDRAC hostname is configured during ESM software installation BIOS configuration.

```
esms1# ssh root@esms1-drac
The authenticity of host 'esms1-drac (aaa.bbb.ccc.ddd)' can't be established.
RSA key fingerprint is 2d:c7:de:12:28:d8:77:3e:ad:fa:12:47:35:3e:8b:12 [MD5].
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'esms1-drac,xxx.xxx.xxx.xxx' (RSA) to the list of known hosts.
root@esms1-drac's password: iDRACrootpassword
#
```

6. Display the gateway IP address for the iDRAC.

```
/admin1-> racadm getconfig -g cfgLanNetworking -o cfgNicGateway
0.0.0.0
/admin1->
```

7. Set the iDRAC gateway IP address.

```
/admin1-> racadm config -g cfgLanNetworking -o cfgNicGateway GatewayIPAddress
Object value modified successfully
/admin1-> racadm getconfig -g cfgLanNetworking -o cfgNicGateway
GatewayIPAddress
/admin1->
```

8. Open another shell window on the CIMS node, and use `ping` to verify the iDRAC gateway IP address is set correctly.

```
esms1# ping esms1-drac
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 4.632/5.448/6.265/0.819 ms
```

9. Start `cmsh` and enter the new configuration into the Bright database.

```
esms1# cmsh
[esms1]% device use esms1
[esms1->device[esms1]]% interfaces
[esms1->device[esms1]->interfaces]% show ipmi0
Parameter                      Value
------------------------------ ------------------------------------------------
Additional Hostnames
DHCP                           no
Gateway                        0.0.0.0
IP                             aaa.bbb.ccc.ddd
MAC                            00:00:00:00:00:00
Network                        site-admin-net
Network device name            ipmi0
Revision
Start if                       ALWAYS
Type                           bmc
VLAN ID                        0
```

10. Set the gateway IP address for the `ipmi0` interface and commit the new configuration into the Bright database.

```
[esms1->device[esms1]->interfaces]% use ipmi0
[esms1->device[esms1]->interfaces[ipmi0]]% set gateway GatewayIPAddress
[esms1->device*[esms1*]->interfaces*[ipmi0*]]% commit
[esms1->device[esms1]->interfaces[ipmi0]]% show
Parameter                      Value
------------------------------ ------------------------------------------------
Additional Hostnames
DHCP                           no
Gateway                        GatewayIPAddress
IP                             aaa.bbb.ccc.ddd
MAC                            00:00:00:00:00:00
Network                        site-admin-net
Network device name            ipmi0
Revision
Start if                       ALWAYS
Type                           bmc
VLAN ID                        0
```

11. Quit `cmsh`.

```
[esms1->device[esms1]->interfaces[ipmi0]]% quit
esms1#
```

3.22 Update iDRAC Firmware

Use this procedure to update Dell iDRAC firmware if directed by Cray technical support.

Procedure 38. Update iDRAC firmware

1. Download the iDRAC firmware using a CrayPort account.
2. Open up a web browser session and enter the IP address of the iDRAC6 port.
3. Log in to the iDRAC as `root` user.
4. The System Summary page will be displayed. On that page you will see your current firmware version under **Server Information**.
5. Under the **Quick Launch Tasks** table, select **Update Firmware**.

Figure 33.


Firefox
idrac-glacier - iDRAC6 - System Summary
https://172.30.72.49/index.html
Most Visited
Getting Started
CDT
Cray
From Internet Explore
DELL
INTEGRATED DELL REMOTE ACCESS CONTROLLER 6 - ENTERPRISE
System
PowerEdge R710
root , Admin
System
iDRAC Settings
Batteries
Fans
Intrusion
Power Supplies
Removable Flash Media
Temperatures
Voltages
Power Monitoring
LCD
Properties
Setup
Power
System Summary
System Details
System Summary
Server Health
Status
Component
Batteries
Fans
Intrusion
Power Supplies
Removable Flash Media
Temperatures
Voltages
Server Information
Power State
ON
System Model
PowerEc
System Revision

a. Select the firmware file (`firmimg.d6` for example, for an iDRAC6) and select **Upload**. The upload may take several minutes.

b. Select **Next**.

c. Select **OK**, to confirm the message: `Are you sure you want to proceed with the update?`

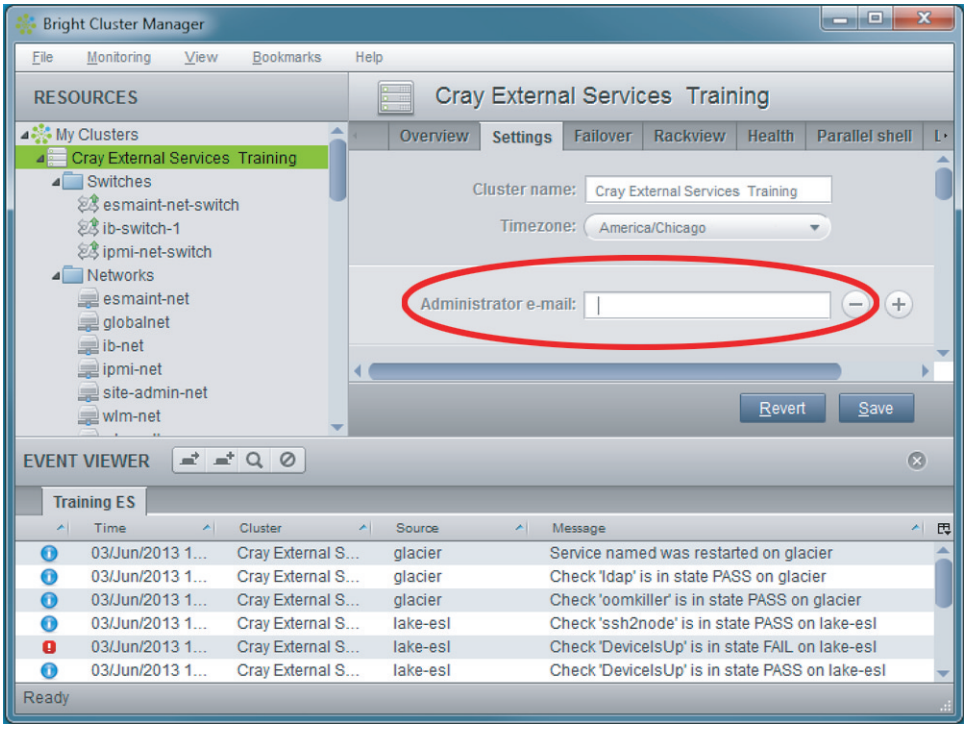
6. The iDRAC resets after confirming the update. To verify the firmware update, open the IP address of the iDRAC and display the firmware version as described in step 4. Also, verify you can launch the Virtual Console Window.

3.23 Configure Administrator E-mail Alerts from the CIMS

Procedure 39. Configure administrator e-mail alerts from the CIMS using `cmgui`

1. Open the `cmgui` and select the DMP system name in the **RESOURCES** tree.
2. Select the **Settings** tab.
3. Click on the + symbol and enter the administrator E-mail address in the **Administrator e-mail** field.

Figure 34. Administrator's E-mail Address



Procedure 40. Configure administrator e-mail alerts from the CIMS using cmsh

1. Log in to the CIMS node as `root` and start `cmsh`.

```
esms1# cmsh
[esms1]%
```

2. Switch to `partition` mode.

```
[esms1]%partition
[esms1->partition]%
```

3. Use the base object in `partition` mode.

```
[esms1->partition]% use base
[esms1->partition[base]]%
```

4. Set the administrator's e-mail address.

```
[esms1->partition[base]]% set administratore-mail johndoe@server.com
[esms1->partition*[base*]]% commit
Tue Jun  4 10:09:27 2013 [notice] esms1: Service postfix was restarted
```

5. (Optional) To include additional administrator e-mail addresses, enter:

```
[esms1->partition[base]]% append administratore-mail janesmith@server.com
[esms1->partition*[base*]]% get administratore-mail
johndoe@server.com
janesmith@server.com
[esms1->partition*[base*]]% commit
Tue Jun  4 10:11:12 2013 [notice] esms1: Service postfix was restarted
```

6. (Optional) To remove `johndoe@server.com` from the administrators e-mail list enter:

```
[esms1->partition[base]]% removefrom administratore-mail johndoe@server.com
[esms1->partition*[base*]]% commit
Tue Jun  4 10:11:12 2013 [notice] esms1: Service postfix was restarted
```

3.24 Configure SSH Keys for eswrap on CDL and Internal Login Nodes

This procedures describes how to configure SSH Keys on CDL and internal login nodes by using `ssh-agent` or passphrase-less RSA®/DSA keys. By default, SSH prompts for a password on each command wrapped by `eswrap`. Consult the site security policies before configuring transparent SSH access. This procedure creates DSA keys.

Procedure 41. Configuring SSH Keys for eswrap on CDL and internal login nodes

1. Log in to the CIMS as `root`.

2. SSH to a CDL node as `root` and create a key with `ssh-keygen`.

```
esms1# ssh eslogin1
Last login: Tue May  7 10:52:08 2013 from esms1.cm.cluster
eslogin1#
```

3. Select the type of key, either RSA or DSA and the number of bits (DSA keys must be 1024 bits). The choice depends on the site security policy. Save the key in the `id_dsa.pub` file.

```
eslogin1# ssh-keygen -t dsa -b 1024
Generating public/private dsa key pair.
Enter file in which to save the key (/root/.ssh/id_dsa): /root/.ssh/id_dsa.pub
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_dsa.pub
Your public key has been saved in /root/.ssh/id_dsa.pub
The key fingerprint is:
56:cd:94:e4:3f:ef:a2:9a:a2:bc:17:c4:63:8b:a6:1a root@eslogin1
The key's randomart image is:
+--[ DSA 1024]----+
|           .o.   |
|           =.    |
|       .  . +    |
|        =.   .   |
|       +So    o  |
|      o.o      o |
|  E  o   .      .|
|   ...  o  .  .. |
|  ..  ++ .o... ..|
+-----------------+
```

4. Copy the contents of the public key file (`id_dsa.pub` or `id_dsa.pub`).

5. Edit the `$HOME/.ssh/authorized_keys` file on the Cray internal login node and append the public key from the CDL `id_dsa.pub` file.

6. Repeat step 2 through step 4 for all CDL nodes that can access the Cray system.

   On a system that does not share user home directories between CDL nodes and the Cray system, you can remove the SSH key prompts for an unknown host by setting up a `known_hosts` file (`$HOME/.ssh/known_hosts`) and/or authorized_keys file (`$HOME/.ssh/authorized_keys`) on the Cray system.

7. SSH to an internal Cray login node (`craylogin`).

```
eslogin1# ssh craylogin
The authenticity of host 'craylogin (10.128.1.132)' can't be established.
DSA key fingerprint is a8:0d:b0:5c:f8:d2:ec:4f:00:b8:69:87:7d:28:ac:05.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'craylogin,10.128.1.132' (DSA) to the list of known hosts.
Password:
Creating directory '/home/users/username'.

Welcome to XE system  craylogin
```

8. Create a `.ssh` directory for login `username` and `cd` into that directory.

```
craylogin users/username> mkdir -p ~/.ssh
craylogin users/username> cd .ssh
```

9. Use secure FTP to transfer the `id_dsa.pub` file you create in step 3 from the `eslogin1` node to the Cray internal login node.

```
craylogin users/username/.ssh> sftp username@eslogin1:.ssh/id_dsa.pub
. Connecting to eslogin1...
The authenticity of host 'eslogin1 (aaa.bbb.ccc.ddd)' can't be established.
DSA key fingerprint is b8:87:2c:43:31:2d:9f:64:2b:30:e3:08:45:cb:78:65.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'eslogin1,aa.bbb.ccc.ddd' (DSA) to the list of known hosts.
Password:
Fetching /home/users/username/.ssh/id_dsa.pub to ./id_dsa.pub
/home/users/username/.ssh/id_dsa.pub    100%  737     0.7KB/s   00:00

craylogin users/username/.ssh> ls
id_rsa.pub  known_hosts
```

10. Copy the `id_dsa.pub` key to an `authorized_keys` file.

```
craylogin users/username/.ssh> cat id_dsa.pub >> authorized_keys
ccraylogin users/username/.ssh> ls
authorized_keys  id_dsa  id_dsa.pub  known_hosts
```

11. Logout of the Cray internal login node.

```
craylogin users/username/.ssh> logout
Connection to craylogin closed.
```

12. Verify you can run wrapped commands without entering a password.

```
eslogin1# eswrap --help
eslogin1# echo $ESWRAP_LOGIN
eslogin1# which xtnodestat
/opt/cray/eslogin/eswrap/1.0.15/bin/xtnodestat

eslogin1# xtnodestat
Current Allocation Status at Mon Sep 10 13:13:00 2012

     C0-0             C1-0             C2-0             C3-0
  n3 ---------------- AAAAAAAAAAAAAAaa ---------------- --------------AA
  n2 ---------------- AAAAAAAAAAAAAAaa ---------------- --------------AA
  n1 ---------------- AAAAAAAAAAAAAAaa ---------------- --------------AA
c2n0 ---------------- AAAAAAAAAAAAAAaa ---------------- --------------AA
  n3 - -------------- AAAAAAAAAAAAAAAA a aaaaaaaaaaaaaa ----------------
  n2 -S-------------- AAAAAAAAAAAAAAAA aSaaaaaaaaaaaaaa ----------------
  n1 -S-------------- AAAAAAAAAAAAAAAA aSaaaaaaaaaaaaaa ----------------
c1n0 - -------------- AAAAAAAAAAAAAAAA a aaaaaaaaaaaaaa ----------------
  n3  --------------- --------------AA  aaaaaaaaaaaaaaa ----------------
  n2 S--------------- --------------AA Saaaaaaaaaaaaaaa ----------------
  n1 S--------------- --------------AA Saaaaaaaaaaaaaaa ----------------
c0n0  --------------- --------------AA  aaaaaaaaaaaaaaa ----------------
    s0123456789abcdef 0123456789abcdef 0123456789abcdef 0123456789abcdef


Legend:
   nonexistent node                      S  service node
;  free interactive compute node         -  free batch compute node
A  allocated (idle) compute or ccm node  ?  suspect compute node
W  waiting or non-running job            X  down compute node
Y  down or admindown service node        Z  admindown compute node

Available compute nodes:          0 interactive,        488 batch


Job ID     User       Size   Age        State           command line
--- ------ --------   -----  ---------  --------  ---------------------------------
a   302511 addy       128    0h17m      run        gamess.ga.x
eslogin1#
```

3.25 Back Up Slave Node Software Images

Back up slave node software images and other configuration files regularly according you site policy. Slave node software images must exist in the default `/cm/images` directory on the CIMS node when restoring a backup of the Bright database from a system configuration XML file. Other site customizations (such as `/etc/fstab`, `/etc/hosts`, LDAP configuration), should also be backed up according to site policy.

Procedure 42. Back up slave node software images

1. Log in to the CIMS node as `root`.

2. Change directories to `/cm/images` and use the following command to backup the entire `/cm` partition to a tarball in `/cm-backup.tar.gz`. Optionally, you can backup only the software images required for all of the slave nodes.

```
esms1# cd /cm/images
esms1# tar zcf cm-backup.tar.gz /cm
tar: Removing leading `/' from member names
tar: Removing leading `/' from hard link targets

esms1# ls -l cm-backup.tar.gz
-rw-r--r-- 1 root root 16384685013 Sep 19 14:25 cm-backup.tar.gz
```

3. Back up other CIMS node site customizations such as `/etc/fstab`, `/etc/hosts` and LDAP configuration files according to site policy.

3.26 Back Up System Configuration Settings to an XML File

The Bright system configuration can be saved to an XML file so that it can be used to recover the system configuration from a backup. To save your configuration, use the `cmd -x` *siteconfigfile*`.xml` command, to generate a human-readable configuration file.

Caution: The `cmd -i` *siteconfigfile*`.xml` command can restore your system configuration only when all slave node software images are in `/cm/images` are present and all other site customizations have been configured and saved in Bright.

Procedure 43. Save the system configuration settings to an XML file

1. Log in to the CIMS as `root`.

```
remote% ssh root@esms1
```

2. Stop the `cmd` daemon.

```
esms1#/etc/init.d/cmd stop
Waiting for CMDaemon (26823) to terminate...
                                          done
```

3. Dump the system configuration to an XML file.

```
# cmd -x SiteConfig_Dump.xml
Wed Feb 13 13:37:50 2013   Info: CMDaemon version 1.4 (r15096)
Wed Feb 13 13:37:50 2013   Info: Reading configuration from /cm/local/apps/cmd/etc/cmd.conf
Wed Feb 13 13:37:50 2013   Info: CMDaemon auditing is disabled
Wed Feb 13 13:37:50 2013   Info: Initialize cmdaemon database
Wed Feb 13 13:37:50 2013   Info: Initialize monitoring database
Wed Feb 13 13:37:50 2013   Info: Database: Mirroring not required to remote master. no partition
Wed Feb 13 13:37:50 2013   Info: Database: no index on timestamp present in MonData table
Wed Feb 13 13:37:50 2013   Info: Database: using mysql's bulkinsert with interval of 3600s
Wed Feb 13 13:37:51 2013   Info: Succesfully stored configuration in SiteConfig_Dump.xml
```

4. Start the `cmd` daemon.

```
# /etc/init.d/cmd start
Waiting for CMDaemon to start...                done
```

Procedure 44. Load system configuration settings from an XML file

Caution: Use the `cmd -i` *siteconfigfile*`.xml` command with caution. You can restore your system configuration using the `cmd -i` command only if you have all the system images available in `/cm/images` and all other site customizations properly configured in Bright. If the `cmd -i` command fails, it could potentially render the system inoperable.

1. Log in to the CIMS as `root`.

```
remote% ssh root@esms1
```

2. Stop the `cmd` daemon.

```
esms1# /etc/init.d/cmd stop
Waiting for CMDaemon (26823) to terminate...
                                          done
```

3. Dump the system configuration to an XML file.

```
esms1# cmd -i SiteConfig.xml
Wed Feb 13 13:44:48 2013   Info: CMDaemon version 1.4 (r15096)
Wed Feb 13 13:44:48 2013   Info: Reading configuration from /cm/local/apps/cmd/etc/cmd.conf
Wed Feb 13 13:44:48 2013   Info: CMDaemon auditing is disabled
Wed Feb 13 13:44:49 2013   Info: Billing Service enabled
Wed Feb 13 13:44:49 2013   Info: Initialize cmdaemon database
Wed Feb 13 13:44:49 2013   Info: Drop cmdaemon database
Wed Feb 13 13:44:49 2013   Info: Create cmdaemon database
Wed Feb 13 13:44:52 2013   Info: Recreate monitoring database
.
.
.
```

4. Start the `cmd` daemon. All slave nodes appear `DOWN` in Bright until you restart the `cmd` service on each slave node.

```
esms1# /etc/init.d/cmd start
Waiting for CMDaemon to start...               done
```

3.27 Resize Partitions on the CIMS

If the CIMS was not configured as an HA system and is being integrated into an HA system, then you must resize the `/cm` partition to accommodate the required Distributed Replicated Block Device (DRBD) partitions. If DRBD partitions are already defined, do not perform this procedure.

Procedure 45. Resize partitions on a CIMS

1. Log in to the CIMS as `root`.

```
remote% ssh root@esms1
```

2. Determine which devices holds the `/cm` partition.

```
esms1# df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/sda5       177G  3.1G  165G   2% /
devtmpfs         32G  240K   32G   1% /dev
tmpfs            32G     0   32G   0% /dev/shm
/dev/sda1       473M   22M  427M   5% /boot
/dev/sdb3       3.3T   30G  3.1T   1% /cm
/dev/sda3        61G  180M   57G   1% /tmp
/dev/sdb1       962G  373M  913G   1% /var
/dev/sdb2       9.4G  155M  8.8G   2% /var/lib/mysql/cmdaemon_mon
```

3. Backup the existing `/cm` partition to a tarball in `/cm-backup.tar.gz`.

```
esms1# tar zcf cm-backup.tar.gz /cm
tar: Removing leading `/' from member names
tar: Removing leading `/' from hard link targets

esms1# ls -l cm-backup.tar.gz
-rw-r--r-- 1 root root 16384685013 Sep 19 14:25 cm-backup.tar.gz
```

4. Use the `parted` utility to verify where `/cm` is mounted on `/dev/sdb` (`/dev/sdb` was determined to hold the `/cm` partition from step 2.

```
esms1# parted /dev/sdb
GNU Parted 2.3
Using /dev/sdb
Welcome to GNU Parted! Type 'help' to view a list of commands.
(parted) p
Model: DELL PERC H710P (scsi)
Disk /dev/sdb: 4723GB
Sector size (logical/physical): 512B/512B
Partition Table: gpt

Number  Start   End     Size    File system  Name                          Flags
 1      17.4kB  1049GB  1049GB  ext3         /var
 2      1049GB  1059GB  10.2GB  ext3         /var/lib/mysql/cmdaemon_mon
 3      1059GB  4723GB  3665GB  ext3         /cm

(parted) quit
esms1#
```

5. Shut down all slave nodes to free `/cm/shared` which is NFS® mounted on all slave nodes. This example has only one slave node named `eslogin1`.

```
esms1# cmsh
[esms1]% device
[esms1->device]% status
eslogin1 ............. [   UP   ]
esms1 ................. [   UP   ]
sw-1ge ................ [  DOWN  ]
[esms1->device]% shutdown eslogin1
eslogin1: Shutdown in progress ...
[esms1->device]%
Wed Sep 19 14:28:20 2012 [notice] esms1: eslogin1 [  DOWN  ]
[esms1->device]% power status
ipmi0 ................. [   OFF   ] eslogin1
ipmi0 ................. [   ON    ] esms1
No power control ......... [ UNKNOWN ] sw-1ge
[esms1->device]% quit
```

6. Shut down the CMDaemon (`cmd`).

```
esms1# /etc/init.d/cmd stop
Waiting for CMDaemon (4838) to terminate... done
```

7. Use `lsof` to determine if `/cm` is being used.

```
esms1# lsof /cm
COMMAND  PID USER   FD   TYPE DEVICE SIZE/OFF      NODE NAME
slapd   3475 ldap  txt    REG   8,19  2949776 159072337 /cm/local/apps/openldap/sbin/slapd
slapd   3475 ldap  mem    REG   8,19   441348 159072328 /cm/local/apps/openldap/lib/libldap_r-2.4.so.2.8.3
slapd   3475 ldap  mem    REG   8,19    81962 159072318 /cm/local/apps/openldap/lib/liblber-2.4.so.2.8.3
slapd   3475 ldap  mem    REG   8,19 10036285 159065221 /cm/local/apps/openldap/db4/lib/libdb-4.6.so
conmand 5035 root  txt    REG   8,19   387897 159121453 /cm/local/apps/conman/sbin/conmand
conmand 5035 root  mem    REG   8,19   781050 159072826 /cm/local/apps/freeipmi/1.1.3/lib/libipmiconsole.so.2.2.1
conmand 5035 root  mem    REG   8,19  6524146 159072821 /cm/local/apps/freeipmi/1.1.3/lib/libfreeipmi.so.12.0.2
conmand 5035 root    5rR  REG   8,19      414 159121420 /cm/local/apps/conman/etc/conman.conf
```

8. step 7 shows that `ldap` and `conman` are using the `/cm` partition. Stop the `ldap` and `conman` services. Because `/cm` contains `/cm/shared`, which is NFS exported, stop the `nfsserver` service, as well.

```
esms1# /etc/init.d/ldap status
Checking for service ldap:                                 running
esms1# /etc/init.d/ldap stop
Shutting down ldap-server                                  done
esms1# /etc/init.d/conman stop
Stopping ConMan: conmand                                   done
esms1# /etc/init.d/nfsserver stop
Shutting down kernel based NFS server: nfsd statd mountd   done
```

9. Unmount `/cm` and run `parted` to resize the `/cm` partition by removing it and creating it with a smaller size.

   a. Unmount `/cm`.

```
esms1# umount /cm
```

b. Start `parted` and list the existing partitions

```
esms1# parted /dev/sdb
GNU Parted 2.3
Using /dev/sdb
Welcome to GNU Parted! Type 'help' to view a list of commands.
(parted) p
Model: DELL PERC H710P (scsi)
Disk /dev/sdb: 4723GB
Sector size (logical/physical): 512B/512B
Partition Table: gpt

Number  Start   End     Size    File system  Name                          Flags
 1      17.4kB  1049GB  1049GB  ext3         /var
 2      1049GB  1059GB  10.2GB  ext3         /var/lib/mysql/cmdaemon_mon
 3      1059GB  4723GB  3665GB  ext3         /cm

(parted)
```

c. Remove the `/cm` partition.

```
(parted) rm 3
(parted) p
Model: DELL PERC H710P (scsi)
Disk /dev/sdb: 4723GB
Sector size (logical/physical): 512B/512B
Partition Table: gpt

Number  Start   End     Size    File system  Name                          Flags
 1      17.4kB  1049GB  1049GB  ext3         /var
 2      1049GB  1059GB  10.2GB  ext3         /var/lib/mysql/cmdaemon_mon

(parted)
```

d. Make a new smaller sized `/cm` partition and quit.

```
(parted) mkpart /cm 1059GB 3930GB
(parted) p
Model: DELL PERC H710P (scsi)
Disk /dev/sdb: 4723GB
Sector size (logical/physical): 512B/512B
Partition Table: gpt

Number  Start   End     Size    File system  Name                          Flags
 1      17.4kB  1049GB  1049GB  ext3         /var
 2      1049GB  1059GB  10.2GB  ext3         /var/lib/mysql/cmdaemon_mon
 3      1059GB  3930GB  2871GB  ext3         /cm

(parted) quit
Information: You may need to run /etc/fstab.
```

10. Run `makefs.ext3` on the device that contained `/cm` (`/dev/sdb3` in this example).

```
esms1# mkfs.ext3 /dev/sdb3
mke2fs 1.41.9 (22-Aug-2009)
Filesystem label=
OS type: Linux
Block size=4096 (log=2)
Fragment size=4096 (log=2)
175243264 inodes, 700972288 blocks
35048614 blocks (5.00%) reserved for the super user
First data block=0
Maximum filesystem blocks=4294967296
21392 block groups
32768 blocks per group, 32768 fragments per group
8192 inodes per group
Superblock backups stored on blocks:
        32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208,
        4096000, 7962624, 11239424, 20480000, 23887872, 71663616, 78675968,
        102400000, 214990848, 512000000, 550731776, 644972544

Writing inode tables:    10/21392
done
Creating journal (32768 blocks): done
Writing superblocks and filesystem accounting information: done
```

11. Mount the `/cm` partition.

```
esms1# mount /cm
esms1# df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/sda5       177G   19G  150G  11% /
devtmpfs         32G  244K   32G   1% /dev
tmpfs            32G     0   32G   0% /dev/shm
/dev/sda1       473M   22M  427M   5% /boot
/dev/sda3        61G  180M   57G   1% /tmp
/dev/sdb1       962G  373M  913G   1% /var
/dev/sdb2       9.4G  155M  8.8G   2% /var/lib/mysql/cmdaemon_mon
/dev/sdb3       2.6T  202M  2.5T   1% /cm
```

12. Restore `/cm` from the backup tarball.

```
esms1# cd /cm
esms1:/cm # cp /root/cm-backup.tar.gz .
esms1:/cm # tar zxf cm-backup.tar.gz
esms1:/ # ls /cm
CLUSTERMANAGERID  conf    local       node-installer  shared
README            images  lost+found  nodeinstaller
```

13. Start `nfsserver` and `cmd`.

```
esms1:/ # /etc/init.d/nfsserver start
Starting kernel based NFS server: mountd statd nfsdrpc.nfsd: address family inet6 not
supported by protocol TCP
 sm-notify                                                          done
esms1:/ # /etc/init.d/cmd start
Waiting for CMDaemon to start...                                    done
```

14. Boot the slave nodes.

```
esms1:/ # cmsh
[esms1]% device
[esms1->device]% list
Type                  Hostname (key)   MAC                Category         Ip
      Network        PowerDistributio
--------------------- ---------------- ------------------ ---------------- -------------- -------------
EthernetSwitch        sw-1ge           00:00:00:00:00:00                   10.141.253.1    esmaint-net
MasterNode            esms1            D4:AE:52:B5:E2:64                   10.141.255.254  esmaint-net
PhysicalNode          eslogin1         D4:AE:52:B5:A4:68  eslogin          10.141.0.1      esmaint-net
[esms1->device]% power status
ipmi0 ..................... [   OFF   ] eslogin1
ipmi0 ..................... [   ON    ] esms1
No power control ......... [ UNKNOWN ] sw-1ge

[esms1->device]% power -n eslogin1 on
ipmi0 ..................... [   ON    ] eslogin1
[esms1->device]%
Wed Sep 19 15:16:08 2012 [notice] esms1: eslogin1 [     INSTALLING      ] (nod
e installer started)
[esms1->device]%
Wed Sep 19 15:16:38 2012 [notice] esms1: eslogin1 [ INSTALLER_CALLINGINIT ] (s
witching to local root)
[esms1->device]%
Wed Sep 19 15:17:35 2012 [notice] esms1: eslogin1 [   UP   ]
[esms1->device]% status
eslogin1 ..................[   UP   ]
esms1 ..................... [   UP   ]
sw-1ge ..................... [  DOWN  ] health check failed
Wed Sep 19 15:18:01 2012 [notice] eslogin1: Check 'DeviceIsUp' is in state PASS on
 eslogin1

[esms1->device]% pexec -n eslogin1 "df -h"

[eslogin1] :
Filesystem         Size  Used Avail Use% Mounted on
/dev/sda5          767G  3.0G  725G   1% /
devtmpfs            32G  192K   32G   1% /dev
tmpfs               32G     0   32G   0% /dev/shm
/dev/sda3           31G  659M   28G   3% /tmp
/dev/sda2           61G  601M   57G   2% /var
master:/cm/shared  2.6T   31G  2.5T   2% /cm/shared
master:/home       177G   19G  150G  11% /home

[esms1->device]% quit
```

15. Create new `drbd` partitions in the free space of `/dev/sdb`

    a. Start `parted` and list partitions on `/dev/sdb`.

```
esms1:/ # parted /dev/sdb
GNU Parted 2.3
Using /dev/sdb
Welcome to GNU Parted! Type 'help' to view a list of commands.
(parted) p
Model: DELL PERC H710P (scsi)
Disk /dev/sdb: 4723GB
Sector size (logical/physical): 512B/512B
Partition Table: gpt

Number  Start   End     Size    File system  Name                            Flags
 1      17.4kB  1049GB  1049GB  ext3         /var
 2      1049GB  1059GB  10.2GB  ext3         /var/lib/mysql/cmdaemon_mon
 3      1059GB  3930GB  2871GB  ext3         /cm

(parted)
```

b. Make a `20GB`, `/drbd1` partition with an `ext2` file system.

```
(parted) mkpart
Partition name?  []? /drbd1
File system type?  [ext2]?
Start? 3930GB
End? 3950GB
(parted) p
Model: DELL PERC H710P (scsi)
Disk /dev/sdb: 4723GB
Sector size (logical/physical): 512B/512B
Partition Table: gpt

Number  Start   End     Size    File system  Name                          Flags
 1      17.4kB  1049GB  1049GB  ext3         /var
 2      1049GB  1059GB  10.2GB  ext3         /var/lib/mysql/cmdaemon_mon
 3      1059GB  3930GB  2871GB  ext3         /cm
 4      3930GB  3950GB  20.0GB               /drbd1
```

c. Make a `1GB`, `/drbd2` partition.

```
(parted) mkpart
Partition name?  []? /drbd2
File system type?  [ext2]?
Start? 3950GB
End? 3951GB
```

d. Make a `1GB`, `/drbd3` partition with an `ext2` file system.

```
(parted) mkpart
Partition name?  []? /drbd3
File system type?  [ext2]?
Start? 3951GB
End? 3952GB
```

e. Make a `16GB`, `/drbd4` partition.

```
(parted) mkpart
Partition name?  []? /drbd4
File system type?  [ext2]?
Start? 3952GB
End? 3968GB
```

f. Make a `/drbd5` partition with the remaining space (in this example `755GB`).

```
(parted) mkpart
Partition name?  []? /drbd4
File system type?  [ext2]?
Start? 3968GB
End? 4723GB

(parted) quit
Information: You may need to update /etc/fstab.
```

16. Show free disk blocks and files.

```
esms1:/ # df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/sda5       177G   19G  150G  11% /
devtmpfs         32G  264K   32G   1% /dev
tmpfs            32G     0   32G   0% /dev/shm
/dev/sda1       473M   22M  427M   5% /boot
/dev/sda3        61G  180M   57G   1% /tmp
/dev/sdb1       962G  349M  913G   1% /var
/dev/sdb2       9.4G  156M  8.8G   2% /var/lib/mysql/cmdaemon_mon
/dev/sdb3       2.6T   31G  2.5T   2% /cm
```

17. Make an `ext3` file system on `/dev/sdb4`.

```
esms1:/ # mkfs.ext3 /dev/sdb4
```

18. Make as `ext3` file system on `/dev/sdb5`.

```
esms1:/ # mkfs.ext3 /dev/sdb5
```

19. Make as `ext3` file system on `/dev/sdb6`.

```
esms1:/ # mkfs.ext3 /dev/sdb6
```

20. Make as `ext3` file system on `/dev/sdb7`.

```
esms1:/ # mkfs.ext3 /dev/sdb7
```

21. Make as `ext3` file system on `/dev/sdb8`.

```
esms1:/ # mkfs.ext3 /dev/sdb8
```

22. Verify the partitions are correct.

```
esms1:/ # parted /dev/sdb
GNU Parted 2.3
Using /dev/sdb
Welcome to GNU Parted! Type 'help' to view a list of commands.
(parted) p
Model: DELL PERC H710P (scsi)
Disk /dev/sdb: 4723GB
Sector size (logical/physical): 512B/512B
Partition Table: gpt

Number  Start   End     Size    File system  Name                          Flags
 1      17.4kB  1049GB  1049GB  ext3         /var
 2      1049GB  1059GB  10.2GB  ext3         /var/lib/mysql/cmdaemon_mon
 3      1059GB  3930GB  2871GB  ext3         /cm
 4      3930GB  3950GB  20.0GB               /drbd1
 5      3950GB  3951GB  999MB                /drbd2
 6      3951GB  3952GB  1000MB               /drbd3
 7      3952GB  3968GB  16.0GB               /drbd4
 8      3968GB  4723GB  755GB                /drbd5

(parted) quit
```

23. Create mount points and mount the new `drbd` partitions.

```
esms1# mkdir -p /drbd1 /drbd2 /drbd3 /drbd4 /drbd5
esms1# mount /dev/sdb4 /drbd1
esms1# mount /dev/sdb5 /drbd2
esms1# mount /dev/sdb6 /drbd3
esms1# mount /dev/sdb7 /drbd4
esms1# mount /dev/sdb8 /drbd5
esms1:/ # df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/sda5       177G   19G  150G  11% /
devtmpfs         32G  264K   32G   1% /dev
tmpfs            32G     0   32G   0% /dev/shm
/dev/sda1       473M   22M  427M   5% /boot
/dev/sda3        61G  180M   57G   1% /tmp
/dev/sdb1       962G  349M  913G   1% /var
/dev/sdb2       9.4G  156M  8.8G   2% /var/lib/mysql/cmdaemon_mon
/dev/sdb3       2.6T   31G  2.5T   2% /cm
/dev/sdb4        19G  173M   18G   1% /drbd1
/dev/sdb5       938M   18M  874M   2% /drbd2
/dev/sdb6       939M   18M  875M   2% /drbd3
/dev/sdb7        15G  166M   14G   2% /drbd4
/dev/sdb8       693G  198M  658G   1% /drbd5
esms1:/ # exit
```

3.28 Configure DHCP to Allow Requests from Unknown Nodes

By default, the CIMS node is configured to ignore PXE boot requests from nodes with unknown MAC addresses. You can configure the Dynamic Host Configuration Protocol (DHCP) server to allow the CIMS node to answer PXE boot requests from nodes with unknown MAC addresses. Consult your site's security policy before you enable this feature.

Procedure 46. Configuring DHCP to allow requests from unknown nodes

1. Edit `/cm/local/apps/cmd/etc/cmd.conf`.
2. Set `LockDownDhcpd = false`.
3. Save your changes and exit.

3.29 Changing the CIMS Firewall Configuration

The CIMS runs the Shorewall Firewall package. Configuration files are located in `/etc/shorewall` (such as the firewall rules in `/etc/shorewall/rules`).

By default, the CIMS allows ICMP traffic from the external interface (`site-admin-net`) and SSH traffic over ports 22 and 8081. Port 8081 (SSL) must be open to use the Bright Cluster Management GUI (`cmgui`) from an external server.

Depending on site requirements, the default firewall settings may need to be modified. If changes are made to the Shorewall configuration, you must restart the `shorewall` service using the following command.

```
esms1# /etc/init.d/shorewall restart
```

CDL Administration Tasks [4]

4.1 Create a CDL Node in Bright

The easiest way to add a new slave node is to clone an existing node that is configured and fully functional in Bright Cluster Manager® (Bright). When you do not have a functioning CDL node, you must clone the default node (`node001`) created during the CIMS installation process.

Procedure 47. Create a CDL node in Bright

The following procedures use the Bright management shell (`cmsh`). They may also be performed using the Bright GUI (`cmgui`). The `cmsh` command prompt displays an asterisk (`*`) when you have uncommitted changes. Be sure to commit your changes using the `commit` command before exiting `cmsh`, or your changes will be lost. When using the `cmgui`, be sure to click the **Save** button as needed to save and commit your changes.

1. Log in to the CIMS as `root` and start `cmsh`.

```
esms1# cmsh
[esms1]%
```

2. Enter `device` mode.

```
[esms1]% device
[esms1->device]%
```

3. List the available devices.

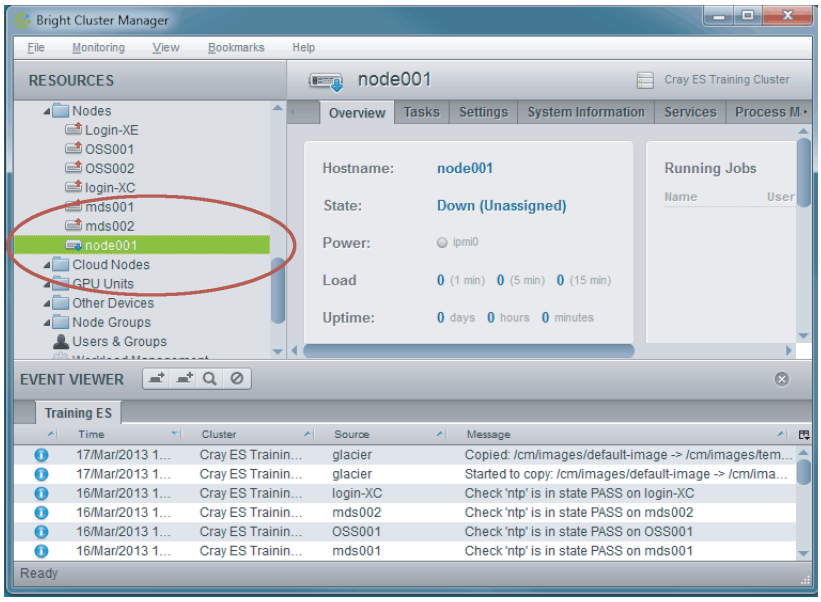
```
[esms1->device]% list
Type                  Hostname (key)   MAC                Category   Ip              Network
--------------------- ---------------- ------------------ ---------- --------------- --------------
EthernetSwitch        switch01         00:00:00:00:00:00             10.141.253.1    esmaint-net
MasterNode            esms1            78:2B:CB:40:CE:CA             10.141.255.254  esmaint-net
PhysicalNode          node001          00:00:00:00:00:00  default    10.141.0.1      esmaint-net
```

4. Clone an existing node such as the default node. This example creates a new CDL node named `eslogin1`.

```
[esms1->device]% clone node001 eslogin1
Base name mismatch, IP settings will not be modified!
```

When the CIMS is installed, Bright creates a default node, `node001`, which is placed in the `default` category and uses the default slave image (`/cm/images/default-image`). This image is assigned to the newly cloned node.

Figure 35. Default Node in Bright



When repeating this procedure for additional CDL nodes, clone the first fully configured and functional CDL node (`eslogin1`) instead of the default node (`node001`).

5. Change the interface settings for the new (cloned) node.

   a. Switch to `interfaces` mode and list interfaces on `eslogin1`.

```
[esms1->device*[eslogin1*]]% interfaces
[esms1->device*[eslogin1*]->interfaces]% list
Type         Network device name  IP              Network
------------ -------------------- --------------- ----------------
bmc          ipmi0                10.148.0.1      ipmi-net
physical     BOOTIF [prov]        10.141.0.1      esmaint-net
```

   b. Set the `BOOTIF` and `ipmi0` interface addresses for the new node. These addresses must be different from those used by the default or original node.

```
[esms1->device*[eslogin1*]->interfaces]% set bootif ip 10.141.0.2
[esms1->device*[eslogin1*]->interfaces*]% set ipmi0 ip 10.148.0.2
[esms1->device*[eslogin1*]->interfaces*]% list
Type         Network device name  IP              Network
------------ -------------------- --------------- ----------------
bmc          ipmi0                10.148.0.2      ipmi-net
physical     BOOTIF [prov]        10.141.0.2      esmaint-net
```

c. Set up the `ib-net` network and interface.

```
[esms1->device*[eslogin1*]->interfaces*]% add physical ib0
[esms1->device*[eslogin1*]->interfaces*[ib0*]]% set network ib-net
[esms1->device*[eslogin1*]->interfaces*[ib0*]]% set ip 10.149.0.2
[esms1->device*[eslogin1*]->interfaces[ib0*]]% show
Parameter                      Value
------------------------------ ------------------------------------------------
Additional Hostnames
Card Type
DHCP                           no
IP                             10.149.0.2
MAC                            00:00:00:00:00:00
Network                        ib-net
Network device name            ib0
Revision
Speed
Type                           physical
```

d. Commit your changes.

```
[esms1->device*[eslogin1*]->interfaces*[ib0*]]% commit
```

e. Exit `ib0`.

```
[esms1->device[eslogin1]->interfaces[ib0]]% exit
```

f. Check the results.

```
[esms1->device[eslogin1]->interfaces]% list
Type         Network device name  IP               Network
------------ -------------------- ---------------- ----------------
bmc          ipmi0                10.148.0.2       ipmi-net
physical     BOOTIF [prov]        10.141.0.2       esmaint-net
physical     ib0                  10.149.0.2       ib-net
```

g. Exit `interface` mode and return to `device` mode.

```
[esms1->device[eslogin1]->interfaces]% exit
```

h. Display the status of the new node.

```
[esms1->device[eslogin1]]% status
eslogin1 ............... [  DOWN  ]  (Unassigned)
```

The node is unassigned because the MAC address has not been set in Bright.

6. Set the MAC address for the CDL node (`eth0` on `esmaint-net`).

```
[esms1->device[eslogin1]]% set mac MACaddress
```

7. Set the management network to `esmaint-net`.

```
[esms1->device[eslogin1*]]% set managementnetwork esmaint-net
```

8. Commit your changes and exit `cmsh`.

```
[esms1->device*[eslogin1*]]% commit
[esms1->device[eslogin1]]% quit
esms1#
```

4.2 Create the Site User Network

Before using a CDL node, you must configure the site user network used by slave CDL nodes and set up the network parameters for each node.

Procedure 48. Configure network parameters for `site-user-net`

This procedure uses the network name `site-user-net`. Substitute the name of your external user (site) network for user access.

1. Log in to the CIMS as `root` and run the `cmsh` command.

```
esms1# cmsh
[esms1]%
```

2. Switch to `network` mode:

```
[esms1]% network
[esms1->network]%
```

3. Display the existing networks.

```
[esms1->network]% list
Name (key)       Type         Netmask bits     Base address     Domain name          IPv6
---------------- ------------ ---------------- ---------------- -------------------- ----
esmaint-net      Internal     16               10.141.0.0       esmaint-net.cluster  no
globalnet        Global       16               0.0.0.0          cm.cluster           no
ib-net           Internal     16               10.149.0.0       ib-net.cluster       no
ipmi-net         Internal     16               10.148.0.0       ipmi.net.cluster     no
site-admin-net   External     24               aaa.bbb.ccc.ddd  your.domain.com      no
```

4. Determine whether a `site-user-net` exists on the CIMS.

   a. If a `site-user-net` already exists on the CIMS, proceed to step 7.

   b. Create `site-user-net` by cloning the `site-admin-net` network.

```
[esms1->network]% clone site-admin-net site-user-net
```

5. The cloned network inherits the same settings as the original network (`site-admin-net`). You must change several settings for `site-user-net`: base address, broadcast address, domain name, gateway, and (if necessary) netmask bits.

a. Display the existing settings.

```
[esms1->network*[site-user-net*]% show
Parameter                     Value
----------------------------- ---------------------
Base address                  aaa.bbb.ccc.ddd
Broadcast address             aaa.bbb.255.255
Domain Name                   your.domain.com
Dynamic range end             0.0.0.0
Dynamic range start           0.0.0.0
Gateway                       aaa.bbb.ccc.ddd
IPv6                          no
Lock down dhcpd               no
MTU                           1500
Management allowed            no
Netmask bits                  24
Node booting                  no
Notes                         <0 bytes>
Revision
Type                          External
name                          site-user-net
```

b. Change the base address.

```
[esms1->network*[site-user-net*]% set baseaddress site-user-netBaseAddress
```

c. Change the broadcast address.

```
[esms1->network*[site-user-net*]% set broadcastaddress site-user-netBroadcastAddress
```

d. Change the domain name.

```
[esms1->network*[site-user-net*]% set domainname site-user-netDomainName
```

e. Change the gateway.

```
[esms1->network*[site-user-net*]% set gateway site-user-netGateway
```

f. If necessary, change the netmask bits.

```
[esms1->network*[site-user-net*]% set netmaskbits NN
```

6. Commit your changes.

```
[esms1->network*[site-user-net*]% commit
```

7. Switch to `device` mode.

```
[esms1->network[site-user-net]% device
```

8. Add an interface to the `site-user-net` network. This example shows the hostname `eslogin1`, the Ethernet port `eth2`, and the example IP address *aaa.bbb.ccc.ddd*. Substitute your CDL node's hostname and IP address when configuring the `eth2` interface. You must repeat this step for each CDL node that is added to the `site-user-net` network.

   Important: If necessary, specify `eth0` instead of `eth2`.

```
[esms1->device]% addinterface -n eslogin1 physical eth2 site-user-net aaa.bbb.ccc.ddd
```

9. Commit your changes.

```
[esms1->device*[eslogin1*]]% commit
```

10. Show the interfaces on `eslogin1`.

```
[esms1->device% use eslogin1
[esms1->device[eslogin1]]% interfaces; list
Type         Network device name  IP               Network
------------ -------------------- ---------------- ----------------
bmc          ipmi0                10.148.0.37      ipmi-net
physical     BOOTIF [prov]        10.141.0.37      esmaint-net
physical     eth2                 aaa.bbb.ccc.ddd  site-user-net
physical     ib0                  10.149.0.37      ib-net
```

11. Display the existing networks.

```
[esms1->device[eslogin1]->interfaces]% network list
Name (key)       Type         Netmask bits     Base address     Domain name          IPv6
---------------- ------------ ---------------- ---------------- -------------------- ----
esmaint-net      Internal     16               10.141.0.0       esmaint-net.cluster  no
globalnet        Global       16               0.0.0.0          cm.cluster           no
ib-net           Internal     16               10.149.0.0       ib-net.cluster       no
ipmi-net         Internal     16               10.148.0.0       ipmi-net.cluster     no
site-admin-net   External     24               aaa.bbb.ccc.ddd  your.domain.com      no
site-user-net    External     24               aaa.bbb.ccc.ddd  your.domain.com      no
[esms1]%
```

12. Exit `cmsh`.

```
[esms1->device[eslogin1]->interfaces]% quit
esms1#
```

4.3 Create the Workload Manager Network (`wlm-net`)

The workload manager network (`wlm-net`) connects the CDL nodes to a switch or gateway and then to the internal login nodes on the Cray system.

Procedure 49. Create the workload manager network (`wlm-net`)

1. Log in to the CIMS as `root` and run the `cmsh` command.

```
esms1# cmsh
[esms1]%
```

2. Switch to `network` mode:

```
[esms1]% network
[esms1->network]%
```

3. Display the existing networks.

```
[esms1->network]% list
Name (key)       Type         Netmask bits     Base address     Domain name          IPv6
---------------- ------------ ---------------- ---------------- -------------------- ----
esmaint-net      Internal     16               10.141.0.0       esmaint-net.cluster  no
globalnet        Global       16               0.0.0.0          cm.cluster           no
ib-net           Internal     16               10.149.0.0       ib-net.cluster       no
ipmi-net         Internal     16               10.148.0.0       ipmi-net.cluster     no
site-admin-net   External     24               aaa.bbb.ccc.ddd  your.domain.com      no
site-user-net    External     24               aaa.bbb.ccc.ddd  your.domain.com      no
[esms1->network]%
```

4. Create `wlm-net` by cloning the `site-admin-net` network.

```
[esms1->network]% clone site-admin-net wlm-net
[esms1->network*[wlm-net*]%
```

5. The cloned network inherits the same settings as the original network (`site-admin-net`). You must change the several settings for the new `wlm-net`: base address, broadcast address, domain name, gateway, and (if necessary) netmask bits.

   a. Display the existing settings.

```
[esms1->network*[wlm-net*]% show
Parameter                    Value
---------------------------- ---------------------
Base address                 aaa.bbb.ccc.ddd
Broadcast address            aaa.bbb.255.255
Domain Name                  your.domain.com
Dynamic range end            0.0.0.0
Dynamic range start          0.0.0.0
Gateway                      aaa.bbb.ccc.ddd
IPv6                         no
Lock down dhcpd              no
MTU                          1500
Management allowed           no
Netmask bits                 24
Node booting                 no
Notes                        <0 bytes>
Revision
Type                         External
name                         wlm-net
```

   b. Change the base address.

```
[esms1->network*[wlm-net*]% set baseaddress wlm-netBaseAddress
```

   c. Change the broadcast address.

```
[esms1->network*[wlm-net*]% set broadcastaddress wlm-netBroadcastAddress
```

   d. Change the domain name.

```
[esms1->network*[wlm-net*]% set domainname wlm-netDomainName
```

   e. Change the gateway.

```
[esms1->network*[wlm-net*]% set gateway wlm-netGateway
```

f. If necessary, change the netmask bits.

```
[esms1->network*[wlm-net*]% set netmaskbits NN
```

6. Commit your changes.

```
[esms1->network*[wlm-net*]% commit
[esms1->network[wlm-net]%
```

7. Verify the `wlm-net` network settings.

```
[esms1->network[wlm-net]]% show
Parameter                     Value
----------------------------- ----------------------------------
Base address                  10.150.0.0
Broadcast address             10.150.255.255
Domain Name                   your.domain.com
Dynamic range end             0.0.0.0
Dynamic range start           0.0.0.0
Gateway                       aaa.bbb.ccc.ddd
IPv6                          no
Lock down dhcpd               yes
MTU                           1500
Management allowed            no
Netmask bits                  24
Node booting                  no
Notes                         <0 bytes>
Revision
Type                          External
name                          wlm-net
[esms1->network[wlm-net]]%
```

8. Switch to `device` mode.

```
[esms1->network[wlm-net]% device
[esms1->device]%
```

9. Add an interface to the `wlm-net` network. This example shows the hostname `eslogin1`, the Ethernet port `eth1`, and the example IP address 10.150.0.1. Substitute your CDL node's hostname and IP address when configuring the `eth1` interface.

   Important: If necessary, specify `eth3` instead of `eth1`. You must repeat this step for each CDL node that is added to the `wlm-net` network.

```
[esms1->device]% addinterface -n eslogin1 physical eth1 wlm-net 10.150.0.1
```

10. Commit your changes.

```
[esms1->device*[eslogin1*]% commit
```

11. Show the interfaces on `eslogin1`.

```
[esms1->device[eslogin1]% interfaces; list
Type         Network device name  IP               Network
------------ -------------------- ---------------- ----------------
bmc          ipmi0                10.148.0.37      ipmi-net
physical     BOOTIF [prov]        10.141.0.37      esmaint-net
physical     eth1                 10.150.0.1       wlm-net
physical     eth2                 aaa.bbb.ccc.ddd  site-admin-net
physical     ib0                  10.149.0.37      ib-net
[esms1->device*[eslogin1]->interfaces]%
```

12. Display the existing networks.

```
[esms1->device[eslogin1]->interfaces]% network list
Name (key)       Type         Netmask bits     Base address     Domain name          IPv6
---------------- ------------ ---------------- ---------------- -------------------- ----
esmaint-net      Internal     16               10.141.0.0       esmaint-net.cluster  no
globalnet        Global       16               0.0.0.0          cm.cluster           no
ib-net           Internal     16               10.149.0.0       ib-net.cluster       no
ipmi-net         Internal     16               10.148.0.0       ipmi-net.cluster     no
site-admin-net   External     24               aaa.bbb.0.0      your.domain.com      no
site-user-net    External     24               aaa.bbb.0.0      your.domain.com      no
wlm-net          External     24               10.150.0.1       your.domain.com      no
```

13. Exit `cmsh`.

```
[esms1->device[eslogin1]->interfaces]% quit
esms1#
```

4.4 Configure Bright Categories for the CDL Nodes

A Bright category controls which software image is used for nodes in that category. During the installation, default categories for each type of login node are created, (`esLogin-XE` and `esLogin-XC`). The default category was assigned to the first CDL node when it was cloned.

The following procedure describes how to customize the `esLogin-XC` category (substitute your site CDL category). The default category for CDL nodes must have a software image, default gateway, default disk setup XML file, and finalize script configured for your site environment.

Procedure 50. Configure category settings for the CDL image

1. Log into the CIMS as `root`.
2. Run the `cmsh` command.

```
esms1# cmsh
[esms1]%
```

3. Switch to `category` mode and list categories and associated software images.

```
[esms1]% category
[esms1->category]% list
Name (key)               Software image
------------------------ ----------------------
default                  default-image
esLogin-XC               default-image
esLogin-XE               default-image
```

4. Use the `esLogin-XC` category (substitute your site CDL category).

```
[esms1->category]% use esLogin-XC
[esms1->category[esLogin-XC]]%
```

5. Set the default gateway for the `esLogin-XC` category. For *gatewayIP*, use the IP address of your site's gateway (usually on `site-user-net`).

```
[esms1->category*[esLogin-XC*]]% set defaultgateway gatewayIP
```

6. Assign a CDL software image (*imagename*) to the `esLogin-XC` category.

```
[esms1->category*[esLogin-XC*]]% set softwareimage imagename
```

Important: Files created or modified by a finalize script must be listed in the `excludelistupdate` exclude list for the category. Software updates will overwrite customized files if the files are not specified in an exclude list for the category. Customized files must also be specified in the `excludelistgrab`, and `excludelistgrabnew` exclude lists to prevent customized files from being copied to the CIMS.

7. Set the install mode for the `esLogin-XC` category to `AUTO`.

```
[esms1->category*[esLogin-XC*]]% set installmode auto
```

8. Commit the changes.

```
[esms1->category*[esLogin-XC*]]%  commit
[esms1->category[esLogin-XC]]%
```

9. (Optional) Change the Bright finalize script for this category. Bright runs the finalize script during node provisioning (node installation) just before turning control over to the software image initialization process.

   The settings in this file:

```
/opt/cray/esms/cray-es-finalize-scripts-XX/default/eslogin_finalize.sh
```

   are added to the `esLogin-XC` (or `esLogin-XE`) category by `ESLinstall`. Changes to `eslogin_finalize.sh` do not occur until they are saved in `cmgui`, or committed using `cmsh`, and the node is rebooted.

   a. Copy the `eslogin_finalize.sh` file.

```
[esms1->category[esLogin-XC]]% quit
esms1# cd /opt/cray/esms/cray-es-finalize-scripts-XX
esms1# mkdir -p etc
esms1# cp -p default/eslogin_finalize.sh etc/site.eslogin_finalize.sh
```

b. Edit the `site.eslogin_finalize.sh` file to make any adjustments.

A finalize script (run before `init`) is used to set a file configuration or to initialize special hardware, sometimes after a hardware check. It is run in order to make software or hardware work before or during the later `init` stage of boot. Use a finalize script to execute commands before `init`, and the commands cannot be stored persistently anywhere else, or it is needed because a choice between (otherwise non-persistent) configuration files must be made based on the hardware before `init` starts.

```
esms1# vi etc/site.eslogin_finalize.sh
esms1# cmsh
[esms1->category]% category use esLogin-XC
[esms1->category[esLogin-XC]]% set finalizescript /opt/cray/esms/cray-es-finalize-scripts-XX\
/etc/site.eslogin_finalize.sh
[esms1->category*[esLogin-XC*]]% commit
```

c. Confirm the finalize script.

```
[esms1->category[esLogin-XC]]% get finalizescript
```

10. (Optional) Change the disk partitions sizes for the CDL node.

The following partition sizes were added to the `esLogin-XC` (or `esLogin-XE`) category from:

```
/opt/cray/esms/cray-es-diskpartitions-XX/default/eslogin-diskfull.xml
```

Changes to `eslogin-diskfull.xml` do not occur until they are saved in `cmgui`, or committed using `cmsh`, and the node is rebooted.

```
swap - 64 GB
/tmp - 32 GB
/var - 64 GB
/ - Remainder of disk
```

a. Quit `cmsh` and copy XML configuration file.

```
[esms1->category[esLogin-XC]]% quit
esms1# cd /opt/cray/esms/cray-es-diskpartitions-XX
esms1# mkdir -p etc
esms1# cp -p default/eslogin-diskfull.xml etc/site.eslogin-diskfull.xml
```

Important: If the default disk setup XML files are updated in a ESM release and the site disk setup XML files have been customized, system administrators must compare the newly released disk setup XML files with the current production disk setup XML files, and merge the changes manually. After the changes have been merged, you must load the updated disk setup file into the Bright database for the node category and reboot the nodes that use that category.

b. Edit the `site.eslogin-diskfull.xml` file to change partition sizes or configuration.

```
esms1# vi  etc/site.eslogin-diskfull.xml
esms1# cmsh
[esms1->category]% category use esLogin-XC
[esms1->category*[esLogin-XE*]]% set disksetup /opt/cray/esms/cray-es-diskpartitions-XX/etc/\
site.eslogin-diskfull.xml
esms1->category[esLogin-XC]]% commit
```

c. Confirm the disk setup is loaded.

```
[esms1->category[esLogin-XC]]% get disksetup
```

4.5 Configure kdump on CDL Nodes (SLES)

kdump is configured on a DMP system by modifying the configuration files on CDL systems. Dump files from slave nodes are stored either on the CIMS using NFS®, or on the slave node local disk. To save dump files to a local disk on a slave node, create a persistent `/var/crash` partition.

Procedure 51. Configure kdump on CDL nodes (SLES)

1. Log in to the CIMS as `root`.
2. Choose a slave node that you can use to test the kdump procedure (in this example `eslogin1`) and clone that slave node's software image. This example clones `ESL-XC-2.2.0-201401160637` to `ESL-XC-2.2.0-kdump`.

```
esms1# cp -pr /cm/images/ESL-XC-2.2.0-201401160637 /cm/images/ESL-XC-2.2.0-kdump
esms1# cmsh
esms1% softwareimage
[esms1->softwareimage]% clone ESL-XC-2.2.0-201401160637 ESL-XC-2.2.0-kdump
[esms1->softwareimage*[ESL-XC-2.2.0-kdump*]]%
```

3. Commit your changes.

```
[esms1->softwareimage*[ESL-XC-2.2.0-kdump*]]% commit
```

4. Create a test category to configure kdump. Switch to `category` mode to create a test category.

```
[esms1->->softwareimage[ESL-XC-2.2.0-kdump]]% category
[esms1->category]%
```

5. Clone an existing CDL category to create a test category. Be sure to clone the default `esLogin-XC` or `esLogin-XE` category to an `esLogin-XC-test` or `esLogin-XE-test` category. These categories have different configurations and software images and are **not** interchangeable. This procedure creates an `esLogin-XC-test` category.

```
[esms1->category]% clone esLogin-XC esLogin-XC-test
[esms1->category*[esLogin-XC-test*]]%
```

6. Assign the kdump CDL image (`ESL-XC-2.2.0-kdump`) to the test category.

```
[esms1->category*[esLogin-XC-test]*]% set softwareimage ESL-XC-2.2.0-kdump
```

7. Add the following line to the exclude lists for the `esLogin-XC-test` category.

   ```
   - /var/crash/*
   ```

   The `vi` editor launches in each command below. Edit and save each of the exclude list files after adding `- /var/crash/*`.

```
[esms1->category*[esLogin-XC-test]*]% set excludelistsyncinstall
[esms1->category*[esLogin-XC-test]*]% set excludelistupdate
[esms1->category*[esLogin-XC-test]*]% set excludelistgrab
[esms1->category*[esLogin-XC-test]*]% set excludelistgrabnew
```

8. Save each file and commit your changes.

```
[esms1->category*[esLogin-XC-test*]% commit
[esms1->category[esLogin-XC-test]%
```

9. Assign the test category and test image to the test node (`eslogin1`) and commit your changes.

```
[esms1->category[esLogin-XC-test]]% device use eslogin1
[esms1->device[eslogin1]]% set category esLogin-XC-test
[esms1->device*[eslogin1*]]% commit
[esms1->device[eslogin1]]%
```

10. If you are saving kdump crash files to the slave node local disk, add the following lines to the slave node's finalize script. This command opens the `vi` editor.

```
[esms1->device[eslogin1]]% set finalizescript

DEV=$( awk -- '{ if ($2 == "/localdisk/var/crash") { print $1; exit 0 } }' < /proc/mounts )
[ -n "$DEV" ] && e2label $DEV crash
```

11. Commit your changes.

```
[esms1->device*]]% commit
```

12. Set the storage location for crash dumps.

    - If crash dumps will be saved to the CIMS proceed to step 13.
    - If crash dumps will be saved to the slave node local disk, proceed to step 14.

13. Save crash files to `/var/crash` on the primary CIMS:

a. Use `fsexports` to determine whether the CIMS is exporting `/var/crash`.

```
[esms1->device]]% use esms1
[esms1->device[esms1]]% fsexports
[esms1>device[esms1]->fsexports]% list
Name (key)                                    Path
--------------------------------------------- ------------------------------
/cm/shared@esmaint-net                        /cm/shared
/home@esmaint-net                             /home
/var/spool/burn@esmaint-net                   /var/spool/burn
/cm/node-installer/certificates@esmaint-net   /cm/node-installer/certificates
/cm/node-installer@esmaint-net                /cm/node-installer
```

b. If `/var/crash` is not exported from the CIMS, then configure and export it to slave nodes.

```
[esms1->device[esms1]->fsexports]% add /var/crash
[esms1->device*[esms1*]->fsexports*[/var/crash*]]% set name /var/crash@esmaint.net
[esms1->device*[esms1*]->fsexports*[/var/crash*]]% set extraoptions no_subtree_check
[esms1->device*[esms1*]->fsexports*[/var/crash*]]% set hosts esmaint-net
[esms1->device[esms1]*]->fsexports*[/var/crash*]]% set write yes
[esms1->device*[esms1*]->fsexports*[/var/crash*]]% commit
[esms1->device[esms1]->fsexports[/var/crash]]%
```

c. Exit `/var/crash` submode.

```
[esms1->device[esms1]->fsexports[/var/crash]]% exit
[esms1>device[esms1]->fsexports]%
```

d. Verify that the CIMS is exporting `/var/crash`.

```
[esms1>device[esms1]->fsexports]% list
Name (key)                                    Path
--------------------------------------------- ------------------------------
/cm/shared@esmaint-net                        /cm/shared
/home@esmaint-net                             /home
/var/spool/burn@esmaint-net                   /var/spool/burn
/cm/node-installer/certificates@esmaint-net   /cm/node-installer/certificates
/cm/node-installer@esmaint-net                /cm/node-installer
/var/crash@esmaint-net                        /var/crash
```

e. Exit `cmsh`.

```
[esms1>device[esms1]->fsexports]% quit
esms1#
```

f. Update the exports.

```
esms1# exportfs -a
```

14. Use the `chroot` shell to edit the `/boot/pxelinux.cfg/default` file in the kdump test image created in step 2 (`ESL-XC-2.2.0-kdump`).

a. Use `chroot` to edit the `/boot/pxelinux.cfg/default` file.

```
esms1# chroot /cm/images/ESL-XC-2.2.0-kdump
esms:/>vi /boot/pxelinux.cfg/default
```

b. Scroll down and locate the following line:

```
# End of documentation, configuration follows:
```

c. Enter the following lines in the `default` configuration file:

```
LABEL kdump
KERNEL vmlinuz
IPAPPEND 3
APPEND initrd=initrd crashkernel=512M CMDS console=tty0 console=ttyS1,115200n8 CMDE
MENU LABEL ^KDUMP      - Normal boot mode with kdump
MENU DEFAULT
```

d. Examine the other `LABEL` entries in the `default` configuration file and remove the line: `MENU DEFAULT`.

e. Exit and save the file.

f. Verify that the `/var/crash` partition exists in the `ESL-XC-2.2.0-kdump` image.

15. Edit the `/etc/sysconfig/kdump` file and modify the following lines:

```
esms1:/> vi /etc/sysconfig/kdump
```

a. Scroll down and locate the `KDUMP_SAVEDIR` entry:

If you want to save crash dump files using NFS to `/var/crash` on the CIMS, modify the `KDUMP_SAVEDIR` line as follows:

```
KDUMP_SAVEDIR="nfs://master/var/crash/"
```

If you want to save crash dump files to the local disk, modify `KDUMP_SAVEDIR` as follows:

```
KDUMP_SAVEDIR="file:///var/crash"
```

Create a persistent partition (`/var/crash`) in the disk setup XML file for the kdump test category (`esLogin-XC-test`). Creating a separate partition for crash dumps on the slave node software image prevents `/var` from filling up and causing problems for the operating system.

b. Locate `KDUMP_DUMPLEVEL` and change it to:

```
KDUMP_DUMPLEVEL=27
```

c. Locate `KDUMP_CONTINUE_ON_ERROR` and change it to:

```
KDUMP_CONTINUE_ON_ERROR="true"
```

d. Locate `KDUMP_NETCONFIG` and change it to you `esmaint-net` interface, `eth0` or `eth2`):

```
KDUMP_NETCONFIG="eth0:dhcp"
```

e. Exit and save the file.

16. Enable the kdump service.

```
esms1:/> chkconfig --set boot.kdump on
```

17. Verify that /lib/mkinitrd/scripts/setup-storage.sh and setup-kdumpfs.sh exist.

    a. If these files do not exist, copy them from the CIMS to /lib/mkinitrd/scripts/ in the /cm/images/ESL-XC-2.2.0-kdump image. You must exit the chroot shell to do this step.

```
esms1:/> exit
esms1# cp -p /lib/mkinitrd/scripts/setup-storage.sh /cm/images/ESL-XC-2.2.0-kdump/lib/mkinitrd/scripts
esms1# cp -p /lib/mkinitrd/scripts/setup-kdumpfs.sh
/cm/images/ESL-XC-2.2.0-kdump/lib/mkinitrd/scripts
```

    b. Return the chroot shell in the ESL image.

```
esms1# chroot /cm/images/ESL-XC-2.2.0-kdump
```

    c. Run mkinitrd_setup to update symbolic links in /lib/mkinitrd/setup and /lib/mkinitrd/boot.

```
esms1:/> mkinitrd_setup
Scanning scripts ...
Resolve dependencies ...
Install symlinks in /lib/mkinitrd/setup ...
Install symlinks in /lib/mkinitrd/boot ...
esms1:/lib/mkinitrd/scripts>
```

18. Exit the chroot shell.

```
esms1:/> exit
esms1#
```

19. Reboot the test node and run kdump.

    a. Start a console window on the test slave node (eslogin1).

```
esms1# cmsh
[esms1]% device; use eslogin1
[esms1->device[eslogin1]]% rconsole
```

    b. Reboot the test node (eslogin1).

```
eslogin1: reboot
eslogin1: Reboot in progress ...
```

    c. When the node reboots, initiate kdump.

```
esms1# ssh eslogin1
eslogin1# echo c > /proc/sysrq-trigger
```

    The dump file is created in /var/crash on the CIMS node if dumping over NFS. The dump file is created in slave node's local /var/crash directory if dumping to the local disk.

20. Make the kdump image the default image for all CDL nodes.

a. Start `cmsh` and assign the test node (`eslogin1`) to the default CDL category `esLogin-XC`.

```
 esms1#  cmsh
[esms1]% device; use eslogin1
[esms1->device[eslogin1]]% set category esLogin-XC
[esms1->device*[eslogin1*]]% commit
```

b. Switch to `category` mode and configure the default `esLogin-XC` category to use the kdump software image.

```
[esms1->device[eslogin1]]% category
[esms1->category]% use esLogin-XC
[esms1->category[esLogin-XC]% set softwareimage ESL-XC-2.2.0-kdump
```

21. Reboot all of the nodes in the `esLogin-XC` category, so that they use the kdump software image.

```
[esms1->category[esLogin-XC]% device
[esms1->device]% reboot -c esLogin-XC
eslogin001: Reboot in progress ...
```

22. Exit `cmsh`.

```
[esms1->device]% quit
```

4.6 Create a kdump Crash Partition on a Slave Node Local Disk

If the slave node does not already have a `/var/crash` partition, then you must create a persistent partition (`/var/crash`) on a slave node's disk setup to store kdump crash files.

Procedure 52. Create a kdump `/var/crash` partition on a slave node

1. Log in to the CIMS as `root` and run the `cmsh` command.

```
esms1# cmsh
[esms1]%
```

2. Clone the production category used for the slave node to create a temporary test category. This procedure creates a category named `esLogin-XC-test`.

```
esms1# category
[esms1->category]% clone esLogin-XC esLogin-XC-test
[esms1->category*[esLogin-XC-test*]]%
```

3. Edit the disk setup XML file for the `esLogin-XC-test` category.

```
[esms1->category*[esLogin-XC-test*]]% set disksetup
```

The `vi` editor starts which enables you to edit the setup XML file. Scroll down in the disk setup XML file and create a new `/var/crash` disk partition.

```
...
<partition id="a3">
    <size>10G</size>
    <type>linux</type>
    <filesystem>ext3</filesystem>
    <mountPoint>/var/crash</mountPoint>
    <mountOptions>defaults,noatime,nodiratime</mountOptions>
</partition>

...
```

4. Commit your changes.

```
[esms1->category*[esLogin-XC-test*]]% commit
[esms1->category[esLogin-XC-test]]%
```

5. Temporarily assign a CDL node (in this example, `eslogin1`) to the `esLogin-XC-test` category.

```
[esms1->category[esLogin-XC-test]]% device
[esms1->device]% use eslogin1
[esms1->device[eslogin1]]% set category esLogin-XC-test
[esms1->device*[eslogin1*]]% commit
[esms1->device*[eslogin1]]% category
[esms1->category]% usedby esLogin-XC-test
Category used by the following:
Type             Name                      Parameter                 Autochange
---------------- ------------------------- ------------------------- ------------
Device           eslogin1                  category                  no
```

6. In a new window log in to the CIMS as `root`, start `cmsh`, and launch a remote console on `eslogin1` while you reboot.

```
esms1# cmsh
[esms1]% device; use eslogin1
[esms1->device[eslogin1]]% rconsole
```

In a another window, reboot the slave node. The node installer will recognize the disk setup has changed, and repartition the disks.

```
[esms1->category[esLogin-XC-test]]% device; use eslogin1
[esms1->device[eslogin1]]% reboot
```

7. Enter `quit` to exit `cmsh`.

8. When the node reboots, SSH to the node and verify that the `/var/crash` partition is available.

```
esms1# ssh eslogin1
Last login: Thu May 23 08:09:22 2014
eslogin1# df
Filesystem        1K-blocks     Used Available Use% Mounted on
/dev/sda1          19685912  9735588   8950324  53% /
udev               65879960      200  65879760   1% /dev
tmpfs              65879960        0  65879960   0% /dev/shm
/dev/sda7         189409992   191948 179596496   1% /local
/dev/sda3           1969424    35784   1833596   2% /tmp
/dev/sda2          19685656   654584  18031088   4% /var
/dev/sda6          17718140   176196  16641900   2% /var/crash
master:/cm/shared 125991776 60142976  59448768  51% /cm/shared
master:/home      217873344 56468864 150337120  28% /home
eslogin1# exit
esms1#
```

9. Start `cmsh` and clone the `esLogin-XC-test` category with the new disk layout to a new category for XC CDL nodes and assign CDL nodes to use the category. To specify a range of nodes use `eslogin[1,2]`, `eslogin[1-10]`.

```
esms1# cmsh
[esms1]% category
[esms1->category]% clone esLogin-XC-test esLogin-crash-paritition
[esms1->category*[esLogin-crash-paritition*]% commit
[esms1->category[esLogin-crash-paritition]% device use eslogin1
[esms1->device[eslogin1]]% set category esLogin-crash-paritition
[esms1->device*[eslogin1*]] commit
[esms1->device[eslogin1]] quit
```

4.7 Set CDL Node Device Parameters in Bright

When you add a new CDL node to Bright, it must be assigned to a Bright category (`esLogin-XE` or `esLogin-XC`, for example) that configures node-specific device information. The physical node and network interface information should already exists in Bright.

Procedure 53. Set CDL node device parameters in Bright

1. Log in to the CIMS as `root` and run the `cmsh` command.

```
esms1# cmsh
[esms1]%
```

2. Switch to `device` mode:

```
[esms1]% device
[esms1->device]%
```

3. Add the new node to the `esLogin-XC` category. This procedure uses the example hostname `eslogin1` and the category name `esLogin-XC`. Substitute your actual CDL hostname and the category name used in the previous procedure.

```
[esms1->device]% use eslogin1
[esms1->device[eslogin1]]% set category esLogin-XC
[esms1->device[eslogin1*]]% commit
[esms1->device[eslogin1]]%
```

4. Set the rack information (device height, position in the rack, and rack index). This example assumes that the device height is 2U, its position in the rack is `0`, and the rack index is `0`. Substitute the actual values for your CDL node.

```
[esms1->device[eslogin1]]% set deviceheight 2
[esms1->device[eslogin1*]]% set deviceposition 0
[esms1->device[eslogin1*]]% set rack 1
```

5. Commit your changes.

```
[esms1->device[eslogin1*]]% commit
[esms1->device[eslogin1]]%
```

6. Exit `cmsh`.

```
[esms1->device[eslogin1]]% quit
esms1#
```

4.8 Configure eswrap

Important: If you are updating or upgrading ESL software, the `eswrap.ini` file in `/opt/cray/eslogin/eswrap/default/etc/` now includes a `slurm` option. Merge the contents of this file with the `/opt/cray/eslogin/eswrap/default/eswrap.ini` file. See Support for Native SLURM on page 180.

The `eswrap` utility is a wrapper that lets users access a subset of Cray Linux Environment (CLE) and Programming Environment (PE) commands from a CDL node. `eswrap` uses Secure Shell (`ssh`) to launch the wrapped command on the Cray system, then displays the output on the CDL node so that it appears to the user that the wrapped command is actually running on the CDL node.

Running `eswrap` creates a symbolic link for each wrapped command in the directory `/opt/cray/eslogin/eswrap/default/bin`. Each symbolic link points to the `eswrap` command, so that running a wrapped command (such as `apstat`) actually runs `eswrap` with the wrapped command as an argument. `eswrap` uses `ssh` to run the command on the specified node of the Cray system (by default, the internal login node with the hostname `login`, unless the `$ESWRAP_LOGIN` environment variable specifies a different node).

The initialization file `eswrap.ini` controls the list of wrapped commands. An administrator can edit this file to include or exclude SLURM commands, Application Level Placement Scheduler (ALPS) commands, STAT (Stack Trace Analysis Tool) commands, Cluster Compatibility Mode (CCM) commands, `aprun`, `qsub`, and selected Cray CLE commands.

See the `eswrap`(8) man page for more information on the `eswrap` configuration variables, including which commands are enabled by default. Use the following commands to display the image-specific version of this CDL man page:

For managed CDL nodes, enter:

```
esms1# cd /cm/images/imagename/opt/cray/eslogin/eswrap/default/man/man8
esms1# man ./eswrap.8
```

For unmanaged CDL nodes, enter:

```
eslogin1# cd /opt/cray/eslogin/eswrap/default/man/man8
eslogin1# man ./eswrap.8
```

Procedure 54. Configure `eswrap`

You must execute the following steps as `root` whenever the `eswrap.ini` file is modified.

1. If configuring an unmanaged CDL node, proceed to step 2. If configuring a managed CDL node, use `chroot` to change to the `root` directory of the CDL image. For *imagename*, substitute the directory name of the CDL image.

```
esms1# chroot /cm/images/imagename
esms1:/>
```

2. Edit the `eswrap.ini` configuration file to enable (wrap) the necessary commands for your environment. Refer to Support for Native SLURM on page 180 for information about support for native SLURM.

```
# vi /opt/cray/eslogin/eswrap/eswrap.ini
```

3. Run the `eswrap` command.

```
# /opt/cray/eslogin/eswrap/default/bin/eswrap --install
```

4. If configuring a managed CDL node, the `chroot` environment.

```
esms1:/> exit
esms1#
```

5. Use this procedure to rerun `eswrap --install` to update the links for the wrapped commands after changes are made to the `eswrap.ini` file.

4.8.1 Support for Native SLURM

The ESL software now supports the Simple Linux™ Utility for Resource Management (SLURM) by wrapping related commands with the native SLURM architecture. If any of these commands are installed on the CDL node, they are moved out of the way before they are wrapped to prevent them from appearing in any paths.

If `slurm=true` is set in the `eswrap.ini` file, the following commands are wrapped when `eswrap --install` runs. If `slurm=false` is set in the `eswrap.ini` file, the following commands are unwrapped and any commands that were moved out of the way are restored when `eswrap --install` runs:

`salloc`	`scontrol`	`smap`	`sshare`
`sattach`	`diag`	`sprio`	`sstat`
`sbatch`	`sinfo`	`squeue`	`strigger`
`sacct`	`sbcast`	`sjobexitmod`	`sreport`
`sview`	`sacctmgr`	`scancel`	`sjstat`
`srun`			

4.9 Update a Managed CDL Node Software Image to SLES11SP3 Using CLE Media

This procedure uses the `CRAYCLEinstall.sh` installation software and the `Cray-CLEbase11SP3-20140319.iso` file (or CLE SLES11SP3 DVD) from CLE release media to update a managed CDL node software image to SLES11SP3. This procedure performs only the SLES upgrade and does not install security/recommended updates or CLE updates. Security/recommended updates and CLE updates are installed during the CLE Support Package installation procedure. See *Installing CLE Support Package on a Cray Development and Login (CDL) Node* (S–2528).

This procedure requires the `xc-sles11sp3-`*5.2.14b12*`.iso` file or `Cray CLE 5.2.UP`*nn* `Software` DVD to obtain the `CRAYCLEinstall.sh` script and other installation software. Make sure to use the correct software version to support your architecture (Cray XE system or Cray XC30 system).

The update sequence is as follows:

1. Copy the `CRAYCLEinstall.sh` installation software from CLE release media to the CIMS node. See Procedure 55 on page 181.
2. Update the CDL node software image to SLES11SP3 using the `CRAYCLEinstall.sh` script and CLE SLES11SP3 media. See Procedure 56 on page 181.

Procedure 55. Copy the CRAYCLEinstall.sh software to the CIMS node

1. Log on to the CIMS node as `root`.

```
remote% ssh root@esms1
```

2. If necessary, create the `/media/cdrom` directory CIMS node.

```
esms1# mkdir -p /media/cdrom
```

3. If necessary, create the `/root/isos` directory on the CIMS node.

```
esms1# mkdir -p /root/isos
```

4. Insert the `Cray CLE 5.2.UP`*nn* software DVD in the optical drive and mount it to `/media/cdrom`.

```
esms1 # mount /dev/cdrom /media/cdrom
```

   Or

   Copy the CLE ISO file (`xc-sles11sp3-`*5.2.14b12*`.iso`) to `/root/isos` on the CIMS node and mount it to `/media/cdrom`. Make sure to use the ISO that supports the correct software environment.

```
esms1# mount -o loop,ro /root/isos/xc-sles11sp3-5.2.14b12.iso /media/cdrom
```

5. Make a temporary directory to store the CLE installation software.

```
esms# mkdir -p /tmp/CLErel
```

6. Copy the `CRAYCLEinstall.sh` software to the `/tmp/`*CLErel* directory on the CIMS node.

```
esms1# cp -pr /media/cdrom/* /tmp/CLErel
```

7. Unmount `/media/cdrom`.

```
esms1# umount /media/cdrom
```

Procedure 56. Update a managed CDL node software image to SLES11SP3

1. Mount the CLE SLES11SP3 release media to `/root/isos` on the on the CIMS node:

   a. If you have the CLE SLES11SP3 release media DVD, insert the CLE SLES11SP3 release media DVD in the optical drive and mount it to `/media/cdrom`.

```
esms1 # mount /dev/cdrom /media/cdrom
```

   b. If you have the CLE SLES11SP3 ISO file (`Cray-CLEbase11SP3-20140319.iso`), copy it to `/root/isos` on the CIMS node and mount it to `/media/cdrom`.

```
esms# mount -o loop,ro /root/isos/Cray-CLEbase11SP3-20140319.iso /media/cdrom
```

2. Change directories to the location of the CRAYCLEinstall.sh script.

```
esms# cd /tmp/CLErel
```

3. The -m option in the following command specifies the mount point for the SLES11SP3 media and the -p option specifies the directory that contains the CLE files. The -u -l command line options trigger the SLES upgrade and not security/recommended updates or CLE updates.

   Important: Do not abort the CRAYCLEinstall.sh program while it performs the SLES11SP3 update. If Ctrl-C interrupts the CRAYCLEinstall.sh program, the RPM database may be in an inconsistent state with duplicate entries. If this happens, restart this step to install remaining RPMs. However, if this step is restarted, the CRAYCLEinstall.sh program will not remove duplicate RPMs in the RPM database for the ESL-XC-2.2.0test software image.

   The following command updates an XC30 ESL software image (ESL-XC-2.2.0test) from SLES11SP1 or SLES11SP2 to SLES11SP3.

```
esms# ./CRAYCLEinstall.sh -m /media/cdrom -v -p `pwd` -u -l -X Aries\
-g /cm/images/ESL-XC-2.2.0test
```

   Note: This step may take more than 20 minutes.

4. Use the following command to determine whether any Novell security/recommended RPMs must be updated.

```
esms# ./CRAYCLEinstall.sh -m `pwd` -v -G -S -X Aries\
-g /cm/images/ESL-XC-2.2.0test
```

   If any RPMs were marked as MISMATCH, NOT INSTALLED, or REMOVE in the output from the previous step, use CRAYCLEinstall.sh to install the Novell security/recommended RPMs on the software image, then use the -G and -S options again to verify the results.

```
esms# ./CRAYCLEinstall.sh -m `pwd` -v -K -S -X Aries\
-g /cm/images/ESL-XC-2.2.0test

esms# ./CRAYCLEinstall.sh -m `pwd` -v -G -S -X Aries\
-g /cm/images/ESL-XC-2.2.0test
```

5. Unmount the release media.

```
esms# umount /media/cdrom
```

6. If the kernel is a lower version that the kernel specified in the SLES11SP3 upgrade (3.0.76-0.11-default), then perform this step to set the kernel version. If the kernel is a higher version, then skip this step.

Start `cmsh` and switch to `softwareimage` mode, and set the kernel version to `3.0.76-0.11-default`.

```
esms1# cmsh
[esms1]% softwareimage
[esms1->softwareimage]% use ESL-XC-2.2.0test
[esms1->softwareimage->ESL-XC-2.2.0test]% get kernelversion
3.0.38-0.5-default
[esms1->softwareimage->ESL-XC-2.2.0test]% set kernelversion 3.0.76-0.11-default
[esms1->softwareimage*->ESL-XC-2.2.0test*]% commit
[esms1->softwareimage->ESL-XC-2.2.0test]%
```

7. Quit `cmsh`.

```
[esms1->softwareimage->ESL-XC-2.2.0test]% quit
esms1#
```

4.10 Upgrade an Unmanaged CDL Node to SLES11SP3 Using CLE Media

This procedure uses the `CRAYCLEinstall.sh` installation software and the `Cray-CLEbase11SP3-20140319.iso` file (or CLE SLES11SP3 DVD) from CLE release media to upgrade the base software on an unmanaged CDL node to SLES11SP3. This procedure performs only the SLES upgrade and does not install security/recommended updates or CLE updates. Security/recommended updates and CLE updates are installed during the CLE Support Package installation procedure. See *Installing CLE Support Package on a Cray Development and Login (CDL) Node* (S–2528).

This procedure requires the `xc-sles11sp3-`*`5.2.14b12`*`.iso` file or `Cray CLE 5.2.UP`*`nn`* `Software` DVD to obtain the `CRAYCLEinstall.sh` installation software. Make sure to use the correct software version to support your architecture (Cray XE system or Cray XC30 system).

The update sequence is as follows:

1. Copy the `CRAYCLEinstall.sh` installation software from CLE release media to the CDL node.
2. Update the CDL node base software to SLES11SP3 using the `CRAYCLEinstall.sh` installation software and CLE SLES11SP3 media.

Procedure 57. Copy the `CRAYCLEinstall.sh` installation software to the CDL node

1. Log on to the CDL node as `root`.

```
remote% ssh root@eslogin1
```

2. If necessary, create the `/media/cdrom` directory CDL node.

```
eslogin1# mkdir -p /media/cdrom
```

3. If necessary, create the `/root/isos` directory on the CDL node.

```
eslogin1# mkdir -p /root/isos
```

4. Insert the `Cray CLE 5.2.UP`*nn* software DVD in the optical drive and mount it to `/media/cdrom`.

```
eslogin1# mount /dev/cdrom /media/cdrom
```

Or

Copy the CLE ISO file (`xc-sles11sp3-`*5.2.14b12*`.iso`) to `/root/isos` on the CDL node and mount it to `/media/cdrom`.

```
eslogin1# mount -o loop,ro /root/isos/xc-sles11sp3-5.2.14b12.iso /media/cdrom
```

5. Make a directory to store the CLE installation software.

```
eslogin1# mkdir -p /tmp/CLErel
```

6. Copy the CLE installation software to the `/tmp/`*CLErel* directory on the CDL node.

```
eslogin1# cp -pr /media/cdrom/* /tmp/CLErel
```

7. Unmount `/media/cdrom`.

```
eslogin1# umount /media/cdrom
```

Procedure 58. Update a unmanaged CDL node to SLES11SP3

1. Copy the CLE SLES11SP3 (`Cray-CLEbase11SP3-20140319.iso` to `/root/isos` on the on the CDL node:

   a. If you have the CLE SLES11SP3 release media DVD, insert the CLE SLES11SP3 release media DVD in the optical drive and mount it to `/media/cdrom`.

```
eslogin1# mount /dev/cdrom /media/cdrom
```

   b. If you have the CLE SLES11SP3 ISO file (`Cray-CLEbase11SP3-20140319.iso`), copy it to `/root/isos` on the CDL node and mount it to `/media/cdrom`.

```
eslogin1# mount -o loop,ro /root/isos/Cray-CLEbase11SP3-20140319.iso /media/cdrom
```

2. Change directories to the location of the `CRAYCLEinstall.sh` script.

```
eslogin1# cd /tmp/CLErel
```

3. The `-m` option in the following command specifies the mount point for the SLES11SP3 media and the `-p` option specifies the directory that contains the CLE files. The `-u -l` command line options trigger the SLES upgrade and not security/recommended updates or CLE updates.

The following command updates an unmanaged XC30 CDL node from SLES11SP1 or SLES11SP2 to SLES11SP3.

```
eslogin1# ./CRAYCLEinstall.sh  -m /media/cdrom -v -p `pwd` -u -l -X Aries -g /
```

4. Use the following command to determine whether any Novell security/recommended RPMs must be updated.

```
eslogin1# ./CRAYCLEinstall.sh -m `pwd` -v -G -S -X Aries -g /
```

If any RPMs were marked as `MISMATCH`, `NOT INSTALLED`, or `REMOVE` in the output from the previous step, use `CRAYCLEinstall.sh` to install the Novell security/recommended RPMs on the software image, then use the `-G` and `-S` options again to verify the results.

```
eslogin1# ./CRAYCLEinstall.sh -m `pwd` -v -K -S -X Aries -g /

eslogin1# ./CRAYCLEinstall.sh -m `pwd` -v -G -S -X Aries -g /
```

5. Unmount the release media.

```
eslogin1# umount /media/cdrom
```

4.11 Update ESL Software on a Managed CDL Node

Procedure 59. Copy the ESL software to the CIMS node

Important: Always install base operating system updates (if necessary) before installing ESL software.

1. Log in to the CIMS as `root`.

```
remote% ssh root@esms1
```

2. If necessary, create the `/media/cdrom` directory.

```
esms1# mkdir -p /media/cdrom
```

3. Mount the ESL XC-2.2.0 or ESL XE-2.2.0 release media to `/media/cdrom` on the CIMS.

   - If you have the release media on DVD, insert the Cray ESL software DVD into the DVD drive and mount it to `/media/cdrom`.

```
esms1# mount /dev/cdrom /media/cdrom
```

   - If you have the release media on an ISO file (for example, `ESL-XC-2.2.0-201401160637a12.iso`) copy the ISO file to the `/root/isos` directory on the CIMS and mount it to `/media/cdrom`.

```
esms1# mount -o loop,ro /root/isos/ESL-XC-2.2.0-201401160637a12.iso /media/cdrom
```

4. Create a `/root/release/`*releasename* directory on the CIMS

and copy all files from the ESL release media to the directory, (`/root/release/ESL-XC-2.2.0-201401160637a12` in this procedure).

```
esms1# mkdir -p /root/release/ESL-XC-2.2.0-201401160637a12
esms1# cp -pr /media/cdrom/* /root/release/ESL-XC-2.2.0-201401160637a12
```

5. Unmount the Cray ESL release media, and if necessary eject the DVD media.

```
esms1# umount /media/cdrom
esms1# eject
```

Procedure 60. Run `ESLinstall`

This procedure installs an ESL software update to a test software image.

1. As `root` on the CIMS, run the `ESLinstall` script to install the Cray ESL software on the test software image (`ESL-XC-2.2.0test` in this procedure).

```
esms1# cd /root/release/ESL-XC-2.2.0-201401160637a12
esms1# ./ESLinstall -v -s ESL-XC-2.2.0test
./ESLinstall: exec to temp script: /tmp/ESLinstall.PID
Installation output will be captured in /var/adm/cray/logs/ESLinstall.PID.log.ESL-XC-2.2.0test
Fri Mar 21 15:56:48 CST 2014: ESLinstall.PID Starting, Version XC-2.2.0-201401160637a12.
. . .
```

   The software is installed to `/cm/images/ESL-XC-2.2.0test`. If no image name is specified, the `ESLinstall` command creates a software image name using the release version string as the image name. To specify a different image name, use the `-s` *imagename* option, as in this example. For more information, run `ESLinstall -h` to see the usage statement.

2. Examine the initial output and note the process ID (PID) of the `ESLinstall` process. `ESLinstall` creates a log file in `/var/adm/cray/logs/ESLinstall.`*PID*`.log.`*imagename* using this PID.

3. When the installation completes, locate the new CDL image in `/cm/images` and note the directory name. You will use this directory in the following procedure when configuring the Bright category for CDL nodes.

```
esms1# ls -1 /cm/images
default-image
ESL-XC-1.3.0
ESL-XC-2.2.0test
ESL-XE-2.0.0
```

4. The ESL software supports the Simple Linux™ Utility for Resource Management (SLURM) by wrapping related commands with the native SLURM commands. If you are updating or upgrading ESL software, the `eswrap.ini` file in `/opt/cray/eslogin/eswrap/default/etc/` now includes a `slurm` option. Merge the contents of this file with the `/opt/cray/eslogin/eswrap/default/eswrap.ini` file.

5. Install CLESP. See *Installing CLE Support Package on a Cray Development and*

Login (CDL) Node (S–2528). Install other Cray development software (CADE or CDT) to fully configure the login node software image. See *Cray Programming Environments Installation Guide* (S–2372).

Procedure 61. Configure a CDL test category in Bright

To test the new software image, create a test category in Bright and assign the test software image (`ESL-XC-2.2.0test`) to a test category. Configure a CDL node to use the test category and reboot the node to test the image. After you are sure that the image boots successfully, you can assign the image to the production category for the CDL nodes. This procedure creates a test category named `esLogin-XC-test` from a default category name `eslogin-XC` that is created when the ESL software is installed.

1. Log into the CIMS as `root` and run the `cmsh` command.

```
remote% ssh root@esms1
esms1# cmsh
[esms1]%
```

2. Switch to `category` mode.

```
[esms1]% category
[esms1->category]%
```

3. Clone the default node category.

```
[esms1->category]% clone esLogin-XC esLogin-XC-test
[esms1->category*[esLogin-XC-test*]%
```

4. Assign the CDL software image under test (`ESL-XC-2.2.0test`), to the `esLogin-XC-test` category.

```
[esms1->category*[eslogin-XC-test*]% set softwareimage ESL-XC-2.2.0test
```

5. Set the install mode for the `ESL-XC-2.2.0test` category to `AUTO`.

```
[esms1->category*[eslogin-XC-test*]% set installmode auto
```

6. Commit the changes.

```
[esms1->category*[esLogin-XC-test*]% commit
[esms1->category[esLogin-XC-test]%
```

7. Switch to `device` mode.

```
[esms1->category[esLogin-XC-test]% device
[esms1->device]%
```

8. Assign a test CDL node (`eslogin1`) to the `esLogin-XC-test` category and commit the change.

```
[esms1->device]% set eslogin1 category esLogin-XC-test
[esms1->device*]% commit
Successfully committed 1 Devices
[esms1->device]%
```

9. Open a console window and reboot the `eslogin1` node.

   a. Open a console window.

```
[esms1->device]%  device use eslogin1
[esms1->device[eslogin1]]%  rconsole
```

   b. Start a new `cmsh` session in another window, then reboot the `eslogin1` node.

```
esms1# cmsh
[esms1]% device use eslogin1
[esms1->device[eslogin1]]%  reboot
eslogin1: Reboot in progress ...
```

10. Verify that `eslogin1` boots without errors.

11. Assign the `eslogin1` node to the production CDL node category. This example uses the category `esLogin-XC`.

```
[esms1->device]% set eslogin1 category esLogin-XC
[esms1->device*]% commit
[esms1->device]%
```

12. Switch to category mode and assign the new CDL image to the production CDL category (in this example `esLogin-XC`).

```
[esms1->device]% category
[esms1->category]% use esLogin-XC
[esms1->category[esLogin-XC]]% set softwareimage imagename
```

13. Set the install mode for the `esLogin-XC` category to `AUTO`.

```
[esms1->category*[esLogin-XC*]% set installmode auto
```

14. Commit the changes.

```
[esms1->category*[esLogin-XC*]% commit
[esms1->category[esLogin-XC]%
```

15. Switch to `device` mode and reboot all the all CDL nodes in the `esLogin-XC` category to deploy the new software image.

```
[esms1->category[esLogin-XC]]% device
[esms1->device]% reboot -c esLogin-XC
```

16. Quit `cmsh`.

```
[esms1->device]% quit
esms1#
```

4.12 Update ESL Software on an Unmanaged CDL Node

Unmanaged CDL nodes (not controlled by a CIMS) can be stand-alone nodes, or disconnected nodes (a CDL node that was originally installed using an CIMS and then later disconnected). This section describes how to update the base operating system software on an stand-alone or disconnected CDL nodes to SLES11SP3.

Procedure 62. Update an unmanaged CDL Node to SLES11SP3 using CLE media

1. Log in as `root` to the CDL node.

```
remote% ssh root@eslogin1
```

2. Create the `/media/cdrom` directory.

```
eslogin1# mkdir -p /media/cdrom
```

3. Mount the Cray ESL release media on the ESL system.

   - If you have the release media on DVD, insert the Cray ESL software DVD into the DVD drive on the CDL and mount it to `/media/cdrom`.

```
eslogin1# mount /dev/cdrom /media/cdrom
```

   - If you have an electronic version of the release media, copy the ISO file to the `/root/isos` directory, and mount the ISO (`ESL-XC-2.2.0-201401160637a12.iso` for example) to `/media/cdrom`.

```
eslogin1# mkdir -p /root/isos
eslogin1# cp -pr path_to_iso/ESL-XC-2.2.0-201401160637a12.iso /root/isos
eslogin1# mount -o loop,ro /root/isos/ESL-XC-2.2.0-201401160637a12.iso /media/cdrom
```

4. Create a directory to store the ESL release files.

```
eslogin1# mkdir -p /root/release/ESL-XC-2.2.0-201401160637a12
```

5. Copy the ESL release files to `/root/release/ESL-XC-2.2.0-201401160637a12`.

```
eslogin1# cp -pr /media/cdrom/* /root/release/ESL-XC-2.2.0-201401160637a12
```

6. As `root` on the CDL node, run the `ESLinstall` script to install the Cray ESL software.

```
eslogin1# cd /root/release/ESL-XC-2.2.0-201401160637a12
eslogin1# ./ESLinstall -v -s local
./ESLinstall: exec to temp script: /tmp/ESLinstall.PID
Installation output will be captured in /var/adm/cray/logs/ESLinstall.PID.log.local
Tue Jan 21 15:56:48 CST 2014: ESLinstall.PID Starting, Version XC-2.2.0-201401160637a12.
. . .
```

7. While the installation runs, examine the initial output and note the process ID (PID) of the `ESLinstall` process. `ESLinstall` creates a log file in `/var/adm/cray/logs/ESLinstall.`*PID*`.log.local` using this PID.

8. Install CLESP. See *Installing CLE Support Package on a Cray Development and Login (CDL) Node* (S–2528). Install other Cray development software (CADE or CDT) to fully configure the login node software image. See *Cray Programming Environments Installation Guide* (S–2372).

4.13 Merge Updates to Disk Setup XML Files

If an ESL release makes updates to disk setup XML files, and you have made site customizations to your CDL disk setup files, you must merge the site customizations with the newly released disk setup XML file. Refer to Merge Updates to Disk Setup XML Files on page 201.

CLFS Administration Tasks [5]

5.1 Create a Generic CLFS Node in Bright

Create a CLFS node by cloning an existing node that is configured and fully functional or the default node (`node001`) in Bright Cluster Manager® (Bright).

Important: Beginning with ESM release XX-3.0.0, MDS node names must contain the string `mds`, and OSS node names must contain the string `oss` so that the single `site.esf_finalize.sh` script can differentiate between the two.

When you do not have a functioning CLFS node, you must clone the default node (`node001`) created during the CIMS installation process. CLFS nodes can then be specialized to function as MDS, OSS, or Tier metadata controller (MDC) or data mover (DM) nodes.

Procedure 63. Creating a generic CLFS node in Bright

1. Log in to the CIMS as `root` and run `cmsh`.

```
esms1# cmsh
[esms1]%
```

2. Enter `device` mode.

```
[esms1]% device
[esms1->device]%
```

3. Perform this procedure for each CLFS node. List the available devices.

```
[esms1->device]% list
Type                 Hostname (key)   MAC                Category   Ip              Network
-------------------- ---------------- ------------------ ---------- --------------- --------------
EthernetSwitch       switch01         00:00:00:00:00:00             10.141.253.1    esmaint-net
MasterNode           esms1            78:2B:CB:40:CE:CA             10.141.255.254  esmaint-net
PhysicalNode         node001          00:00:00:00:00:00  default    10.141.0.1      esmaint-net
```

4. Clone a node to create a new or second CLFS node.

   Important: Always make sure that newly cloned nodes boot from `default-image` without errors before you begin other configuration tasks. When repeating this procedure to create additional CLFS nodes, set the newly cloned node to the `default` category and boot the node using the `default-image` which configures the node in the Bright software.

a. If a fully configured CLFS node exists, clone the configured node to create another MDS or OSS node and assign it to the `default` category. Otherwise proceed to step 4.b.

```
[esms1->device]% clone esfsnode esfs-mds001
Base name mismatch, IP settings will not be modified!
[esms1->device*[esfs-mds001*]]% commit
[esms1->device[esfs-mds001]]% exit
[esms1->device*]% set esfs-mds001 category default
```

b. Clone the default node (`node001`) to create a generic CLFS node and assign it to the `default` category.

```
[esms1->device]% clone node001 esfs-mds001
Base name mismatch, IP settings will not be modified!
[esms1->device*[esfs-mds001*]]% commit
[esms1->device[esfs-mds001]]% exit
[esms1->device*]% set esfs-mds001 category default
```

c. Commit the changes.

```
[esms1->device*]% commit
[esms1->device]%
```

5. Change the interface settings for the new CLFS node.

a. Switch to `interfaces` mode and list interfaces on `esfs-mds001`.

```
[esms1->device]% use esfs-mds001
[esms1->device]% interfaces
[esms1->device[esfs-mds001]->interfaces]% list
Type         Network device name  IP              Network
------------ -------------------- --------------- ----------------
bmc          ipmi0                10.148.0.1      ipmi-net
physical     BOOTIF [prov]        10.141.0.1      esmaint-net
```

b. Set the `BOOTIF` and `ipmi0` interface addresses for the new node. These addresses must be different from those used by the default or original node.

```
[esms1->device*[esfs-mds001*]->interfaces]% set bootif ip 10.141.0.2
[esms1->device*[esfs-mds001*]->interfaces*]% set ipmi0 ip 10.148.0.2
[esms1->device*[esfs-mds001*]->interfaces*]% list
Type         Network device name  IP              Network
------------ -------------------- --------------- ----------------
bmc          ipmi0                10.148.0.2      ipmi-net
physical     BOOTIF [prov]        10.141.0.2      esmaint-net
```

c. Configure the `ib-net` network and interface. Cray recommends that `ib0` on OSS or MDS nodes connect to the IB switch, `ib1` should not be used. Connect `ib2` and `ib3` to the storage array controllers.

```
[esms1->device*[esfs-mds001*]->interfaces*]% add physical ib0
[esms1->device*[esfs-mds001*]->interfaces*[ib0*]]% set network ib-net
[esms1->device*[esfs-mds001*]->interfaces*[ib0*]]% set ip 10.149.0.2
[esms1->device*[esfs-mds001*]->interfaces[ib0]]% show
Parameter                      Value
------------------------------ ----------------------------------------------
Additional Hostnames
Card Type
DHCP                           no
IP                             10.149.0.2
MAC                            00:00:00:00:00:00
Network                        ib-net
Network device name            ib0
Revision
Speed
Type                           physical
```

d. Commit your changes.

```
[esms1->device*[esfs-mds001*]->interfaces*[ib0*]]% commit
```

e. Exit `ib0`.

```
[esms1->device[esfs-mds001]->interfaces[ib0]]%  exit
```

f. Check the results.

```
[esms1->device[esfs-mds001]->interfaces]% list
Type         Network device name  IP               Network
------------ -------------------- ---------------- ----------------
bmc          ipmi0                10.148.0.2       ipmi-net
physical     BOOTIF [prov]        10.141.0.2       esmaint-net
physical     ib0                  10.149.0.2       ib-net
```

g. Exit `interface` mode and return to `device` mode.

```
[esms1->device[esfs-mds001]->interfaces]% exit
```

h. Display the status of the new node.

```
[esms1->device[esfs-mds001]]% status
esfs-mds001 ............... [  DOWN  ]  (Unassigned)
```

The node is unassigned because the MAC address has not been set in Bright.

6. Set the MAC address for the CLFS node (`eth0` on `esmaint-net`).

```
[esms1->device[esfs-mds001]]% set mac MACaddress
```

7. Set the management network to `esmaint-net`.

```
[esms1->device[esfs-mds001*]]% set managementnetwork esmaint-net
```

8. Set the rack information (device height, position in the rack, and rack index). This example assumes that the device height is 2U, its position in the rack is `0`, and the rack index is `0`. Substitute the actual values for the slave node.

```
[esms1->device[esfs-mds001*]]% set deviceheight 2
[esms1->device[esfs-mds001*]]% set deviceposition 0
[esms1->device[esfs-mds001*]]% set rack 1
```

9. Commit your changes and quit `cmsh`.

```
[esms1->device*[esfs-mds001*]]% commit
[esms1->device[esfs-mds001]]% quit
esms1#
```

10. If you are configuring an MDS node, configure the network parameters for the `site-user-network` and then power cycle test boot the node from the `default-image`. If you are configuring an OSS node, power cycle the node to boot it from the `defaut-image`.

5.2 Create `site-user-net` (MDS Nodes Only)

Configure the external site user network (`site-user-net`) which is used by CLFS MDS nodes for authentication services (LDAP, for example) and file permissions (there are no user login accounts on CLFS nodes).

Procedure 64. Configure the site user network (for MDS nodes only)

1. Log in to the CIMS as `root` and run `cmsh`.

```
esms1# cmsh
[esms1]%
```

2. Switch to `network` mode:

```
[esms1]% network
[esms1->network]%
```

3. Display the existing networks.

```
[esms1->network]% list
Name (key)       Type         Netmask bits     Base address     Domain name          IPv6
---------------- ------------ ---------------- ---------------- -------------------  ----
esmaint-net      Internal     16               10.141.0.0       esmaint-net.cluster  no
globalnet        Global       16               0.0.0.0          cm.cluster           no
ib-net           Internal     16               10.149.0.0       ib-net.cluster       no
ipmi-net         Internal     16               10.148.0.0       ipmi-net.cluster     no
site-admin-net   External     24               aaa.bbb.ccc.ddd  your.domain.com        no
```

4. Determine whether `site-user-net` exists on the CIMS

   a. If a `site-user-net` already exists on the CIMS, proceed to step 5.

b. If there is no site user network, create `site-user-net` by cloning the `site-admin-net` network.

```
[esms1->network]% clone site-admin-net site-user-net
[esms1->network*[site-user-net*]%
```

c. The cloned network inherits the same settings as the original network (`site-admin-net`). You must change several settings for the new `site-user-net`: base address, broadcast address, domain name, gateway, and (if necessary) netmask bits. Display the existing settings.

```
[esms1->network*[site-user-net*]% show
Parameter                     Value
----------------------------- --------------------
Base address                  aaa.bbb.ccc.ddd
Broadcast address             aaa.bbb.255.255
Domain Name                   your.domain.com
Dynamic range end             0.0.0.0
Dynamic range start           0.0.0.0
Gateway                       aaa.bbb.ccc.ddd
IPv6                          no
Lock down dhcpd               no
MTU                           1500
Management allowed            no
Netmask bits                  24
Node booting                  no
Notes                         <0 bytes>
Revision
Type                          External
name                          site-user-net
```

d. Change the base address.

```
[esms1->network*[site-user-net*]% set baseaddress site-user-netBaseAddress
```

e. Change the broadcast address.

```
[esms1->network*[site-user-net*]% set broadcastaddress site-user-netBroadcastAddress
```

f. Change the domain name.

```
[esms1->network*[site-user-net*]% set domainname site-user-netDomainName
```

g. Change the gateway.

```
[esms1->network*[site-user-net*]% set gateway site-user-netGateway
```

h. If necessary, change the netmask bits.

```
[esms1->network*[site-user-net*]% set netmaskbits NN
```

i. Commit your changes.

```
[esms1->network*[site-user-net*]% commit
```

5. Switch to `device` mode.

```
[esms1->network[site-user-net]% device
```

6. Add an interface to the `site-user-net` network. This example shows the hostname `esfs-mds001`, the Ethernet port `eth1`, and the example IP address *aaa.bbb.ccc.ddd*. Substitute your CLFS node's hostname and IP address when configuring the `eth1` interface. You must repeat this step for each CLFS node that is added to the `site-user-net` network.

```
[esms1->device]% addinterface -n esfs-mds001 physical eth1 site-user-net aaa.bbb.ccc.ddd
```

7. Commit your changes.

```
[esms1->device*]% commit
```

8. Show the interfaces on `esfs-mds001`.

```
[esms1->device]%  use esfs-mds001
[esms1->device[esfs-mds001]% interfaces; list
Type         Network device name  IP               Network
------------ -------------------- ---------------- ----------------
bmc          ipmi0                10.148.0.2       ipmi-net
physical     BOOTIF [prov]        10.141.0.2       esmaint-net
physical     eth1                 aaa.bbb.ccc.ddd  site-user-net
physical     ib0                  10.149.0.2       ib-net
```

9. Display the existing networks.

```
[esms1->device[esfs-mds001]->interfaces]% network list
Name (key)       Type         Netmask bits     Base address     Domain name          IPv6
---------------- ------------ ---------------- ---------------- -------------------- ----
esmaint-net      Internal     16               10.141.0.0       esmaint-net.cluster  no
globalnet        Global       16               0.0.0.0          cm.cluster           no
ib-net           Internal     16               10.149.0.0       ib-net.cluster       no
ipmi-net         Internal     16               10.148.0.0       ipmi-net.cluster     no
site-admin-net   External     24               aaa.bbb.0.0      your.domain.com      no
site-user-net    External     24               aaa.bbb.0.0      your.domain.com      no
```

10. Exit interface mode `cmsh`.

```
[esms1->device[esfs-mds001]->interfaces]% exit
[esms1->device[esfs-mds001]
```

5.3 Install an ESF Software Image on the CIMS Node

This section describes how to create a CLFS image by installing and customizing CLFS software on the CIMS. The procedures in this chapter describe an initial (full) software installation for a managed CLFS node.

5.3.1 Before You Begin

Configure a CLFS node in Bright by cloning the default node, configuring interfaces, and booting the CLFS node with `default-image` before installing the CLFS software. Make sure that the following tasks have been completed in Bright before you install ESF software.

- The CLFS node has been defined in Bright by cloning the default slave node and changing the network interface settings.

- The CLFS node boots successfully with the `default-image` slave image.

5.3.2 Install the ESF Software

This section describes how to create a CLFS image by installing and customizing CLFS software on the CIMS.

The ESF installation software creates two default categories for CLFS nodes on the CIMS, `esFS-MDS`, and `esFS-OSS` if they do not exist.

Procedure 65. Install an ESL software image on the CIMS

1. Log in to the CIMS as `root`.

```
remote% ssh root@esms1
```

2. If necessary, create the `/media/cdrom` directory.

```
esms1# mkdir -p /media/cdrom
```

3. Download or copy the CentOS 6.4 software ISO file (`CentOS-6.4-x86_64-bin-DVD1.iso`) to `/root/isos`. This ISO file is used during the software installation process.

   a. Insert the DVD into the DVD drive and mount it to `/media/cdrom`.

```
esms1# mount /dev/cdrom /media/cdrom
```

   b. Copy the contents of the CentOS 6.4 DVD to `/root/isos`.

```
esms1# dd if=/dev/cdrom of=/root/isos/CentOS-6.4-x86_64-bin-DVD1.iso
```

4. Download or copy the Bright 6.1 and CentOS 6.4 software ISO file (`bright6.1-centos6.iso`) to `/root/isos`. This ISO file is used during the software installation process.

   a. Insert the DVD into the DVD drive and mount it to `/media/cdrom`.

```
esms1# mount /dev/cdrom /media/cdrom
```

   b. Copy the contents of the Bright 6.1 and CentOS 6.4 DVD to `/root/isos`.

```
esms1# dd if=/dev/cdrom of=/root/isos/bright6.1-centos6.iso
```

5. Mount the ESF release media on the CIMS node.

   - If you have the release media on DVD, insert the Cray ESF software DVD into the DVD drive and mount it to `/media/cdrom`.

```
esms1# mount /dev/cdrom /media/cdrom
```

   - If you have an electronic version of the release media, mount the Cray ESF ISO to `/media/cdrom`. The ISO file name depends on the supported

architecture, release number, and installer version. This procedure shows the example ISO name `ESF-XX-2.2.0-201401151643a03.iso`, and assumes that the ISO file is in the `/root/isos/` directory.

```
esms1# mount -o loop,ro /root/isos/ESF-XX-2.2.0-201401151643a03.iso /media/cdrom
```

6. Copy all files from the ESF release media to the CIMS. Cray recommends storing the release files in a new subdirectory that uniquely identifies the release. The following examples use the directory `/root/release/ESF-XX-2.2.0-201401151643a03.iso`.

```
esms1# mkdir -p /root/release/ESF-XX-2.2.0-201401151643
esms1# cp -pr /media/cdrom/* /root/release/ESF-XX-2.2.0-201401151643
```

7. Unmount the Cray ESF release media. If you are using a physical DVD, also eject the DVD.

```
esms1# umount /media/cdrom
esms1# eject
```

8. Run the `ESFinstall` script to install the Cray ESF software on the CIMS.

```
esms1# cd /root/release/ESF-XX-2.2.0-201401151643
esms1# ./ESFinstall -v
./ESFinstall: exec to temp script: /tmp/ESFinstall.PID
Installation output will be captured in /var/adm/cray/logs/ESFinstall.PID.log.ESF-XX-2.2.0-201401151643
Tue Jun 3 15:56:48 CST 2013: ESFinstall.PID Starting, Version ESF-XX-2.2.0-201401151643-a03.
```

By default, `ESFinstall` uses the release version string for the image directory name and creates the image in `/cm/images`. For example, the previous command creates the image in `/cm/images/ESF-XX-2.2.0-201401151643`. To specify a different image directory name, use the `-s` *imagename* option, where *imagename* is the full path to the ESF software image file. For more information, run `ESFinstall -h` to see the usage statement.

9. While the installation runs, examine the initial output and note the process ID (PID) of the `ESFinstall` process. `ESFinstall` creates a log file in `/var/adm/cray/logs/ESFinstall.`*PID*`.log.`*imagename* using this PID.

10. When the installation completes, locate the new CLFS image in `/cm/images` and note the directory (image) name. You will use this directory in the following sections to customize the image. You will also use the directory name as the name of the image when configuring the Bright category for CLFS nodes.

```
esms1# ls /cm/images
ESF-XX-2.2.0-201401151643  ESL-XC-1.0.2-201302211318  ESL-XE-1.1.1-kdump   default-image
ESL-TEST-4102013           ESL-XE-1.1.1-201211150916  ESL-XE-1.1.1_CLE4.1  default-image.previous
```

Important: Each time you create, add, or modify an image in Bright, enter relevant information in the `Notes` property for the image with either the `cmsh` or `cmgui`. The `cmsh` `softwareimage` mode `set notes` command launches a `vim` editor that enables you to enter relevant information about the image. Enter `:help` in the editor to display help text.

```
esms1# cmsh
[esms1]% softwareimage
[esms1->softwareimage]% list
Name (key)                          Path
----------------------------------- ----------------------------------------
ESF-XX-2.2.0-201401151643           /cm/images/ESF-XX-2.2.0-201401151643
default-image                       /cm/images/default-image
[esms1->softwareimage]% use ESF-XX-2.2.0-201401151643
[esms1->softwareimage[ESF-XX-2.2.0-201401151643]]% set notes

(in vim editor) created new ESF-XX-2.2.0-201401151643 image - Jane Doe, 6-15-2013.
[esms1->softwareimage*[ESF-XX-2.2.0-201401151643*]]% commit
[esms1->softwareimage[ESF-XX-2.2.0-201401151643]]% quit
```

11. Verify the `lustre` service is running for the ESF software image.

    a. Use the `chroot` shell to open a shell in the ESF software image and verify the `lustre` service is enabled.

```
esms1# chroot /cm/images/ESF-XX-2.2.0-201401151643
[root@esms1 /]# chkconfig --list lustre
lustre 0:off 1:off 2:on 3:on 4:on 5:on 6:off
[root@esms1 /]# exit
esms1#
```

    If the `lustre` is not enabled, start it with the `chkconfig lustre on` command.

12. (Optional) If the CLFS node supports Fibre Channel host bus adapters, set the software image kernel parameter `rdloaddriver` to `scsi_dh_rdac`. See SCSI RDAC Driver Kernel Parameters for Fibre Channel Storage on page 244.

```
esms1# cmsh
[esms1]% softwareimage
[esms1->softwareimage]% use ESF-XX-2.2.0-201401151643
[esms1->softwareimage[ESF-XX-2.2.0-201401151643]]% show
Parameter                      Value
------------------------------ ------------------------------------------------
Boot FSPart                    98784247966
Creation time                  Thu, 23 Jan 2014 12:43:46 CST
Enable SOL                     yes
FSPart                         98784247966
Kernel modules                 <37 in submode>
Kernel parameters
Kernel version                 2.6.32-358.18.1.el6.x86_64
Locked                         no
Name                           ESF-XX-2.2.0-201401151643
Notes                          <0 bytes>
Path                           /cm/images/ESF-XX-2.2.0-201401151643
Revision
SOL Flow Control               no
SOL Port                       ttyS1
SOL Speed                      115200
[esms1->softwareimage[ESF-XX-2.2.0-201401151643]]%  set kernelparameters rdloaddriver=scsi_dh_rdac
[esms1->softwareimage*[ESF-XX-2.2.0-201401151643*]]% commit
[esms1->softwareimage[ESF-XX-2.2.0-201401151643]]% show
Parameter                      Value
------------------------------ ------------------------------------------------
Boot FSPart                    98784247966
Creation time                  Thu, 23 Jan 2014 12:43:46 CST
Enable SOL                     yes
FSPart                         98784247966
Kernel modules                 <37 in submode>
Kernel parameters              rdloaddriver=scsi_dh_rdac
Kernel version                 2.6.32-358.18.1.el6.x86_64
Locked                         no
Name                           ESF-XX-2.2.0-201401151643
Notes                          <0 bytes>
Path                           /cm/images/ESF-XX-2.2.0-201401151643
Revision
SOL Flow Control               no
SOL Port                       ttyS1
SOL Speed                      115200
[esms1->softwareimage]% quit
```

5.4 Recommended MDS/MGT Volume Size

Cray recommends an MGT volume size of 1 GiB.

5.5 QLogic Switch Fibre Channel CLI Utilities

QLogic Fibre Channel QConvergeConsole CLI software is provided with the ESF software image in `/opt/QLogic_Corporation/QConvergeConsoleCLI`. QConvergeConsole CLI is used to configure and manage QLogic Fibre Channel adapters. To start QConvergeConsole CLI in interactive mode, issue the following commands:

Procedure 66. Start QConvergeConsole CLI FC adapter software

1. Log in to the CIMS as `root`.
2. SSH to the CLFS node.

```
esms1# ssh esfs-mds1
esfs-mds1#
```

3. The following command starts `qaucli` in interactive mode.

```
esfs-mds1# /opt/QLogic_Corporation/QConvergeConsoleCLI/qaucli
Loading iSCSI Data ...

        QConvergeConsole

        CLI - Version 1.1.0 (Build 51)

    Main Menu

    1:  Adapter Information
    2:  Adapter Configuration
    3:  Adapter Updates
    4:  Adapter Diagnostics
    5:  Adapter Statistics
    6:  Refresh
    7:  Help
    8:  Exit

        Please Enter Selection:
```

5.6 Merge Updates to Disk Setup XML Files

If the disk setup XML files are changed for a DMP release, and if the disk setup XML files are customized for your site, then the newly released disk setup XML file updates must be merged with the site customizations. Then, the disk setup XML files must be loaded into each category in Bright. Failure to load these files could result in corruption of Lustre MGT or MDT. For each Cray Development and Login (CDL) and CLFS category in Bright, perform the following procedure:

Procedure 67. Merge updates to disk setup XML files

1. Log in to the CIMS as `root` and run `cmsh`.

```
esms1# cmsh
[esms1]%
```

2. Switch to `category` mode and list the node categories:

```
[esms1]% category
[esms1->category]% list
Name (key)               Software image
------------------------ ----------------------
default                  default-image
esFS-MDS                 ESF-XX-1.2.0-2013062112+
esFS-MDS-kdump           ESF-XX-1.2.0-kdump
esFS-MDS-test            centos-mds-20110810-ima+
esFS-OSS                 centos-oss-2.2-20120808
esLogin-XC               default-image
esLogin-XE               ESL-XE-2.0.0-2013060613+
esLogin-kdump            ESL-XC-trunk-kdump
esfs-even-scratch        ESF-XX-1.2.0-2013062112+
esfs-failed-scratch      ESF-XX-1.2.0-2013062112+
esfs-odd-scratch         ESF-XX-1.2.0-2013062112+
```

3. For each category listed in step 2, select the category (`esfs-even-scratch` in this example) and copy the disk setup XML file to a temporary file.

   a. Select the category and copy its disk setup XML file.

```
[esms1->category]% use esFS-MDS
[esms1->category[esfs-even-scratch]]% get disksetup > /tmp/esfs-even-scratch.disksetup.xml
```

   b. Compare the current disk setup to the following files and determine the appropriate file to load into Bright.

   For a CLFS node whose internal disk size is greater than 300 GB use:

```
/opt/cray/esms/cray-es-diskpartitions-XX/default/esfs-diskfull.xml
```

   For a CLFS node whose internal disk size is less than 300 GB use:

```
 /opt/cray/esms/cray-es-diskpartitions-XX/default/esfs-small-diskfull.xml
```

   For a CDL node whose internal disk size is greater than 300 GB use:

```
/opt/cray/esms/cray-es-diskpartitions-XX/default/eslogin-diskfull.xml
```

   For a CDL node whose internal disk size is less than 300 GB use:

```
/opt/cray/esms/cray-es-diskpartitions-XX/default/eslogin-small-diskfull.xml
```

   c. Make any site customizations as needed.

4. Load the disk setup XML file into Bright for the `esfs-even-scratch` category.

```
[esms1->category[esfs-even-scratch]]% set disksetup
/opt/cray/esms/cray-es-diskpartitions-XX/default/esfs-diskfull.xml
[esms1->category*[esfs-even-scratch*]]%
```

5. Confirm the disk setup XML file is loaded.

```
[esms1->category*[esfs-even-scratch*]]% get disksetup
/opt/cray/esms/cray-es-diskpartitions-XX/default/esfs-diskfull.xml
```

6. Commit the change.

```
[esms1->category*[esfs-even-scratch*]]% commit
```

7. Repeat this procedure for each category affected by the new disk setup XML file.

5.7 Configure CLFS Failover (esfsmon 2.0.0)

The Lustre® file system uses multiple Cray CLFS server nodes to supply data storage services to provide a unified view of a cluster file system. As such, there are multiple points of failure. However, there are also multiple routes for data flow and multiple services that can respond to I/O service requests in the event of failure.

As file system infrastructures become very large and complex, failures are inevitable. Failures are detected by hardware, software, or firmware within components or system that comprise the Lustre file system. Generally, a good failover strategy is based on the expectation that failures are spread over time fairly evenly and that there is a reasonable amount of time for human intervention to correct failures (replacing hard drives).

The Cray failover strategy is to maintain the whole system as fully functional and operating without on-going errors. When a failure occurs, the automated failover mechanisms move operations in the failed path to other functional resources and provide notification to the system administrator.

Refer to the *Bright Cluster Manager 6.1 Administrator Manual* PDF file on the CIMS node for more information about monitoring configurations for the system.

5.7.1 Storage Configuration Overview

Multiple Lustre file systems are supported on external storage nodes. The resiliency features in CLFS node failover ensure that under normal conditions, failures will not result in the loss of access to the file system or the loss of data. These features are:

- OSS path redundancy to the storage arrays via failover driver software
- RAID rebuild after a disk failure
- OSTs fail over when an OSS fails
- MDT failover when an MDS fails
- Continued file system access after loss of an LNET router network connection

5.7.2 Failover Conditions

The following conditions trigger a failover action:

- Power failure to a node

- Failure to TCP ping the node
- Failure of a node to LNET ping at least one other node
- Failure of a node to have the expected complement of mounts available
- Failure of a node to respond to Bright management daemon (CMDaemon) requests

5.7.3 Failover Functional Tests

Functional tests are performed on all nodes according to their assigned Bright category (described below).

- All nodes in a particular category are tested in parallel
  - Failed nodes are not tested
  - Some tests do not apply to the standby MDS node unless it is acting as the primary
- Test descriptions
  - **DRAC function** Verify connectivity with the DRAC by checking power status of each node.
  - **IP connectivity with CIMS** Ping each node from the CIMS over the management network.
  - **LNET ping** MDS nodes will attempt to LNET ping two OSS nodes. OSS nodes will attempt to ping two MDS nodes, if available, or the MDS and an OSS node. Failure of both ping attempts is a failure. Status of the IB port is also checked.
  - **Lustre Mounts** Check that the appropriate lustre mounts are mounted.
  - **DeviceIsUp** Check that the server is responding to Bright management daemon requests. General operating system health check.

5.7.4 HA/Failover of I/O Paths for Servers Connected to Storage Arrays

The driver for the RAID arrays supports path failover when redundant paths between an OSS and its attached array fails. When the path failure has been corrected, failback is automatic.

5.7.5 Disk Failure and Rebuild: RAID Hardware Capability

Disk failure detection and automatic rebuild of the replaced drive are features of the RAID array in each LUN. The only manual portion of the repair process is the physical replacement of the failed disk drive.

5.7.6 Failover Features and Bright Monitoring

Bright software provides a monitoring framework that enables administrators to:

- Inspect monitoring data to the required level for existing resources
- Configure gathering of monitoring data for new resources
- See current and past problems or abnormal behavior
- Notice trends that help the administrator predict likely future problems
- Manage current and likely future problems by triggering alerts, taking necessary actions to improve the situation, or investigate further

CLFS failover under Bright Cluster Manager® (Bright) consists of five parts: a monitor script, an action script, a failback command, a configuration file, and the `lustre_control` utility.

- `esfsmon_healthcheck` is the monitoring script which resides in `/cm/local/apps/cmd/scripts/healthchecks` on the CIMS node. Use the monitoring `healthchecks` mode in Bright to configure the `esfsmon_healthcheck` monitoring script. A separate `esfsmon` monitor process runs for each Lustre file system. The monitoring process runs on the CIMS as a Bright health check every 120 seconds by default and can be reconfigured. The `esfsmon_healthcheck` monitoring script executes commands on the CIMS and all monitored nodes via the Bright daemon (CMDaemon) to assess the ability of each node to serve the Lustre file system. Status messages are sent to both the Bright software and `/var/log/messages` on the CIMS.
- `esfsmon_action` is the action script located in `/cm/local/apps/cmd/scripts/action` on the CIMS and is configured as the `esfsmon_action` monitoring action in Bright.`esfsmon_action` runs on the CIMS as a Bright healthcheck action when the `esfsmon healthcheck` reports a failure, and takes appropriate failover action if the esfsmon 2.0.0 health check is not in `RUNSAFE` mode. `esfsmon_action` executes `lustre_control` on the CIMS node to affect an MDS or OSS failover. Status messages are sent to both the Bright software and `/var/log/messages`.
- `esfsmon_failback` is the failback command which runs on the CIMS to bring previously failed CLFS server nodes back to service after any faults have been corrected.
- `esfsmon.conf` is the configuration file that contains environmental variables and file system definitions used by `esfsmon`.
- `lustre_control` runs on the CIMS to perform the failover and failback of the Lustre assets.

5.7.6.1 esfsmon_healthcheck Monitor Testing Sequence

The testing sequence of the esfsmon_healthcheck monitor is as follows:

- Validate configuration
  - Verify that CLFS nodes exist
  - Determine whether monitoring should be suspended. Existence of /var/esfsmon/esfsmon_suspend_*filesystem* file indicates a suspended mode. This is entered automatically during the failover process to prevent false positive error indications.
- Functional testing by node category. Nodes must be in the Bright categories shown below:

  esfs-odd-*filesystem*

  All odd numbered nodes in *filesystem*

  esfs-even-*filesystem*

  All even numbered nodes in *filesystem*

  esfs-failed-*filesystem*

  All failed nodes in *filesystem*

5.7.7 Configure esfsmon_healthcheck Monitor

The esfsmon_healthcheck monitor is installed as a master node health check in Bright. It is available to monitor any Lustre file system whose MDS and OSS servers are managed by a CIMS server running Bright. The following procedures describe how to configure esfsmon 2.0.0 to monitor a Lustre file system and how to view the esfsmon 2.0.0 configuration.

5.7.7.1 Install esfsmon_healthcheck and esfsmon_action Scripts

Before esfsmon can be configured and used, the esfsmon_healthcheck and esfsmon_action scripts must be installed in Bright. The following procedure installs the esfsmon_healthcheck and esfsmon_action scripts in Bright. This procedure is performed once for all the Lustre file systems that will be monitored.

Procedure 68. Install esfsmon_healthcheck and esfsmon_action

1. Log in to the CIMS as root and start cmsh.

```
esms1# cmsh
[esms1]%
```

2. Switch to `monitoring` mode and add an `esfsmon` health check.

```
[esms1]% monitoring healthchecks
[esms1->monitoring->healthchecks]% add esfsmon
[esms1->monitoring->healthchecks*[esfsmon*]]% show
Parameter                      Value
------------------------------ ---------------
Class of healthcheck           misc
Command
Description
Disabled                       no
Extended environment           no
Name                           esfsmon
Notes                          <0 bytes>
Only when idle                 no
Parameter permissions          optional
Revision
Sampling method                samplingonnode
State flapping count           7
Timeout                        5
Valid for                      node,headnode
```

3. Set the command.

```
[esms1->monitoring->healthchecks]% set command
/cm/local/apps/cmd/scripts/healthchecks/esfsmon_healthcheck
[esms1->monitoring->healthchecks*[esfsmon*]]% set description "CLFS Lustre Filesystem Monitor"
[esms1->monitoring->healthchecks*[esfsmon*]]% set parameterpermissions required
[esms1->monitoring->healthchecks*[esfsmon*]]% set timeout 120
[esms1->monitoring->healthchecks*[esfsmon*]]% show
Parameter                      Value
------------------------------ ---------------------------------------------------------------
Class of healthcheck           misc
Command                        /cm/local/apps/cmd/scripts/healthchecks/esfsmon_healthcheck
Description                    CLFS Lustre Filesystem Monitor
Disabled                       no
Extended environment           no
Name                           esfsmon
Notes                          <0 bytes>
Only when idle                 no
Parameter permissions          required
Revision
Sampling method                samplingonnode
State flapping count           7
Timeout                        120
Valid for                      node,headnode
```

4. Commit your changes.

```
[esms1->monitoring->healthchecks*[esfsmon*]]% commit
[esms1->monitoring->healthchecks[esfsmon]]%
```

5. Configure `esfsmon_action`.

```
[esms1->monitoring->healthchecks[esfsmon]]% monitoring actions
[esms1->monitoring->actions]% add esfsmon_action
[esms1->monitoring->actions*[esfsmon_action*]]% show
Parameter                      Value
------------------------------ ------------------
Command
Description
Name                           esfsmon_action
Revision
Run on                         headnode
Timeout                        5
isCustom                       yes
```

6. Set the command and parameters.

```
[esms1->monitoring->actions*[esfsmon_action*]]% set command
/cm/local/apps/cmd/scripts/actions/esfsmon_action
[esms1->monitoring->actions*[esfsmon_action*]]% set timeout 900
[esms1->monitoring->actions*[esfsmon_action*]]% set description "Action for esfsmon failures"
[esms1->monitoring->actions*[esfsmon_action*]]% show
Parameter                      Value
------------------------------ --------------------------------------------------
Command                        /cm/local/apps/cmd/scripts/actions/esfsmon_action
Description                    Action for esfsmon failures
Name                           esfsmon_action
Revision
Run on                         headnode
Timeout                        900
isCustom                       yes
```

7. Commit your changes.

```
[esms1->monitoring->actions*[esfsmon_action*]]% commit
[esms1->monitoring->actions[esfsmon_action]]%
```

5.7.8 Configure `esfsmon.conf`

The `esfmon.conf` file configures esfsmon 2.0.0 to monitor a file system (in this example, the file system is `scratch`).

Edit the `esfmon.conf` and make the following changes:

```
ESFSMON_STATE_DIR=/var/esfsmon
```

Path to esfsmon 2.0.0 operational state (DO NOT CHANGE!)

```
ESFSMON_DATA_DIR=/tmp/esfsmon
```

Path to esfsmon 2.0.0 operational data (DO NOT CHANGE!)

```
ESFSMON_LUSTRE_CONTROL=/opt/cray/esms/cray-lustre-control-XX/default/bin/lustre_control
```

Full path to `lustre_control`

`ESFSMON_SUSPEND_BASE=$ESFSMON_STATE_DIR/esfsmon_suspend_`*LustreFilesystem*

Flag indicating that `esfsmon` is in a suspended state. Existence of this file indicates suspended state. Each esfsmon 2.0.0 instance will append the Lustre file system name to the file.

`ESFSMON_RUNSAFE_BASE=$ESFSMON_STATE_DIR/esfsmon_runsafe_`*LustreFilesystem*

Flag indicating that esfsmon 2.0.0 is in a monitor-only (RUNSAFE) mode. Existence of this file indicates monitor-only (RUNSAFE) mode. Each `esfsmon` instance will append the Lustre file system name to the file.

`ESFSMON_DNE_BASE=$ESFSMON_STATE_DIR/esfsmon_dne_`*LustreFilesystem*

Flag indicating that the file system is in DNE mode.

`ESFSMON_MDS_FO_DISABLED_BASE=$ESFSMON_STATE_DIR/esfsmon_mds_fo_disabled_`*LustreFilesystem*

Flag indicating that MDS fail over (FO) is disabled. Existence of this file indicates MDS FO is disabled. Each esfsmon 2.0.0 instance will append the Lustre file system name to the file.

`ESFSMON_FO_DATA_BASE=$ESFSMON_STATE_DIR/esfsmon_fo_`*LustreFilesystem*

This file contains the hostname of the failed node. Each esfsmon 2.0.0 instance will append the Lustre file system name to the file.

`ESFSMON_DEBUG_BASE=$ESFSMON_STATE_DIR/esfsmon_debug_`*LustreFilesystem*

Flag indicating whether to run debug mode or not. Each esfsmon 2.0.0 instance will append the Lustre file system name to the file.

`ESFSMON_EVEN_CAT_`*filesystem*`=esfs-even-filesystem`

Category for all even-numbered nodes in *filesystem*.

`ESFSMON_ODD_CAT_`*filesystem*`=esfs-odd-filesystem`

Category for all odd-numbered nodes in *filesystem*.

`ESFSMON_FAILED_CAT_`*filesystem*`=esfs-failed-filesystem`

Category for all odd-numbered nodes in *filesystem*.

`ESFSMON_BASENAME_`*filesystem=base-hostname*

CLFS node base name for each file system. Variable name is `ESFSMON_BASENAME_`*filesystem=base-hostname*. For example, `lustre1-mds001` and `lustre1-oss001` have a basename of `lustre1-`.

`ESFSMON_FO_MDS_`*filesystem*

Hostname of the failover MDS node for *filesystem* if **not** in DNE mode.

`CMSH="/cm/local/apps/cmd/bin/cmsh -c"`

Command path and argument to run `cmsh` commands.

`CMSH="/cm/local/apps/cmd/bin/cmsh -c"`

Command path and argument to run `cmsh` commands.

`ESFSMON_IB_FABRIC_`*filesystem*

Identifies which LNet file system that `esfsmon_healthcheck` is using. `esfsmon_healthcheck` supports a shared MDS/MGS on multiple LNet fabrics. For example, there may be *N* file systems that share a single MGS server which is also the failover MDS for each file system. Each file system may have its own `o2ib` fabric, for example `o2ib1..o2ib`*N*. The `esfsmon lnet ping` determines which of the *N* NIDs it must ping for each file system.

5.7.9 Activate esfsmon 2.0.0

Make sure the file system-specific support files are in place before activating esfsmon 2.0.0 in Bright. These include the esfsmon 2.0.0 suspend and `RUNSAFE` control files in `/var/esfsmon` for the file system being monitored. Without these in place, an inadvertent failover may be attempted when esfsmon 2.0.0 is activated for the file system.

The following files must exist on the CIMS node before configuring esfsmon 2.0.0 for a file system.

- `/var/esfsmon/esfsmon_suspend_`*filesystem* — This file suspends monitoring of *filesystem*
- `/var/esfsmon/esfsmon_runsafe_`*filesystem* — This file puts `esfsmon` into `RUNSAFE` (monitor only) mode of *filesystem*

5.7.10 Configure esfsmon 2.0.0 Health Check in Bright

Procedure 69. Configure esfsmon 2.0.0 health check in Bright

1. Log in to the CIMS as `root` and run the `cmsh` command.

```
esms1# cmsh
[esms1]%
```

2. Switch to health check configuration mode for the CIMS.

```
[esms1]% monitoring setup healthconf headnode
[esms1->monitoring->setup[HeadNode]->healthconf]%
```

3. List the health checks that are configured.

```
[esms1->monitoring->setup[HeadNode]->healthconf]% list
HealthCheck              HealthCheck Param  Check Interval
------------------------ ------------------ ------------------------
DeviceIsUp                                  120
ManagedServicesOk                           120
chrootprocess                               900
cmsh                                        1800
diskspace                2% 10% 20%         1800
exports                                     1800
failedprejob                                900
failover                                    1800
interfaces                                  1800
ldap                                        1800
mounts                                      1800
mysql                                       1800
ntp                                         300
oomkiller                                   1800
schedulers                                  1800
smart                                       1800
[esms1->monitoring->setup[MasterNode]->healthconf]%
```

4. Use the following commands to add a monitor for a file system named `scratch`. The first command adds another instance of the `esfsmon` healthcheck without any parameters. This base is configured for `scratch`. Note that configuration changes do not take effect until they are committed to the Bright database with the `cmsh commit` command.

```
[esms1->monitoring->setup[HeadNode]->healthconf]% add esfsmon
[esms1->monitoring->setup*[HeadNode*]->healthconf*[esfsmon*]]% show
Parameter                      Value
------------------------------ ------------------------------------------------
Check Interval                 120
Disabled                       no
Fail Actions
Fail severity                  10
GapThreshold                   2
HealthCheck                    esfsmon
HealthCheckParam
LogLength                      3000
Only when idle                 no
Pass Actions
Stateflapping Actions
Store                          yes
ThresholdDuration              1
Unknown Actions
Unknown severity               10
[esms1->monitoring->setup*[HeadNode*]->healthconf*[esfsmon*]]%
```

5. Set the fail action and the health check parameters to `scratch`.

```
[esms1->monitoring->setup*[HeadNode*]->healthconf*[esfsmon*]]% set checkinterval 60
[esms1->monitoring->setup*[HeadNode*]->healthconf*[esfsmon*]]% set failactions esfsmon_action
[esms1->monitoring->setup*[HeadNode*]->healthconf*[esfsmon*]]% set healthcheckparam scratch
[esms1->monitoring->setup*[HeadNode*]->healthconf*[esfsmon*]]% show
Parameter                      Value
------------------------------ ------------------------------------------------
Check Interval                 120
Disabled                       no
Fail Actions                   enter: esfsmon_action()
Fail severity                  10
GapThreshold                   2
HealthCheck                    esfsmon
HealthCheckParam               scratch
LogLength                      3000
Only when idle                 no
Pass Actions
Stateflapping Actions
Store                          yes
ThresholdDuration              1
Unknown Actions
Unknown severity               10
[esms1->monitoring->setup*[HeadNode*]->healthconf*[esfsmon*]]% commit
[hopms->monitoring->setup[HeadNode]->healthconf[esfsmon]]% quit
esms1#
```

6. The esfsmon 2.0.0 monitor for the `scratch` file system is setup but is suspended due to the existence of the`/var/esfsmon/esfsmon_suspend_scratch` file. Activate monitoring the scratch file system by removing `/var/esfsmon/esfsmon_suspend_scratch`. To enable failover action to take place, remove the `/var/esfsmon/esfsmon_runsafe_scratch` file.

5.7.11 Control esfsmon 2.0.0 Bright Failover Monitor

The Bright failover monitor and failover action are controlled by the existence or non-existence of the following files in the `/var/esfsmon` directory on the CIMS node:

- `esfsmon_suspend_`*filesystem*
  - If present, monitoring of file system is suspended.
  - Set by the `esfsmon_action` script during failover to prevent false positive failure indications.
  - Set by the `esfsmon_failback` script during failback to prevent false positive failure indications.
  - May be set by the administrator to manually suspend monitoring. If manually set, it must be manually removed.
- `esfsmon_runsafe_`*filesystem*
  - If present, monitoring of *filesystem* is in RUNSAFE mode.

  - Errors will be reported but no failover action will be performed.
  - May be set by the administrator to manually enter RUNSAFE mode.
  - If set, it must be manually removed to de-activate RUNSAFE mode and enable failover action.
- `esfsmon_mds_fo_disabled_`*filesystem*
  - If present, failover of MDS nodes is disabled.
  - May be set by the administrator to manually disable MDS failover.
  - If set, it must be manually removed to enable MDS failover.
- `esfsmon_dne_`*filesystem*
  - If present, esfsmon 2.0.0 treats *filesystem* as a DNE configuration with active/active MDS pairs.

5.7.12 Avoid Inadvertent Failover Operations

In order to ensure that inadvertent failover actions are avoided, the administrator should suspend esfsmon 2.0.0 by touching `/var/esfsmon/esfsmon_suspend_`*filesystem* during operations where any of the following may occur.

- Powering off or powering on a node
- Loss of InfiniBand® connectivity to a node
- Loss of Lustre mounts or change in the number of Lustre mounts from the normal value
- Performing a manual failover

Remove the `/var/esfsmon/esfsmon_suspend_`*filesystem* file to resume failover monitoring when the above conditions no longer apply.

5.7.13 Tune the esfsmon 2.0.0 File System Check Interval

The esfsmon 2.0.0 monitor is configured in Bright as the `esfsmon` health check on the CIMS. It takes a file system name as a parameter. Each esfsmon 2.0.0 monitor is named `esfsmon:`*filesystem* in Bright and can be tuned separately. The following commands will list the current check intervals (in seconds) of the master node health checks. The suggested check interval for `esfsmon` is 120 seconds and should not be changed without consulting Cray.

Procedure 70. Tuning the `esfsmon:`*filesystem* check interval

1. Log in to the CIMS as `root` and run the `cmsh` command.

```
esms1# cmsh
[esms1]%
```

2. Show health check intervals for the CIMS.

```
[esms1]% monitoring setup healthconf headnode
[esms1->monitoring->setup[HeadNode]->healthconf]% list
HealthCheck              HealthCheck Param  Check Interval
------------------------ ----------------- ----------------------
DeviceIsUp                                 120
ManagedServicesOk                          120
cmsh                                       1800
esfsmon                  scratch1          120
esfsmon                  scratch2          120
exports                                    1800
failedprejob                               900
failover                                   1800
ldap                                       1800
mounts                                     1800
mysql                                      1800
[hopms->monitoring->setup[MasterNode]->healthconf]% quit
esms:#
```

3. To change the interval of the `esfsmon:`*filesystem*, perform the following commands: (using the `scratch1` file system and setting the interval to 160 seconds in this example).

```
esms# cmsh
[esms1]% monitoring setup healthconf headnode
[esms1->monitoring->setup[HeadNode]->healthconf]% set esfsmon:scratch1 checkinterval 160
[esms1->monitoring->setup*[HeadNode*]->healthconf*]% commit
[esms1->monitoring->setup[HeadNode]->healthconf]% quit
esms#
```

5.7.14 Return a Node to Service

Once a node has been failed over it will no longer be monitored. The failover action moves the failed node to an internal failed node category.

To return a node to service, the administrator must execute the `esfsmon_failback` command with the failed node hostname as the sole argument. The `esfsmon_failback` command performs the following operations.

- Verifies that the node being restored is currently in the failed node category
- Moves the node to the proper active node category
- Calls `lustre_control` to failback the Lustre targets

5.7.15 OSS Failover

A custom monitor script runs periodically on the CIMS node as a Bright health check to verify the operational health of all CLFS nodes. Each OSS has three primary functions, the first two of which are absolutely required for proper CLFS operation. These are to:

- Communicate with other nodes in the LNET InfiniBand fabric
- Mount OSTs and providing file system services for them
- Communicate with a management server via TCP/IP. Loss of IP communication would not necessarily require a failover action, depending on site policies.

When the monitor is operating, hardware or software events that prevent the OSS from performing its primary function trigger an automated failover process that attempts to move the OSTs to a designated failover OSS.

5.7.15.1 Failover Actions

The underlying mechanism supporting failover of OSTs from one OSS to another is the format on each OST that specifies another OSS as a failed node. Custom health checks in Bright verify the basic functions listed above and if a failure is detected or if an OSS fails to respond to the Bright queries, the CIMS monitor script will shutdown power for the failed OSS, which un-mounts its OSTs, and then call `lustre_control` to perform a failover of the OSTs. Note that this typically doubles the OST load on the failover OSS.

5.7.15.2 Corrective Actions

The actions required after an OSS failover are to determine the cause of the failure and correct it. Log entries by Bright and the monitor script should point toward the primary cause.

5.7.15.3 Recovery Actions

Once the OSS has been repaired or patched and is ready for service, its OSTs should be un-mounted from the failover OSS and remounted on their home OSS. This can be done with the `esfsmon_failback` command and typically can be done with the file system up and active.

It is highly recommended that the `esfsmon_failback` command be used to recover the previously failed OSS back into the system. The `esfsmon_failback` script takes the OSS hostname as an argument and performs the necessary OST unmounts and remount operations via the `lustre_control` utility. It also performs all the administrative housekeeping to put the OSS back into the correct node category so it will be monitored again.

5.7.16 MDT/MGS Failover

At a high level, the primary functions of the metadata server (MDS) and metadata targets, which are storage devices (MDT), are the same as those of the OSS: Communicating on two networks (LNET and IP) and mounting and providing file system services for disks formatted for Lustre. The only difference is that the disks are the MDT and the MGS volumes (MGT).

When the esfsmon 2.0.0 monitor is operating, hardware or software event that prevent the MDS from performing its primary function trigger an automated failover process that attempts to move the MDTs to a designated failover MDS.

5.7.16.1 Failover Actions

The designation in format of a *failnode* is as described above for OSS except in this case the MDT is formatted, typically to reference a standby MDS or another active MDS. Failure monitoring with Bright queries and the CIMS monitor script is the same as described above for OSS. An MDS failure also triggers the CIMS to initiate a power reset followed by a call to the standby or designated MDS to mount the MDT(s) of the failed MDS.

5.7.16.2 Corrective Actions

The action required after an MDS failover are to determine the cause of the failure and correct it. Log entries by Bright and the monitor script should point to the primary cause.

5.7.16.3 Recovery Actions

Once the MDS has been repaired or patched and is ready for service, its MDT(s) should be un-mounted from the failover MDS and remounted on their home MDSs. This can be done with the `esfsmon_failback` command and can often be done while the file system is mounted and in use by clients.

It is highly recommended that the `esfsmon_failback` command be used to recover the previously failed MDS back into the system. The `esfsmon_failback` script takes the MDS hostname as an argument and performs the necessary unmount and remount operations via the `lustre_control` utility. It also performs all the administrative housekeeping to put the MDS back into the correct node category so it will be monitored again.

5.7.16.4 MGS Failover

Most configurations feature a merged MDS and MGS server, with separate disks for MGT and MDT. All of the MDS discussion applies equally to MGS failover with the added requirement for different formatting. Specifically the MGT LUN itself does not specify a failnode parameter. Since it is the repository for configuration data of the file system, it does not need to report its new location and is functional by simply being remounted on a failover server. However, the corollary requirement to this one is that every other CLFS node (OST and MDT) in the configuration must be aware of this new location.

This is accomplished by adding a second (i.e. alternate) `mgsnode` parameter to the format of every OST and MDT which points to the MGS/MDS failover node. Finally, MGS failover can take significantly longer than MDS failure alone, or OSS failures, because every client in the configuration must become aware of the new MGS node address. For large configurations this may exceed existing timeout values.

5.8 Configure kdump on CentOS™

kdump is configured on a CLFS system by modifying the configuration files for the software image and Bright category. Dump files from slave nodes are stored either on the CIMS using NFS™, or on the slave node local disk. To save dump files to a local disk on a slave node, create a persistent `/var/crash` partition. Configure kdump for a generic CLFS category (such as `esfs-generic`) before you clone the generic category.

Procedure 71. Configure kdump on CentOS

1. Log in to the CIMS as `root`.
2. To configure kdump on a CLFS node (in this example `esfs-mds001`) clone the slave node software image to a test image. This example clones `ESF-XX-2.2.0-201401151643` to `ESF-XX-2.2.0-kdump`.

```
esms1# cd /cm/images
esms1# cp -pr ESF-XX-2.2.0-201401151643 ESF-XX-2.2.0-kdump
esms1# cmsh
[esms1%] softwareimage
[esms1->softwareimage]% clone ESF-XX-2.2.0-201401151643 ESF-XX-2.2.0-kdump
```

3. Commit your changes.

```
[esms1->->softwareimage*[ESF-XX-2.2.0-kdump*]]% commit
[esms1->->softwareimage[ESF-XX-2.2.0-kdump]]%
```

4. Create a test category, or use the generic `esfs-generic` category to configure kdump.

```
[esms1->->softwareimage[ESF-XX-2.2.0-kdump]]% category
[esms1->category]%
```

5. Use the generic `esfs-generic` category.

```
[esms1->category]% use esfs-generic
[esms1->category*[esfs-generic*]%
```

Or

Clone a production CLFS category (*production_category*) or create a test category (*test_category*) and substitute that category throughout this procedure.

```
[esms1->category]% clone production_category test_category
[esms1->category*[test_category*]%
```

6. Assign the kdump software image (`ESF-XX-2.2.0-kdump`) to the `esfs-generic` category.

```
[esms1->category*[esfs-generic*]% set softwareimage ESF-XX-2.2.0-kdump
```

7. Add `/var/crash` to the exclude lists for the `ESF-XX-2.2.0-kdump` image. The `vim` editor launches and enables the exclude list file to be edited.

Add `- /var/crash/*` to the list of excluded files:

```
[esms1->category*[esfs-generic*]% set excludelistsyncinstall
[esms1->category*[esfs-generic*]% set excludelistupdate
[esms1->category*[esfs-generic*]% set excludelistgrab
[esms1->category*[esfs-generic*]% set excludelistgrabnew
```

8. Save each file and commit your changes.

```
[esms1->category*[esfs-generic*]% commit
[esms1->category[esfs-generic]%
```

9. Assign the `esfs-generic` category to a CLFS node (`esfs-mds001`) and commit your changes.

```
[esms1->category[esfs-generic]]% device use esfs-mds001
[esms1->device[esfs-mds001]]% set category esfs-generic
[esms1->device*[esfs-mds001*]]% commit
[esms1->device[esfs-mds001]]%
```

10. If you are saving kdump crash files to the slave node local disk, add the following lines to the finalize script for the `esfs-generic` category. This command opens the `vim` editor. Scroll down and add the lines before `exit 0`.

```
[esms1->device[esfs-oss1]]% category use esfs-generic
[esms1->category[esfs-generic]]% set finalizescript
DEV=$( awk -- '{ if ($2 == "/localdisk/var/crash") { print $1; exit 0 } }' < /proc/mounts )
[ -n "$DEV" ] && e2label $DEV crash
```

11. Commit your changes.

```
[esms1->category*[esfs-generic*]]%  commit
```

12. Set the storage location for crash dumps. If crash dumps will be saved to the CIMS proceed to step 13. If crash dumps will be saved to the slave node local disk, proceed to step 14.

13. To save crash files to the /var/crash directory on the primary CIMS:

    a. Use fsexports to determine whether the CIMS is exporting /var/crash.

```
[esms1->category[esfs-generic]]% device use esms1
[esms1->device[esms1]]% fsexports
[esms1->device[esms1]->fsexports]% list
Name (key)                                    Path
--------------------------------------------- ------------------------------
/cm/shared@esmaint-net                        /cm/shared
/home@esmaint-net                             /home
/var/spool/burn@esmaint-net                   /var/spool/burn
/cm/node-installer/certificates@esmaint-net   /cm/node-installer/certificates
/cm/node-installer@esmaint-net                /cm/node-installer
```

    b. If /var/crash is not exported from the CIMS, then configure and export it to slave nodes.

```
[esms1->device[esms1]->fsexports]% add /var/crash
[esms1->device*[esms1*]->fsexports*[/var/crash*]]% set name /var/crash@esmaint-net
[esms1->device*[esms1*]->fsexports*[/var/crash*]]% set extraoptions no_subtree_check
[esms1->device*[esms1*]->fsexports*[/var/crash*]]% set hosts esmaint-net
[esms1->device[esms1]*]->fsexports*[/var/crash*]]% set write yes
[esms1->device*[esms1*]->fsexports*[/var/crash*]]% commit
```

    c. Exit /var/crash submode.

```
[esms1->device[esms1]->fsexports[/var/crash]]% exit
[esms1>device[esms1]->fsexports]%
```

    d. Verify that the CIMS is exporting /var/crash.

```
[esms1>device[esms1]->fsexports]% list
Name (key)                                    Path
--------------------------------------------- ------------------------------
/cm/shared@esmaint-net                        /cm/shared
/home@esmaint-net                             /home
/var/spool/burn@esmaint-net                   /var/spool/burn
/cm/node-installer/certificates@esmaint-net   /cm/node-installer/certificates
/cm/node-installer@esmaint-net                /cm/node-installer
/var/crash@esmaint-net                        /var/crash
```

    e. Exit cmsh.

    f. Update the exports.

```
esms1# exportfs -a
```

14. Use the chroot shell to edit the /boot/pxelinux.cfg/default file in kdump test image created in step 2 (ESF-XX-2.2.0-kdump).

    a. Use chroot to edit the /boot/pxelinux.cfg/default file.

```
esms1# chroot /cm/images/ESF-XX-2.2.0-kdump
esms:/>vi /boot/pxelinux.cfg/default
```

b. Scroll down and locate the following line:

```
# End of documentation, configuration follows:
```

c. Enter the following lines in the `default` configuration file:

```
LABEL kdump
KERNEL vmlinuz
IPAPPEND 3
APPEND initrd=initrd crashkernel=512M CMDS console=tty0 console=ttyS1,115200n8 CMDE
MENU LABEL ^KDUMP      - Normal boot mode with kdump
MENU DEFAULT
```

d. Examine the other `LABEL` entries in the `default` configuration file and remove the line: `MENU DEFAULT`.

e. Exit and save the file.

15. Verify that `/var/crash` exists in the `ESF-XX-2.2.0-kdump` image and is a directory. If necessary:

```
esms1:/> mkdir /var/crash
esms1:/> ls -l /var/crash
```

16. Edit the `/etc/kdump.conf` file and add or modify the following lines:

```
esms1:/> vi /etc/kdump.conf
```

a. Add the following lines at the end of the `/etc/kdump.conf` file.

```
path /var/crash
core_collector makedumpfile -c --message-level 1 -d 27
link_delay 60
default reboot
```

If you want to save crash dump files to the local add this line to the `kdump.conf` file. If the file system type is `ext4`, then replace `ext3` with `ext4`.

```
ext3 LABEL=crash
```

Create a persistent partition (`/var/crash`) in the disk setup XML file for the kdump test category (`ESF-XX-2.2.0-kdump`). Creating a separate partition for crash dumps on the slave node software image prevents `/var` from filling up and causing problems for the operating system.

b. Exit and save the file.

17. Enable the `kdump` service.

```
esms1:/> chkconfig kdump on
```

18. Exit the `chroot` shell.

```
esms1:/> exit
esms1#
```

19. Reboot the `esfs-mds001` test node and run kdump.

a. Start a console window on the test slave node (`esfs-mds001`).

```
esms1# cmsh
[esms1]% device; use esfs-mds001
[esms1->device[esfs-mds001]]% rconsole
```

b. Reboot the test node (`esfs-mds001`).

```
esfs-mds001: reboot
esfs-mds001: Reboot in progress ...
```

c. When the node reboots, initiate kdump.

```
esms1#  ssh esfs-mds001
esfs-mds001#echo c > /proc/sysrq-trigger
```

If dumping over NFS to the CIMS, the dump file is created in `/var/crash` on the CIMS node. If dumping to the slave node's local disk, the dump file is created in `/var/crash` on the slave node's local disk.

20. Assign the kdump software image to the `esfs-generic` category. Switch to `category` mode and configure the production CLFS category to use the kdump software image.

```
[esms1->device[esfs-oss1]]% category
[esms1->category]% use esfs-generic
[esms1->category[esfs-generic]]% set softwareimage ESF-XX-2.2.0-kdump
```

21. Reboot all of the nodes in the `esfs-generic` category, so that they use the kdump software image.

```
[esms1->category[esfs-generic]]% device
[esms1->device]% reboot -c esfs-generic
esfs-oss1: Reboot in progress ...
```

22. Exit `cmsh`.

```
[esms1->device]% quit
```

5.9 Configure a NetApp™ Storage System

Thsee instructions apply to both SAS (Serial Attached SCSI) and Fibre Channel (FC) RAIDs and supersede the documentation supplied by the RAID manufacturer.

Use the SANtricity™ storage management software from NetApp, Inc. to manage external NetApp RAID storage devices. SANtricity is provided as a separate package and is installed from a CD on the CIMS. The RAID controllers are set to IP addresses on the `esmaint-net` network in the 10.141.100.xxx range. See Figure 2 and CIMS Network Configuration on page 30.

5.9.1 Install SANtricity Storage Manager Software for NetApp Devices

The SANtricity software is generally preinstalled and the SANtricity media is shipped with the system. However, if the CIMS does not have the software installed, you can install it. The SANtricity SMClient executable is found in `/opt/SMgr/client`.

Procedure 72. Install SANtricity storage management software

1. Log in to the CIMS as `root`.
2. If you are installing from the SANtricity CD, insert it into the CIMS CD drive and mount it.

```
esms1# mount /dev/cdrom /media/cdrom
mount: block device /dev/sr0 is write-protected, mounting read-only
```

   Or, if you are installing from the `SMIA-LINUXX64-10.80.A0.47.bin` file, copy `SMIA-LINUXX64-10.80.A0.47.bin` to `/root/release`.

```
esms1# cp ./SMIA-LINUX-10.70.A0.25.bin /root/release/
```

3. Set the `DISPLAY` environment variable.

```
esms1# export DISPLAY=:0.0
```

4. Verify that the X Window System is functioning by launching `xterm` or executing the `xlogo` utility.

```
esms1# xterm
```

   Exit the `xterm` window.

5. Run the executable file.

   If you are installing from the CD:

```
esms1# /bin/bash /media/cdrom/Linux*x86_64/install/SMIA-LINUXX64-10.80.A0.47.bin
```

   Or, if you are installing from a directory:

```
esms1# /root/release/SMIA-LINUXX64-10.80.A0.47.bin
/root/release/SMIA-LINUXX64-10.80.A0.47.bin
Preparing to install...
Extracting the JRE from the installer archive...
Unpacking the JRE...
Extracting the installation resources from the installer archive...
Configuring the installer for this system's environment...

Launching installer...
```

6. Click **Next**. The **License Agreement** window displays.
7. Accept the license agreement and click **Next**. The **Select Installation Type** window displays.
8. Click **Typical (Full Installation)**, then click **Next**.

   The **Multipathing Driver Warning** window displays.

9. Click **OK**. The **Preinstallation Summary** window displays.
10. Click **Install**.

    The **Installing SANtricity** window displays and shows the installation progress. When the installation completes, an **Install Complete** window appears.
11. Click **Done**. The SANtricity client is installed in `/usr/bin/SMclient` and is currently running.
12. Close the file browser and eject the CD.

```
esms1# eject
```

5.9.2 Configure LUNs for NetApp Devices

Create a Volume Group and the LUNs that are members of it.

Procedure 73. Create a volume group for NetApp devices

You must be logged on to the CIMS as `root`.

1. Start the SANtricity software.

```
esms1# /usr/bin/SMclient
```

   The **SANtricity Storage Manager** window displays.
2. If the **Select Addition Method** window displays, choose one of the following options; otherwise, skip to step 3:

   - **Automatic** — Select this option if you did not assign IP addresses to the storage array controllers using a serial connection. The SANtricity software automatically detects the available controllers, in-band, using the Fibre Channel or InfiniBand link.
   - **Manual** — Select this option if `esmaint-net` IP addresses are assigned to the RAID controllers. Refer to IP address scheme discussed in CIMS Network Configuration on page 30. Figure 2 shows how RAID controllers are connected to the `es-maint` network. The rest of this procedure assumes that you selected the **Manual** option.
3. Double-click the name for the Storage Array that you want to configure. The **Array Management** window displays.
4. Click the **Logical/Physical** tab.
5. Right-click **Unconfigured Capacity** and select **Create Volume**. The **Create Volume** wizard displays.
6. Click **Next** on the **Introduction (Create Volume)** window.
7. Select the **Manual** option on the **Specify Volume Group (Create Volume)** window.

8. Select the tray, and desired slots, and click **Add**.
9. Verify that the RAID level is correct (for example RAID 5).
10. Click **Calculate Capacity**.
11. Click **Next** on the **Specify Volume Group (Create Volume)** window.

    When you create the first Volume Group, you are prompted to create the first volume.

Procedure 74. Create and configure volumes for NetApp devices

1. Enter a new volume capacity.
2. Specify units as GB or MB.
3. Enter a name.
4. Select the **Customize Settings** option.
5. Click **Next** in the **Specify Capacity/Name (Create Volume)** window.
6. Verify the settings on the **Customize Advanced Volume Parameters (Create Volume)** window. These settings are used for the all of the LUNs.
   - For **Volume I/O characteristics type**, verify that **File System** is selected.
   - For **Preferred Controller Ownership**, verify that **Slot A** is selected. This places the LUN on the A Controller.
7. Click **Next** in the **Customize Advanced Volume Parameters (Create Volume)** window.
8. In the **Specify Volume to LUN Mapping** window, select the **Default mapping** option.
9. If not configuring multipath, select **Host type**, and select **Linux™** from the drop-down menu. If you intend to configure multipath, set **Host type** to **Linux (DM-MP)**.
10. Click **Finish** in the **Specify Volume to LUN Mapping** window.
11. When prompted to create more LUNs in the **Creation Successful (Create Volume)** window, select **Yes** unless this is the last volume you are creating. If this is the last volume, select **No** and skip to step 15.
12. In the **Allocate Capacity (Create Volume)** window, verify that **Free Capacity** is selected on **Volume Group 1 (RAID 5)**.
13. Click **Next** in the **Allocate Capacity (Create Volume)** window.
14. Repeat step 1 through step 13 to create all of the volumes.
15. Click **OK** in the **Completed (Create Volume)** window.

16. Create a hot spare. The hot spare provides a ready backup if any of the drives in the Volume Group fail.
    a. Right-click on a drive in the right portion of the window and select **Hot Spare Coverage**.
    b. Select the **Manually Assign Individual Drives** option.
    c. Click **OK**.
    d. Click **Close**.
17. Exit the tool.

5.9.3 Multipath Host Mappings

If you are configuring multipath, and are experiencing errors from NetApp™ storage devices such as `Volume not on preferred path due to AVT/RDAC failover`, be sure to set the host mappings parameter to **Linux (DM-MP)** or **Linux** (on older firmware). Use the SANtricity™ Storage Manager Software command **Host Mappings->Default Group->Change Default Host Operating System** to configure the host mappings setting.

5.9.4 Configure Remote Logging of NetApp™ Storage System Messages

NetApp storage systems use Simple Network Management Protocol (SNMP) to provide boot RAID messages on the `esmaint-net` network. Configure the community settings for the RAID device using the serial console connection (a custom cable is shipped with NetApp devices for this purpose). Make the console connection in a similar manner to how you configure a switch. (See Add a Managed Switch or Device to the Bright Configuration on page 126.) Refer to your NetApp storage system documentation for more information.

5.9.5 Add a NetApp RAID Storage System to Bright

RAID devices are added as type `genericdevice` in `cmsh` or as **Other** devices using the `cmgui`. The procedure below shows how to add a RAID device to Bright using `cmsh`.

Procedure 75. Add a NetApp RAID storage system to Bright

1. Install the SANtricity (`SMclient`) software on the CIMS. Refer to Install SANtricity Storage Manager Software for NetApp Devices on page 222.
2. Set the RAID controller IP addresses to valid `esmaint-net` addresses (in the 10.141.100.xxx range). Refer to IP address scheme discussed in CIMS Network Configuration on page 30. Figure 2 shows how RAID controllers are connected to the `esmaint-net` network.

3. Log in to the CIMS as `root` and run `cmsh`.

```
esms1# cmsh
[esms1]%
```

4. Switch to `device` mode.

```
[esms1]% device
```

5. Add the NetApp RAID controller(s) under **Other** devices in the `cmgui` resource tree, or as a `genericdevice` in `cmsh`. This procedure adds a NetApp 3992 controller A for example.

```
[esms1->network[storage-net]]% device
[esms1->device]% add genericdevice netapp3992-cntrlA
```

6. Set the device information for model, rack number, device height in rack units, and device U position in the rack.

```
[esms1->device*[netapp3992-cntrlA*]]% set model LSI3992
[esms1->device*[netapp3992-cntrlA*]]% set rack rack_num
[esms1->device*[netapp3992-cntrlA*]]% set deviceheight rack_units
[esms1->device*[netapp3992-cntrlA*]]% set deviceposition rack_position
```

7. Set the RAID controller Ethernet port IP address on the `esmaint-net`, for example 10.141.100.10.

```
[esms1->device*[netapp3992-cntrlA*]]% set ip controller_ip_address
```

8. Set MAC address for the RAID controller Ethernet port.

```
[esms1->device*[netapp3992-cntrlA*]]% set mac MAC_address
```

9. Set the network to `esmaint-net`.

10. Set power control to `custom` (power is controlled from the SANtricity client software).

```
[esms1->device*[netapp3992-cntrlA*]]% set powercontrol custom
```

11. (Optional) Add notes for this controller. The following command opens the `vi` editor where notes can be added for this device. For example: `oss001 connection to RAID controller A, Even LUNs`.

```
 [esm1->device*[netapp3992-cntrlA*]]% set notes
```

12. Show the controller settings.

```
 [esm1->device*[netapp3992-cntrlA*]]% show
Parameter                     Value
----------------------------- -------------------------------------
Activation                    Tue, 14 May 2013 08:20:30 CDT
Additional Hostnames
Container index               0
Custom ping script
Custom ping script argument
Custom power script
Custom power script argument
Device height                 4
Device position               30
Ethernet switch
Hostname                      netapp3992-cntrlA
Ip                            10.141.100.10
Mac                           00:0B:5F:CE:2F:40
Model
Network                       esmaint-net
Notes                         <47 bytes>
Partition                     base
Power control                 custom
PowerDistributionUnits
Rack                          1
Revision
Tag                           00000000a000
Type                          GenericDevice
Userdefined1
Userdefined2
```

5.10 Rediscover New LUNs

This procedure causes the CLFS to rediscover the new LUNs created in Procedure 74 on page 224.

Procedure 76. Rebooting the CLFS and verifying LUNs are recognized

1. Log in to the CIMS as the `root`.
2. SSH to the CLFS nodes and enter the following command to verify that the LUNs are recognized:

```
esms1# ssh esfs-oss001
```

3. Reboot the node or probe the SCSI bus to verify the LUNs are available to the MDS or OSS node. Two paths to LUN 5 (`/dev/sdg` and `/dev/sdm`) indicate multipath is enabled.

```
[root@esfs-oss001 ~]# lsscsi
[0:2:0:0]    disk    DELL     PERC H710P       3.13  /dev/sda
[5:0:0:0]    cd/dvd  PLDS     DVD+-RW DS-8A8SH KD51  /dev/sr0
[7:0:0:0]    disk    LSI      INF-01-00        0780  /dev/sdb
[7:0:0:1]    disk    LSI      INF-01-00        0780  /dev/sdc
[7:0:0:2]    disk    LSI      INF-01-00        0780  /dev/sdd
[7:0:0:3]    disk    LSI      INF-01-00        0780  /dev/sde
[7:0:0:4]    disk    LSI      INF-01-00        0780  /dev/sdf
[7:0:0:5]    disk    LSI      INF-01-00        0780  /dev/sdg
[8:0:0:0]    disk    LSI      INF-01-00        0780  /dev/sdh
[8:0:0:1]    disk    LSI      INF-01-00        0780  /dev/sdi
[8:0:0:2]    disk    LSI      INF-01-00        0780  /dev/sdj
[8:0:0:3]    disk    LSI      INF-01-00        0780  /dev/sdk
[8:0:0:4]    disk    LSI      INF-01-00        0780  /dev/sdl
[8:0:0:5]    disk    LSI      INF-01-00        0780  /dev/sdm
```

4. List the disk devices by using the `fdisk` command to verify that the LUNs (volumes) are configured.

```
[root@esfs-oss001 ~]# fdisk -l
```

5.11 Partition the LUNs

After you finish creating, formatting, and zoning the LUNs on the RAID, you must partition them. Refer to the NetApp documentation for the SANtricity Storage Manager software for more information.

5.12 Clone the Generic `esfs-generic` Category to `esfsmon` 2.0.0 Categories

After the `esfs-generic` category is fully configured to support generic CLFS nodes, then clone this category to create `esfs-odd-`*filesystem*, `esfs-even-`*filesystem*, and `esfs-failed-`*filesystem* categories for `esfsmon` 2.0.0 and assign odd and even CLFS nodes to the appropriate category.

Procedure 77. Clone the `esfs-generic` category to odd, even, and failed categories

1. Log in to the CIMS as `root`.
2. Start `cmsh` and switch to `category` mode.

```
esms1# cmsh
[esms1]% category
```

3. Clone the `esfs-generic` category to even, odd, and failed categories required for `esfsmon` 2.0.0.

```
[esms1->category]% clone esfs-generic esfs-even-filesystem
[esms1->category*[esfs-even-filesystem*]]% clone esFS-MDS esfs-odd-filesystem
[esms1->category*[esfs-odd-filesystem*]]% clone esFS-MDS esfs-failed-filesystem
[esms1->category*[esfs-failed-filesystem*]]% commit
[esms1->category*[esfs-failed-filesystem]]% list

Name (key)               Software image
------------------------ ------------------------
default                  default-image
esFS-MDS                 ESF-XX-2.2.0-201401151643+
esfs-generic             ESF-XX-2.2.0-201401151643+
esFS-OSS                 ESF-XX-2.2.0-201401151643+
esLogin-XC               ESL-XC-2.2.0-201401160637
esfs-even-filesystem      ESF-XX-2.2.0-201401151643+
esfs-failed-filesystem    ESF-XX-2.2.0-201401151643+
esfs-odd-filesystem       ESF-XX-2.2.0-201401151643+
[esms1->category*[esfs-failed-filesystem]]%
```

4. Exit `cmsh`.

```
[esms1->category*[esfs-failed-filesystem]]% quit
```

5.13 Add Nodes to CLFS Categories

The CLFS node categories must be named specifically for `esfsmon` 2.0.0 released with ESM XX-3.0.0. CLFS nodes must be assigned to even or odd categories based on their node name (`$HOSTNAME`). When the `esfs-generic` category is fully configured, clone it to create the `esfs-odd-`*filesystem*, `esfs-even-`*filesystem*, and `esfs-failed-`*filesystem* categories.

Procedure 78. Add nodes to CLFS `esfsmon` 2.0.0 categories

1. Log in to the CIMS as `root` and run `cmsh`.

```
esms1# cmsh
[esms1]%
```

2. Switch to `device` mode:

```
[esms1]% device
```

3. Add a node to the `esfs-generic` category. This procedure uses the example hostname `esfs-mds001`.

```
[esms1->device]% use esfs-mds001
[esms1->device[esfs-mds001]]% set category esfs-generic
[esms1->device[esfs-mds001*]]%
```

4. Set the rack information (device height, position in the rack, and rack number).

This example assumes that the device height is 2U, its position in the rack is `12`, and the rack number is `1`. To display a list of all the possible settings, enter a question mark (`?`) and press `Enter`, or enter `set` and press the `Tab` key.

```
[esms1->device[esfs-mds001*]]% set deviceheight 2
[esms1->device[esfs-mds001*]]% set deviceposition 12
[esms1->device[esfs-mds001*]]% set rack 1
```

5. Commit your device changes.

```
[esms1->device[esfs-mds001*]]% commit
[esms1->device[esfs-mds001]]%
```

6. Repeat step 3 through step 5 to add each node to the `esfs-odd-`*filesystem*, `esfs-even-`*filesystem*, and `esfs-failed-`*filesystem* categories.

7. Check the CLFS categories to verify they are configured properly.

```
[esms1->device[esfs-mds001]]% category
[esms1->category]]% list
Name (key)                Software image
------------------------  ------------------------
default                   default-image
default-diskless          default-image
esfs-generic              ESF-XX-2.2.0-201401151643
esFS-MDS                  ESF-XX-2.2.0-201401151643
esFS-OSS                  ESF-XX-2.2.0-201401151643
esfs-even-filesystem       softwareimage
esfs-failed-filesystem     softwareimage
esfs-odd-filesystem        softwareimage
esLogin-XC                ESL-XC-1.0.2-2013022113+
esLogin-XE                ESL-XE-1.1.1_CLE4.1


[esms1->category]% usedby esfs-even-filesystem
Category used by the following:
Type             Name                      Parameter                 Autochange
---------------- ------------------------- ------------------------- ------------
Device           esfs-mds002               category                  no
Device           esfs-mds004               category                  no
Device           esfs-oss002               category                  no
Device           esfs-oss004               category                  no

[esms1->category]% usedby esfs-odd-filesystem
Category used by the following:
Type             Name                      Parameter                 Autochange
---------------- ------------------------- ------------------------- ------------
Device           esfs-mds001               category                  no
Device           esfs-mds003               category                  no
Device           esfs-oss001               category                  no
Device           esfs-oss003               category                  no
```

8. Exit `cmsh`.

```
[esms1->category]% quit
esms1#
```

5.14 Create a CLFS Node Group

Node groups are used to make operating on several nodes simple and efficient. Nodes may belong to several groups at the same time. There are no parameters associated with a node group other than the member nodes.

Cray recommends creating a node group for large configurations.

Procedure 79. Create a CLFS node group

1. Log into the CIMS node as `root`.
2. Run the `cmsh` command.

```
esms1# cmsh
[esms1]%
```

3. Switch to `nodegroup` mode:

```
[esms1]% nodegroup
[esms1->nodegroup]%
```

4. Use the `add` command to add a node group. This example creates a new node group called `mds_nodes`.

```
[esms1->nodegroup]% add mds_nodes
[esms1->nodegroup*[mds_nodes*]]%
```

5. Use the `append` command to add nodes to the group. Multiple nodes can be added as a list (*N*) or a range (`esfs-mds001..esfs-mds`*N*).

```
[esms1->nodegroup*[mds_nodes*]]% append nodes esfs-mds001..esfs-mds002
```

6. Commit your changes.

```
[esms1->nodegroup*[mds_nodes*]]% commit
```

7. List node groups.

```
[esms1->nodegroup[mds_nodes]]% list
Name (key)               Nodes
------------------------ ----------------------
mds-nodes                esfs-mds001,esfs-mds002
eslogin-all              eslogin1, eslogin2
oss-all                  esfs-oss001,esfs-oss001,esfs-oss003,esfs-oss004
```

8. Repeat this procedure to create an `oss_nodes` group, and append each OSS node to the `oss_nodes` group.
9. Exit `cmsh`.

```
[esms1->nodegroup[mds_nodes]]% quit
esms1#
```

5.15 Configure a Generic Category for CLFS Nodes (`esfs-generic`)

Bright categories are used to associate a specific software image, finalize script, and other configuration information to a specific node type. The `default` category is assigned to the first CLFS node when it was cloned from the default node (`node001`). The ESF installation software creates a Bright category named `esFS-MDS` by default (the `esFS-OSS` category is no longer used) to build an ESF software image. This `esFS-MDS` category must be configured as a generic category for CLFS nodes (`esfs-generic`), then cloned into the required odd and even `esfsmon` generic to support `esfsmon` 2.0.0. The generic `esfs-generic` category must define the network settings, default disk partitions, exclude lists, finalize script, support kdump, and include other customizations for your site configuration before it is cloned to the odd and even categories required by `esfsmon` 2.0.0.

`esfsmon` 2.0.0 requires node categories in Bright be designated as odd, even, and failed. After the `esfs-generic` category is fully configured, clone the category to `esfs-odd-`*filesystem*, `esfs-even-`*filesystem*, and `esfs-failed-`*filesystem* categories, then assign CLFS nodes to either the even or odd categories.

The ESM XX-3.0.0 release provides a single finalize script, `site.esf_finalize.sh`, that configures both MDS and OSS nodes. The node names (`$HOSTNAME`) **must** contain the string `mds` or `oss` string so that the `site.esf_finalize.sh` script can configure both node types. Also, CLFS node names must be numbered so that they can be placed in an odd or even category.

Procedure 80. Configuring the Bright `esfs-generic` category

This procedure clones the default `esFS-MDS` category to `esfs-generic`, then customizes the `esfs-generic` category network settings, default disk partitions, exclude lists, finalize script, kdump, etc. When the `esfs-generic` category configuration is complete, clone that category to make the `esfs-odd-`*filesystem*, `esfs-even-`*filesystem*, and `efs-failed-`*filesystem* categories, then assign CLFS nodes to either the even or odd categories.

1. Log in to the CIMS node as `root` and run `cmsh`.

```
esms1# cmsh
[esms1]%
```

2. Switch to `category` mode and list categories and associated software images.

```
[esms1]% category
[esms1->category]% list
Name (key)               Software image
------------------------ ----------------------
default                  default-image
default-diskless         default-image
esFS-MDS                 default-image
esFS-OSS                 default-image
esLogin-XC               ESL-XC-1.0.2-2013022113+
esLogin-XE               ESL-XE-1.1.1_CLE4.1
```

3. Clone the `esFS-MDS` category to `esfs-generic` and commit the change.

```
[esms1->category]% clone esFS-MDS esfs-generic
[esms1->category*]% commit
```

4. Assign the CLFS image (`ESF-XX-2.2.0-201401151643`) created by `ESFinstall`, to the `esfs-generic` category.

```
[esms1->category]% use esfs-generic
[esms1->category[esfs-generic]]% set softwareimage ESF-XX-2.2.0-201401151643
[esms1->category*[esfs-generic*]]%
```

5. If required, set the default gateway so that MDS nodes can contact an external LDAP server on `site-user-net`. For *gateway*, use the IP address of your site's gateway (on `site-user-net`).

```
[esms1->category*[esfs-generic*]]% set defaultgateway gateway
```

6. Commit the changes.

```
[esms1->category*[esfs-generic*]]% commit
[[esms1->category[esfs-generic]]%
```

7. Configure the `esfs-generic` category with the proper disk partition sizes and configuration.

   A custom disk partition layout can be applied using an XML schema to a category of nodes. A disk partition layout that is applied to an individual node within a category overrides the category setting. Add `<blockdev>` XML entries to the `site.esfs-diskfull.xml` file if there are several LUNs on a SAS RAID that are used for MDTs. The XML file currently defines `/dev/sda` through `/dev/sdz`.

   Changes made to the `esfs-diskfull.xml` file do not occur until they are saved in `cmgui`, or committed in `cmsh`, and the node is rebooted. Bright detects that the partitioning has changed and invokes a `FULL` install. (Bright also ignores the `NOSYNC` install mode and will repartition the drives on other nodes in the category as well.)

   Older model CLFS nodes can use `esfs-small-diskfull.xml` for disk partitioning to accommodate smaller hard drives.

Important: If the default disk setup XML files are updated in a ESM release or if the site disk setup XML files have been customized, system administrators must compare the newly released disk setup XML files with the current production disk setup XML files, and merge the changes manually. After the changes have been merged, you must load the updated disk setup file into the Bright database for the node category and reboot all the nodes that use that category.

a. Quit `cmsh` and copy the default XML configuration to an `etc` directory to prevent it from being overwritten during software updates.

```
[esms1->category[esfs-generic]]% quit
esms1# cd /opt/cray/esms/cray-es-diskpartitions-XX
esms1# mkdir -p etc
esms1# cp -p default/esfs-diskfull.xml etc/site.esfs-diskfull.xml
```

b. Edit the `site.esfs-diskfull.xml` file to change partition sizes or configuration.

```
esms1# vi etc/site.esfs-diskfull.xml
```

c. Save the changes and return to `cmsh`.

8. Set the disk setup parameter for the `esfs-generic` category and commit the changes.

```
esms1# cmsh
[esms1]% category use esfs-generic
[esms1->category[esfs-generic]]% set disksetup /opt/cray/esms/cray-es-diskpartitions-XX\
/etc/site.esfs-diskfull.xml
[esms1->category*[esfs-generic*]]% commit
```

9. Confirm your changes to the `site.esfs-diskfull.xml` script.

```
[esms1->category[esfs-generic]]% get disksetup
<?xml version="1.0" encoding="UTF-8"?>

<diskSetup xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
   <device>
      <blockdev>/dev/sdablockdev>/dev/sda></blockdev>
      <blockdev>/dev/sdablockdev>/dev/sdb></blockdev>
      <blockdev>/dev/sdablockdev>/dev/sdc></blockdev>
 . . .
```

5.16 Configure the Node Finalize Script for a CLFS Category

The node finalize script is configured for the default `esFS-MDS` category during ESF software installation. The default `esFS-MDS` category is cloned to a generic CLFS category `esfs-generic` that is further customized and cloned into the required `esfsmon` 2.0.0. categories.

The finalize script runs before `init` and is used to set a file configuration or to initialize special hardware, sometimes after a hardware check. It is run in order to make software or hardware work before, or during the later `init` stage of boot. Use a finalize script to execute commands before `init`, when the commands cannot be stored persistently anywhere else, or when it is needed because a choice between (otherwise non-persistent) configuration files must be made based on the hardware before `init` starts.

A single node finalize script, `site.esf_finalize.sh`, is configured to support both MDS or OSS nodes beginning with release ESM XX-3.0.0. The `site.esf_finalize.sh` script differentiates between MDS and OSS nodes by checking the node name (`$HOSTNAME`), which must include either the string `mds` or `oss`.

Important: Files created or modified by a finalize script must be listed in the `excludelistupdate` exclude list for the category. Software updates will overwrite customized files if the files are not specified in an exclude list for the category. Customized files must also be specified in the `excludelistgrab`, and `excludelistgrabnew` exclude lists to prevent customized files from being copied to the CIMS node.

Procedure 81. Configure the node finalize script for CLFS nodes

1. Log in to the CIMS node as `root`.
2. Make a copy of the default finalize script to an `etc` directory to prevent it from being overwritten during software updates.

```
esms1# cd /opt/cray/esms/cray-es-finalize-scripts-XX
esms1# mkdir -p etc
esms1# cp -p default/esf_finalize.sh etc/site.esf_finalize.sh
```

3. Edit the `etc/site.esf_finalize.sh` script.

```
esms1# vi etc/site.efs_finalize.sh
```

4. **MDS configuration —** Edit the IB subnet manager section. Cray recommends that the subnet manager for the IB fabric should be run on `ib0` of the MDS node(s) for system configurations that have many CLFS file system nodes connected to an IB switch. Uncomment the following command line if you want to start the IB subnet manager on `ib0` for MDS nodes.

```
# echo "/usr/sbin/opensm --daemon -g \`/usr/sbin/ibstat mlx4_0 1 | grep GUID | awk '{print \$3}'\`" >>\
 /localdisk/etc/rc.d/rc.local
```

5. Uncomment the following lines if the MDS node must be configured for LDAP.

```
# chroot /localdisk chkconfig nslcd on
# chroot /localdisk chkconfig nscd on
```

6. If you want to set up multipath name mappings to configure multipath for MDS nodes, uncomment the commands in the next section. Enter the actual

WWID's and aliases here. This was configured in previous releases in the `/etc/multipath/bindings` file. Configuring multipath name mappings in a `site.esf_finalize.sh` finalize script is the best practice.

```
#cat << EOMP >> /localdisk/etc/multipath.conf
#multipaths {
#   multipath {
#      wwid  360080000000000000000000000000000
#      alias mgt
#   }
#   multipath {
#      wwid  360080000000000000000000000000000
#      alias mdt
#   }
#}
#EOMP
```

7. **OSS configuration —** Uncomment the following lines to disable IB cards not present.

```
#sed -i -e "s/MTHCA_LOAD=yes/MTHCA_LOAD=no/" /localdisk/etc/infiniband/openib.conf
#sed -i -e "s/QIB_LOAD=yes/QIB_LOAD=no/" /localdisk/etc/infiniband/openib.conf
#sed -i -e "s/MLX4_EN_LOAD=yes/MLX4_EN_LOAD=no/" /localdisk/etc/infiniband/openib.conf
#sed -i -e "s/CXGB3_LOAD=yes/CXGB3_LOAD=no/" /localdisk/etc/infiniband/openib.conf
#sed -i -e "s/NES_LOAD=yes/NES_LOAD=no/" /localdisk/etc/infiniband/openib.conf
```

8. Uncomment the following lines to configure SRP for IB connected RAID.

```
#sed -i -e "s/SDP_LOAD=yes/SDP_LOAD=no/" /localdisk/etc/infiniband/openib.conf
#sed -i -e "s/SRP_LOAD=no/SRP_LOAD=yes/" /localdisk/etc/infiniband/openib.conf
#sed -i -e "s/SRP_DAEMON_ENABLE=no/SRP_DAEMON_ENABLE=yes/" /localdisk/etc/infiniband/openib.conf
```

9. Uncomment one of the following lines to start an IB subnet manager on `ib2` for OSS nodes. Cray recommends that `ib0` on OSS nodes be connected to the IB switch, that `ib1` should not be used, and that `ib2` and `ib3` connect to the storage array controllers.

   a. If one port from the IB card (`mlx4_1` in this example) is connected to storage, uncomment the following line to start the IB subnet manager on `ib2` (the IB interface connected to the storage array). Change the `2` to designate the IB interface port connected to the storage array.

```
#echo "/usr/sbin/opensm --daemon -g \`/usr/sbin/ibstat mlx4_1 2 | grep GUID | awk '{print \$3}'\`" >> \
/localdisk/etc/rc.d/rc.local
```

   b. If both ports of the IB card are connected to storage, each line will activate one port. Uncomment one or both lines according to your site's configuration.

```
#echo "/usr/sbin/opensm --daemon -g \`/usr/sbin/ibstat mlx4_1 1 | grep GUID | awk '{print \$3}'\`" >> \
 /localdisk/etc/rc.d/rc.local
#echo "/usr/sbin/opensm --daemon -g \`/usr/sbin/ibstat mlx4_1 2 | grep GUID | awk '{print \$3}'\`" >> \
/localdisk/etc/rc.d/rc.local
```

10. Uncomment the following lines to increase IB write performance.

```
#echo "### esFS max_sect setting" >>  /localdisk/etc/srp_daemon.conf
#echo "a       max_sect=65535" >>  /localdisk/etc/srp_daemon.conf
```

11. If you want to set up multipath name mappings to configure multipath for

MDS nodes, uncomment the commands in the next section. Enter the actual WWID's and aliases here. This was configured in previous releases in the `/etc/multipath/bindings` file. Configuring multipath name mappings in a `site.esf_finalize.sh` finalize script is the best practice.

```
#cat << EOMP >> /localdisk/etc/multipath.conf
#
#multipaths {
#   multipath {
#      wwid  360080000000000000000000000000000
#      alias ost0
#   }
#   multipath {
#      wwid  360080000000000000000000000000000
#      alias ost1
#   }
#   multipath {
#      wwid  360080000000000000000000000000000
#      alias ost2
#   }
#   multipath {
#      wwid  360080000000000000000000000000000
#      alias ost3
#   }
#}
#
#EOMP
```

12. Run the `cmsh` and set the finalize script for the `esfs-generic` category.

```
esms1# cmsh
[esms1]% category
[esms1->category]% use esfs-generic
[esms1->category[esfs-generic]]% set finalizescript\
/opt/cray/esms/cray-es-finalize-scripts-XX/etc/site.esf_finalize.sh
[esms1->category*[esfs-generic*]]% commit
[esms1->category[esfs-generic]]%
```

13. Verify the `site.esf_finalize.sh` finalize script is correct.

```
[esms1->category[esfs-generic]]% get finalizescript
[esms1->category[esfs-generic]]% quit
```

5.17 Configure LDAP on MDS Nodes

LDAP is configured in the category finalize script for the MDS node. The `named`, `nslcd`, `nscd`, and `ldap` services are turned off by default. To enable these services, uncomment the corresponding lines in the finalize script.

Important: Files created or modified by a finalize script must be listed in the `excludelistupdate` exclude list for the category. Software updates will overwrite customized files if the files are not specified in an exclude list for the category. Customized files must also be specified in the `excludelistgrab`, and `excludelistgrabnew` exclude lists to prevent customized files from being copied to the CIMS node.

Procedure 82. Configure LDAP on MDS nodes

1. Log in to the CIMS node as `root`.
2. Edit the `site.esf_finalize.sh` script.

```
esms1# cd /opt/cray/esms/cray-es-finalize-scripts-XX/etc
esms1# vi site.esf_finalize.sh
```

3. Enable the name service caching daemon (`nscd`) and the LDAP name service daemon `nslcd`. `named`, `nslcd`, `nscd`, and `ldap` are turned off in the default ESF image.

   Add or uncomment the following commands in the `site.esf_finalize.sh` script:

```
chkconfig nscd on
service nscd start
chkconfig nslcd on
service nslcd start
```

4. Save `site.esf_finalize.sh` file and exit the editor.
5. Use the `chroot` shell to edit the ESF software image (in this example, the software image is named `ESF-XX-2.2.0-201401151643`):

```
esms1# cd /cm/images
esms1# chroot ESF-XX-2.2.0-201401151643
[root@esms1 /]#
```

6. Verify the LDAP service (`ldap`), is turned off in the ESF software image. The MDS should not act as an LDAP server. Use the following command to disable the LDAP server instance. The LDAP client configuration settings are typically in `/etc/passwd` and `/etc/nsswitch.conf`, and in the pluggable authentication modules (PAM).

```
[root@esms1 /]# chkconfig ldap off
```

7. Edit `/etc/openldap/ldap.conf` file, and uncomment (enable) the `TIMELIMIT 15` line:

```
[root@esms1 /]# cd /etc/openldap
[root@esms1 /]# cp ldap.conf ldap.conf.orig
[root@esms1 /]# vi ldap.conf
TIMELIMIT     15
```

8. Add/edit the following lines and enter the specific IP and base value settings for your site's information structure:

```
URI ldap://aaa.bbb.ccc.ddd/
BASE dc=somedomain,dc=somedomain,dc=com
base ou=people,dc=somedomain,dc=somedomain,dc=com
```

9. Edit `/etc/nslcd.conf` file:

```
[root@esms1 /]# cd /etc
[root@esms1 /]# cp nslcd.conf nslcd.conf.orig
[root@esms1 /]# vi nslcd.conf
```

10. Add/edit the following lines and enter the specific IP and base value settings for your site's information structure:

```
uri ldap://aaa.bbb.ccc.ddd/
base dc=somedomain,dc=somedomain,dc=com
base ou=people,dc=somedomain,dc=somedomain,dc=com
```

11. Exit the `chroot` shell.

12. Reboot all slave nodes using the ESF-XX-2.2.0-201401151643 software image.

5.18 Configure Device Mapper Multipath on CLFS Nodes

You must configure the device mapper (DM) multipath feature if there are multiple paths to devices from CLFS nodes. You do not need to perform this procedure if the CLFS node is not cabled for DM multipath. Use `/dev/mapper` device names instead of `/dev/disk/by-id` for any device that has multiple paths defined and is controlled by DM multipath.

All CLFS nodes configured for DM multipath have a connection to each storage array controller. Specific WWID values must be added to the `multipaths {}` section of the `/etc/multipath.conf` file so that devices have the same name each time the node boots.

Procedure 83. Configure DM multipath on CLFS nodes

1. After `ESFinstall` has completed, the `multipathd` service must be enabled for the ESF software image.

```
esms1# chroot /cm/images/softwareimage
[root@esms1/]# chkconfig multipathd on
```

2. Copy `multipath.conf.cray` to `/etc` in the ESF software image.

```
[root@esms1/]# cp -p /opt/cray/esfs/cray-esf-multipath-XX/default/etc/multipath.conf.cray\
 /etc/multipath.conf
```

3. Exit the `chroot` shell.

```
[root@esms1/]# exit
esms1#
```

4. Identify the disk names for DM multipath on the MDS nodes. Boot the MDS 1 node and use SSH to login.

```
esms1# ssh mds001
Last login: Tue Jun 11 15:42:52 2013 from esms1.cm.cluster
[root@mds001 ~]#
```

5. Use `multipath -ll` command to display the disk devices.

```
mds001# multipath -ll
mpatha (360080e50002f82ca000002ca5102bcc4) dm-0 LSI,INF-01-00
size=2.2T features='2 pg_init_retries 50' hwhandler='1 rdac' wp=rw
|-+- policy='round-robin 0' prio=6 status=active
| `- 0:0:0:0 sda 8:0  active ready running
`-+- policy='round-robin 0' prio=1 status=enabled
  `- 0:0:1:0 sdb 8:16 active ghost running
```

6. In step 5, the WWID of `360080e50002f82ca000002ca5102bcc4` is the device labeled as `mdt0` instead of `mpatha`.

7. On some systems, the disk device may need to be modified using the `parted` command to remove unneeded partitions. Identify which disk device (`/dev/sda` form) corresponds to this WWID.

```
mds001# ls -l /dev/disk/by-id | grep scsi-360080e50002f82ca000002ca5102bcc4
lrwxrwxrwx 1 root root  9 Jun 10 09:28 scsi-360080e50002f82ca000002ca5102bcc4 -> ../../sda
[root@mds001 ~]# parted /dev/sda
GNU Parted 2.1
Using /dev/sda
Welcome to GNU Parted! Type 'help' to view a list of commands.

(parted) print
Model: LSI INF-01-00 (scsi)
Disk /dev/sda: 2398GB
Sector size (logical/physical): 512B/512B
Partition Table: gpt

Number  Start   End     Size    File system     Name       Flags
 1      17.4kB  20.5GB  20.5GB  ext3            /
 2      20.5GB  22.5GB  2048MB  ext3            /var
 3      22.5GB  24.6GB  2048MB  ext3            /tmp
 4      24.6GB  41.0GB  16.4GB  linux-swap(v1)  /dev/sda4
 5      41.0GB  2398GB  2357GB  ext3            /local
```

8. Remove all partitions so that Lustre® can use the entire device. This example pauses after removing partitions 2 through 5 to show that only one partition is left.

```
(parted) rm 5
(parted) rm 4
(parted) rm 3
(parted) rm 2
(parted) p
Model: LSI INF-01-00 (scsi)
Disk /dev/sda: 2398GB
Sector size (logical/physical): 512B/512B
Partition Table: gpt

Number  Start   End     Size    File system  Name  Flags
 1      17.4kB  20.5GB  20.5GB  ext3         /

(parted) rm 1
(parted) p
Error: /dev/sda: unrecognised disk label
(parted) quit
Information: You may need to update /etc/fstab.
```

9. Exit the MDS node and SSH to the other MDS node. Repeat step 4 through step 8 to identify the disk names for multipath.

10. Identify the disk names for multipath on the OSS nodes. Boot each OSS node and use SSH to login.

```
esms1# ssh esfs-oss001
Last login: Tue Jun 11 15:45:32 2013 from esms1.cm.cluster
[root@oss001 ~]#
```

11. Display the disk devices with the `multipath` command.

```
[root@esfs-oss001 ~]# multipath -ll
mpatha (360080e50001f8c64000000b4513098b2) dm-2 LSI,INF-01-00
size=7.3T features='3 queue_if_no_path pg_init_retries 50' hwhandler='1 rdac' wp=rw
|-+- policy='round-robin 0' prio=6 status=active
| `- 7:0:0:5 sdg 8:96  active ready running
`-+- policy='round-robin 0' prio=1 status=enabled
  `- 8:0:0:5 sdm 8:192 active ghost running
mpathb (360080e50001f87c8000000b951309893) dm-3 LSI,INF-01-00
size=7.3T features='3 queue_if_no_path pg_init_retries 50' hwhandler='1 rdac' wp=rw
|-+- policy='round-robin 0' prio=6 status=active
| `- 8:0:0:4 sdl 8:176 active ready running
`-+- policy='round-robin 0' prio=1 status=enabled
  `- 7:0:0:4 sdf 8:80  active ghost running
mpathc (360080e50001f8c64000000ae51309849) dm-1 LSI,INF-01-00
size=7.3T features='3 queue_if_no_path pg_init_retries 50' hwhandler='1 rdac' wp=rw
|-+- policy='round-robin 0' prio=6 status=active
| `- 7:0:0:3 sde 8:64  active ready running
`-+- policy='round-robin 0' prio=1 status=enabled
  `- 8:0:0:3 sdk 8:160 active ghost running
.
.
.
```

12. On some systems, the disk devices may need to be modified with the `parted` command to remove unneeded partitions. Identify which disk device (`/dev/sda` form) relates to this WWID.

```
[root@esfs-oss001 ~]# ls -l /dev/disk/by-id | grep scsi-360080e50001f8c64000000b4513098b2
lrwxrwxrwx 1 root root  9 Jun  3 14:40 scsi-360080e50001f8c64000000b4513098b2 -> ../../sdm

[root@esfs-oss001 ~]# parted /dev/sdm
GNU Parted 2.1
Using /dev/sda
Welcome to GNU Parted! Type 'help' to view a list of commands.

(parted) print
Model: LSI INF-01-00 (scsi)
Disk /dev/sdm: 2398GB
Sector size (logical/physical): 512B/512B
Partition Table: gpt

Number  Start   End     Size    File system     Name       Flags
 1      17.4kB  20.5GB  20.5GB  ext3            /
 2      20.5GB  22.5GB  2048MB  ext3            /var
 3      22.5GB  24.6GB  2048MB  ext3            /tmp
 4      24.6GB  41.0GB  16.4GB  linux-swap(v1)  /dev/sda4
 5      41.0GB  2398GB  2357GB  ext3            /local
```

13. Remove all partitions so that Lustre can use the entire device. This example pauses after removing partitions 2 through 5 to show that only one partition is left.

```
(parted) rm 5
(parted) rm 4
(parted) rm 3
(parted) rm 2
(parted) p
Model: LSI INF-01-00 (scsi)
Disk /dev/sdm: 2398GB
Sector size (logical/physical): 512B/512B
Partition Table: gpt

Number  Start   End     Size    File system  Name  Flags
 1      17.4kB  20.5GB  20.5GB  ext3         /

(parted) rm 1
(parted) p
Error: /dev/sdm: unrecognised disk label
(parted) quit
Information: You may need to update /etc/fstab.
```

14. Exit the OSS node SSH login and repeat step 10 through step 13 for the other OSS nodes to identify the disk names for DM multipath.

15. After the disk names for multipath have been identified, edit finalize script (see Configure the Node Finalize Script for a CLFS Category on page 234 to modify the `site.esf_finalize.sh` script for the node category in `/opt/cray/esms/cray-es-finalize-scripts-XX/etc/`). Add specific WWID values to the `multipaths {}` section of the `/etc/multipaths.conf` file the make sure that the device has the same name every time the node boots. Specify a label for each disk, such as `/dev/mapper/ost5` or `/dev/mapper/mdt0`, then use that label for the Lustre configuration file.

    MGT and MDT aliases and WWID settings:

```
cat << EOMP >> /localdisk/etc/multipath.conf
multipaths {
   multipath {
      wwid  360080000000000000000000000000000
      alias mgt
   }
   multipath {
      wwid  360080000000000000000000000000000
      alias mdt0
   }
multipath {
      wwid  360080000000000000000000000000000
      alias mdt1
   }
}

EOMP
```

OST aliases and WWID settings:

```
cat << EOMP >> /localdisk/etc/multipath.conf
multipaths {
   multipath {
      wwid  360080000000000000000000000000000
      alias ost0
   }
   multipath {
      wwid  360080000000000000000000000000000
      alias ost1
   }
   multipath {
      wwid  360080000000000000000000000000000
      alias ost2
   }
   multipath {
      wwid  360080000000000000000000000000000
      alias ost3
   }
multipath {
      wwid  360080000000000000000000000000000
      alias ost4
   }
multipath {
      wwid  360080000000000000000000000000000
      alias ost5
   }
}

EOMP
```

16. Reboot all MDS and OSS nodes that were configured with multipath.
17. Confirm that the MDT device `/dev/mapper/mdt` is available on the MDS nodes.

```
esms1# ssh esfs-mds001 fdisk -l /mapper/mdt0
```

18. Confirm that all of the OST devices `/dev/mapper/ost*` devices are available on the OSS nodes.

```
esms1# ssh esfs-oss001 fdisk -l /dev/mapper/ost0
esms1# ssh esfs-oss001 fdisk -l /dev/mapper/ost1
esms1# ssh esfs-oss001 fdisk -l /dev/mapper/ost2
esms1# ssh esfs-oss001 fdisk -l /dev/mapper/ost3
esms1# ssh esfs-oss001 fdisk -l /dev/mapper/ost4
esms1# ssh esfs-oss001 fdisk -l /dev/mapper/ost5
```

19. The disk device name `"/dev/mapper/mdt0"` can be used for the MDT and MGT and `"/dev/mapper/ost0"` can be used for OST0 when configuring the Lustre `fs_defs` file and similarly for the other OST devices. The example

below shows how to configure `/dev/mapper/ost[0-5]` to reference the appropriate devices with even numbered OST devices on `oss002` and odd numbered OST devices on `oss001`.

```
## MDT
## MetaData Target
mdt: node=esfs-mds001
     dev=/dev/mapper/mdt0
     fo_node=esfs-mds002
     index=0
## MGT
## Management Target
mgt: node=esfs-mds001
     dev=/dev/mapper/mdt0
     fo_node=esfs-mds002
     index=0

## Object Storage Target(s)
ost: node=esfs-oss00[2,1]
     dev=/dev/mapper/ost[0-5]
     fo_node=esfs-oss00[1,2]
     index=0
```

5.19 SCSI RDAC Driver Kernel Parameters for Fibre Channel Storage

OSS and MDS slave nodes with Fibre Channel (FC) host bus adapters (HBAs) should use a different boot image than the OSS and MDS using SAS or IB HBAs. When booting an OSS or MDS with the CLFS software image, the `scsi_dh_rdac` driver is not loaded at the correct time. This causes nodes that are attached to storage via FC HBAs to encounter I/O errors, which (if enough LUNs are present) can significantly slow boot times. This condition can be corrected by preloading the RDAC module using a kernel parameter before the system starts the `qla2`*xxx* FC module. To create the FC software image, clone the exiting ESF software image and add the kernel parameter. The CLFS software image can be modified using the **Settings** tab from the `cmgui`. Enter `rdloaddriver=scsi_dh_rdac` in the **Kernel Parameters** field. Cray recommends this kernel parameter for CLFS nodes with Fibre Channel attached storage.

Procedure 84. Add SCSI RDAC kernel parameter to an ESF software image

1. Log in to the CIMS node as `root`.
2. From a UNIX® shell, copy the production ESF image (`ESF-XX-2.2.0-201401151643`), and wait for the copy operation to complete.

```
esms1# cd /cm/images
esms1# cp -pr ESF-XX-2.2.0-201401151643 ESF-XX-2.2.0-FC
```

3. Start `cmsh` and clone the functional CLFS software image.

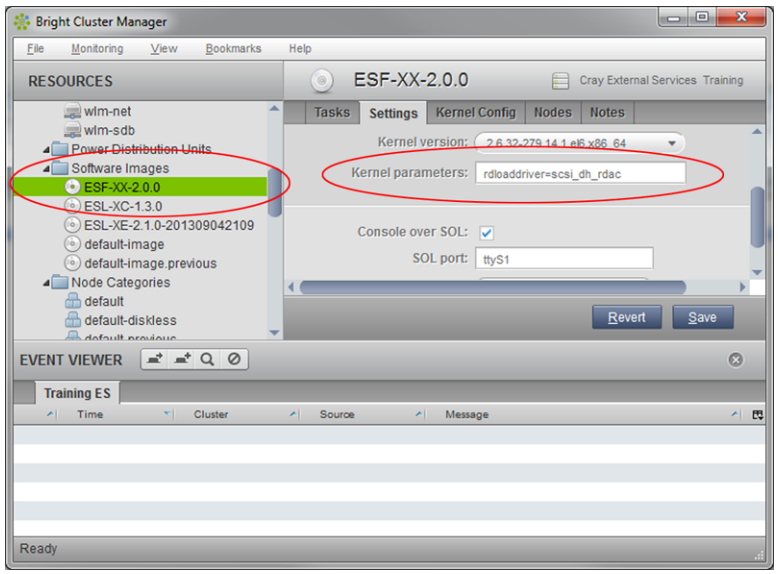
```
esms1# cmsh
[esms1]% softwareimage
[esms1->softwareimage]% listName (key)                Path
------------------------ ------------------------------------- ------------------------
ESF-XX-2.2.0-201401151643  /cm/images/ESF-XX-2.2.0-201401151643     2.6.32-358.18.1.el6.x86_64
ESL-XC-1.3.0               /cm/images/ESL-XC-1.3.0                  3.0.74-0.6.8-default
ESL-XE-2.1.0-201309042109  /cm/images/ESL-XE-2.1.0-201309042109     2.6.32.59-0.7-default
default-image              /cm/images/default-image                 3.0.80-0.5-default
default-image.previous     /cm/images/default-image.previous        3.0.80-0.5-default

[esms1->softwareimage]% clone ESF-XX-2.2.0-201401151643 ESF-XX-2.2.0-FC
[esms1->softwareimage*[ESF-XX-2.2.0-FC*]]% commit
[esms1->softwareimage[ESF-XX-2.2.0-FC]]%
```

4. Set kernel parameters to `rdloaddriver=scsi_dh_rdac`.

```
[esms1->softwareimage[ESF-XX-2.2.0-FC]]%  set kernelparameters rdloaddriver=scsi_dh_rdac
[esms1->softwareimage*[ESF-XX-2.2.0-FC*]]% commit
```

The `cmgui` can also be used to set kernel parameters. Select the software image from the Resources tree, then select **Settings**, and enter the kernel parameter `rdloaddriver=scsi_dh_rdac` in the **Kernel Parameters** field as shown in Figure 36.

Figure 36. Set RDAC Kernel Parameters for Fibre Channel



Lustre Procedures on DMP Systems [6]

6.1 Distributed Name Space Usage and Administration

In versions of Lustre prior to 2.5, all metadata operations were serviced by a single metadata server (MDS). When using distributed namespace (DNE) features, the metadata operations in one file system can be spread across multiple MDS nodes. Previously, the single MDS node could present a possible bottleneck in file system scaling; DNE enables multiple MDS's to share the load. Phase 1 DNE is not metadata striping, where metadata for a single directory is split across multiple MDS's to improve performance in almost all cases. Rather, it is a directory-assignment scheme, where responsibility for a particular directory's metadata is given to a specific MDS. This allows significant performance improvements for some workloads. Multiple MDTs can be exported from one MDS, and MDTs operate in active-active failover mode for metadata (active failover for data is already supported).

The root of the file system resides on MDT0; therefore, all directories are created in the root directory of the file system by default. When a directory is created on an MDT other than MDT0, the metadata for that directory and its child directories is served from only that MDS. MDT0 services all directories is not explicitly created on a different MDT.

For this release, DNE does not take action without administrator intervention. Administrators (`root`) must explicitly create directories on other MDTs (other than MDT0) using the `lfs mkdir` command. Plan directory locations in advance to spread the work load across MDTs and leverage the DNE performance benefits. The DNE performance improvements are not leveraged if all the file system activity occurs in one directory.

The `lfs mkdir` command is used to create directories on MDTs other than MDT0, which must be done manually to leverage the performance benefits provided by DNE.

6.1.1 Administrator Tools for DNE

The following administration tools are used to managed DNE:

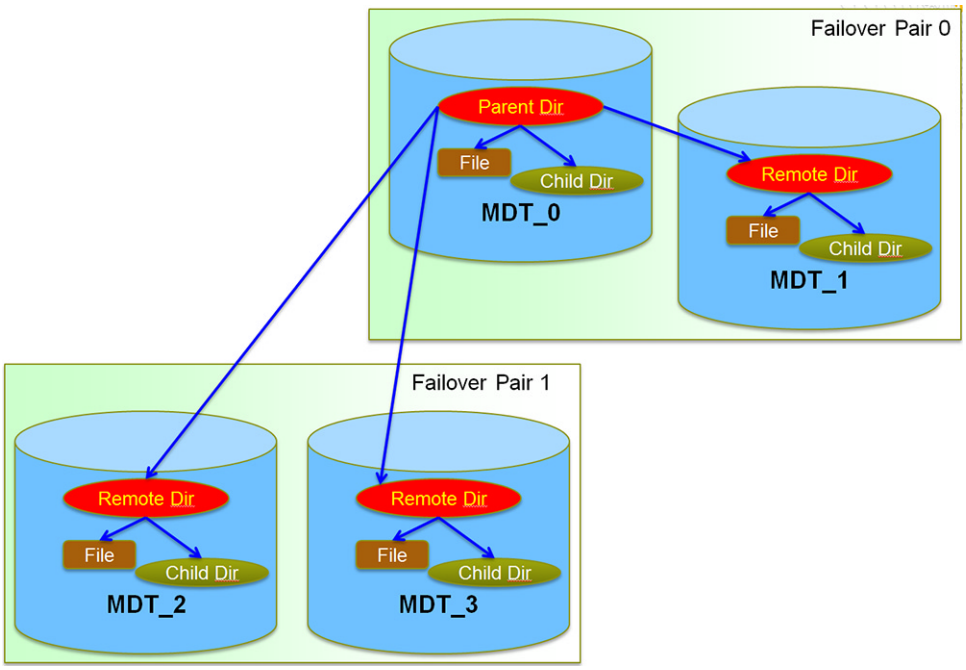
- `lustre_control` file system definitions (`fs_defs`) files
- `lustre_control` commands (`stop`, `start`, `status`, etc) with DNE enabled
- `lfs mkdir` command used to create remote directories

- `esfsmon` 2.0.0 tools used to manage failover for active-active CLFS nodes
- Lustre Monitoring Tool (LMT)

6.1.2 Remote Directories

MDT0 contains mount point for the file system labeled Parent Dir in Figure 37. Each pair of MDTs make a failover pair. Remote directories (labeled Remote Dirs) are created from Parent Dir on specified MDTs. Remote directory trees are contained on a single MDT (remote directory chaining is not currently supported by Cray). The right to create remote directories is delegated to a group or list of groups by a Lustre setting on each MDT. But without remote directory chaining, remote directories are only created starting from MDT0. For DNE Phase 1, Cray will support up to 4 MDS failover pairs.

Figure 37. DNE Remote Directories



Individual directories are explicitly placed on remote MDTs by a privileged system administrator. The right to create remote directories is delegated to a group or list of groups by a Lustre setting on each MDT. Remote directories can only be created from MDT0. Cray recommends that a directory on exist MDT0 that contains all the directory entries pointing to remote directories. Moving a remote directory to another MDT requires moving all the data and hard links across MDTs and is not supported.

6.1.3 Create Directories on MDTs Other Than MDT0

The examples in this section are run from a Lustre client (CDL node) with a Lustre file system mounted.

Usage: `lfs mkdir <--index |-i` *mdt_index*`> <`*my_remote_directory*`>`

Example 17 creates *my_remote_directory* and all of its child directories on MDT1 (but in the `/lus/` directory on MDT0).

Example 17. `lfs mkdir` creates remote directory on MDT1

```
eslogin1:/lus/# lfs mkdir -i 1 my_remote_directory
```

Example 18 creates a directory on MDT2, (if you had a third MDT). This command fails because of the rule that a directory that is not on MDT0 must have all its children on the same MDT.

Example 18. `lfs mkdir` fails using remote directory chaining

```
eslogin1:/lus/# lfs mkdir -i 2 my_remote_directory/some_other_directory
```

Example 18 uses remote directory chaining, which is not allowed. The root of the file system is always on MDT0 (`/lus/`). The `lfs mkdir` command attempts to create *some_other_directory* on MDT2 as a child of *my_remote_directory* which resides on MDT1. The `lfs mkdir` command restricts you to one change of MDT in a directory path name:

```
/lus/my_remote_directory/some_other_directory
  |-On MDT0  |           |
             |-On MDT1   |
                         |-On MDT2
```

6.1.4 Enable Non-root Users to Create Directories

Specific groups or all users can create directories only if a `proc` variable on each MDS is set. The `lctl` command controls Lustre via an `ioctl` interface, allowing access to configuration, maintenance and debugging features. The following example shows the how to configure the `enable_remote_dir_gid` parameter from the MGS so that it persists across reboots. Set the `conf_param` option to `0` to disable, a specific group ID (GID) number, or `-1` for all GIDs.

Example 19. Enable non-root users to create directories on MDT0000

Example 19 sets a permanent configuration parameter for `MDT0000` via the MGS. This command must be run on the MGS node.

```
mgs1# lctl conf_param filesystem-MDT0000.mdt.enable_remote_dir_gid=0 | GID | -1
```

Example 20 sets a permanent configuration parameter for `MDT0001` via the MGS. This command must be run on the MGS node.

Example 20. Enable non-root users to create directories on MDT0001

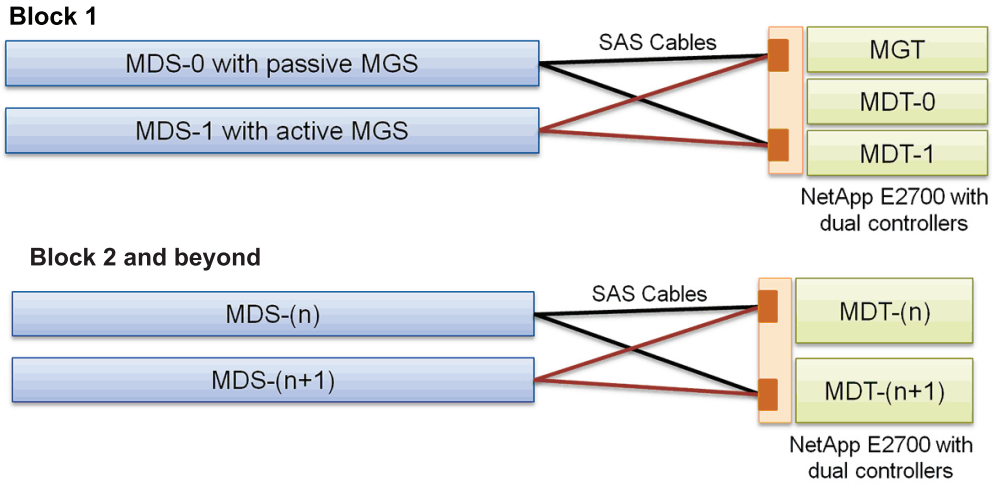
```
mgs1# lctl conf_param filesystem-MDT0001.mdt.enable_remote_dir_gid=0 | GID | -1
```

6.1.5 Hardware Requirements

Cray supports the distributed namespace (DNE) configurations in Figure 38. The maximum supported blocks is 4, which includes 8 servers, 8 MDTs, 1 MGT, and 4 NetApp 2700 dual controller storage arrays.

The NetApp 2700 storage array supports 12Gbit SAS controllers and is the preferred storage device for the MDT/MGT interconnect. Each NetApp 2700 is limited to one 24-bay tray with 2.5-in disk drives. More MDS/MDT blocks may be added for systems requiring large inode counts. All NetApp 2700s in a CLFS system are configured identically. The failover configuration is standard and required.

Figure 38. CLFS DNE Supported Hardware Configurations



6.1.5.1 Failover

The hardware configurations in Figure 38 support active-active failover using `esfsmon` 2.0.0. Refer to the section about `esfsmon` for configuring failover on DNE configurations.

6.1.6 Add MDTs

MDTs can be added as long as they comply with Cray's supported hardware configurations discussed in Hardware Requirements on page 250. A general procedure for this task follows:

Procedure 85. Add MDTs

1. Create a new `fs_defs` file that includes the new MDT.

2. Determine the index of your MDT (0,1,2, etc, in the order listed in the `fs_defs` file.
3. Install the `fs_defs` file.
4. Boot the new MDT and use the following command to format the file system:

`lustre_control reformat -l` *filesystemname*`-MDT`*index*

5. Enter the command below to add a second MDT (MDT 1) on a file system named `scratch`.

```
mdt1# lustre_control reformat -l scratch-MDT0001
```

6. Use `lustre_control` to start the system.

6.2 Migrating from Lustre® 1.8.x to 2.5

Caution: The CIMS node must run the most recent released version of ESM software (ESM XX-3.0.0) and ESF software (ESF XX-2.2.0) in order to migrate from Lustre 1.8.x to 2.5.

This section describes the format differences between Lustre 1.8.x and 2.5, their purpose, and the process of upgrading from 1.8.x to 2.5. See Procedure 86 on page 255.

6.2.1 Related Publications

The following documents contain additional information that may be helpful:

- *Managing Lustre for the Cray Linux Environment (CLE)* (S–0010)
- *Installing Lustre File System by Cray (CLFS) Software* (S–2521)

Note: The Bright administration guide and user guides are stored on the CIMS node (as Adobe® Acrobat® PDF files) in the `/cm/shared/docs/cm` directory.

6.2.2 Introduction

Lustre 2.5 represents a significant advance in Lustre design with the addition of many new features and support for future improvements. To accommodate the new features, 2.5 uses a somewhat different on-disk file system format than the one used by 1.8.x. The format differences are limited to Lustre's internal metadata about the file system. They do not affect the format or storage of user data. The format differences fall into two categories: those related to file identifiers (FIDs) that replace inodes in some cases and those related to quota support.

Lustre 2.5 provides tools to add the new metadata structures used by 2.5 to an existing 1.8.x file system. Some of these tools run automatically when the 2.5 servers are started; some require the administrator to perform explicit upgrade operations at the time 2.5 is installed.

6.2.3 FIDS

A file identifier (FID), is a unique identifier for a Lustre file or object. It is independent of the back end file system, for example, `ldiskfs`. Lustre 1.8.x uses inodes to uniquely identify the objects belonging to a file. In Lustre 2.5, FIDs replace inodes for this purpose. The drawback of using inodes is that a file's inode can change over the life of the file. For example, when a file is restored from a file level backup, it will be assigned a new inode/generation number. Afterwards, Lustre 1.8.x can no longer locate its objects. FIDs, on the other hand, never change once assigned. Following restore from a file level backup, the FIDs are still correct and Lustre 2.5 can locate its objects. Lustre 2.5, in addition to supporting the device level backup of 1.8.x, also supports file level backup and restore. Lustre 2.5 still uses inodes internally to interact with the `ldiskfs` back end file system.

To facilitate this interaction, Lustre 2.5 maintains a map of FIDs to inodes. This map is called the *Object Index* (OI). When upgrading from 1.8.x to 2.5 the OI must be created through a process called *OI scrub*. The OI scrub occurs automatically when a 1.8.x formatted file system without an OI is mounted by 2.5 servers. An OI scrub may also be triggered if Lustre discovers a missing or bad FID to inode mapping during normal operation. Finally, an OI scrub can be started manually by running `lfsck`.

`lfsck` checks and repairs errors in the OI. The OI maintains the FID to inode mapping, the inode to FID mapping is stored in the inode itself. The FID of the object identified by the inode is stored in the extended attributes area of the inode known as *linkEA*. The inode also contains the FID of the file's parent directory. This feature is known as *FID-in-dirent*. The linkEA and FID-in-dirent information enables Lustre to efficiently generate full path names from the inode. These names are used in POSIX-style path name permission checks, to produce better error messages and to support changelog applications like `lustre_rsync`.

Storing the parent FID in the inode also improves the performance of `readdir` and other directory operations. Unlike the OI, which Lustre 2.5 requires, the FID information in the inode is optional. If the FID information in the inode is missing, then directory lookup performance is affected and `changelog` features will not be fully supported, but otherwise Lustre will be totally functional. The FID information is optional, and the upgrade process to add FID information to the inode is also optional.

To populate the inodes with the appropriate FIDs, the Lustre administrator must set the `dirdata` attribute on each MDT and then run `lfsck` with the `-t` namespace option. The `lfsck` process runs in the background; the rate at which it updates inodes can be tunable parameter under the control of the system administrator. After the `dirdata` attribute is set, the Lustre file system cannot be downgraded to work with 1.8.x servers.

6.2.4 Quota Support

Lustre 2.5 addresses several limitations of the previous quota design. Among the improvements to quotas in 2.5 are:

- Quota limits can be changed while slaves are offline
- OSTs can be added and deleted without corrupting space usage statistics
- Master recovery can be completed without all targets being online
- A full `quotacheck` is no longer required following `e2fsck`
- Quota enforcement is enabled/disabled by file system rather than per-target
- Infrastructure is restructured for future growth, better performance, and improved functionality

To support these improvements, Lustre 2.5 has changed both the on-disk format of quota information and the user interface to quota functionality.

Quota specification has three components: space usage accounting, quota limit definition, and enabling enforcement.

Space Usage Accounting: In previous versions of Lustre, the `lfs quotacheck` must be run to generate the database of space used on each target by each user and group. In Lustre 2.5, the `quotacheck` command is deprecated. Instead, newly formatted 2.5 file systems have space usage accounting enabled by default, and the statistics are automatically kept up to date. When upgrading from 1.8.x to 2.5, the usage statistics must be initialized for existing files before quota limits can be enforced. The statistics are generated by running `tunefs.lustre --quota` on each target. This `tunefs` command also sets the `QUOTA` attribute of the target, which enables automatic accounting when the target is mounted.

Initializing usage statistics is a one time operation. It can be done as part of the upgrade process or sometime later. Note however that quota enforcement and accounting are disabled until the `tunefs.lustre --quota` command is executed on all targets.

Quota Limit Definition: The definition of user and group quota limits does not change with 2.5. The storage format does change. Prior to Lustre 2.5, the quota limits are stored in a file specific to the back end file system. With 2.5, this information is moved in a Lustre defined index along with other Lustre metadata. The quota limits are converted automatically to the new format and storage location when the MDT is upgraded to 2.5.

Enable Quota Enforcement: In Lustre 2.5, quota enforcement is independent of the space usage accounting. The accounting information is always maintained, even when enforcement is disabled. Enforcement is enabled/disabled for the entire file system. The command is:

Example 21. Enable quota enforcement

```
lctl <set | conf> _param fsname/quota/ost | mdt=u | g | ug | none
```

The `lfs quotaon|off` command and per-target `quota_type` parameter are no longer used in Lustre 2.5.

6.2.5 Performance Expectations

For optimum long term performance and functionality, all of the disk format changes described above are recommended. However, each of the upgrade processes has a performance cost.

6.2.5.1 Object Index Creation and Repair

OI scrub is designed to have minimal impact on system performance. It runs in the background while the file system remains online. Clients can continue to access files while it runs. The system administrator can control the overhead of the scrubbing process by tuning the maximum number objects examined per second. If no limit is set, OI scrub will run as fast as possible. On an unloaded system, with no limit set, experiments have shown OI scrub to process in excess of 100,000 objects per second.

OI scrub status can be monitored through the `/proc` file:

`osd-ldisk/`*`mdt_device`*`/oi_scrub`

6.2.5.2 Add FIDs to inodes

Updating all the inode extended attributes to include the related FID and parent FID information is a one time operation. (Note, updates to individual inodes may occur when `lfsck` repairs file system corruption.) The process is similar to an explicitly invoked OI scrub and has similar performance characteristics.

Progress of the inode update can be monitored with the `/proc` file:

`mmd/`*`mdt_device`*`/lfsck_namespace`

6.2.5.3 Space Usage Statistics

When upgrading a 1.8.x formatted file system, a database of the space usage statistics must be created. The usage data is gathered and stored when the quota flag is set on each target. The speed of the data collection in 2.5 is similar to the speed of an `lfs quotacheck` in earlier Lustre versions.

After the initial creation, Lustre updates the space usage statistics automatically as files change. Updating the statistics does impose overhead and has been reported to affect metadata performance by as much as 5%. Cray internal testing has shown that the accounting overhead has no measurable effect on performance.

6.2.6 Upgrade Procedure

This procedure is scripted, and assumes the person performing this update has experience with the Cray Linux Environment (CLE) and Lustre administration.

Procedure 86. Upgrade Lustre 1.8.x to Lustre 2.5

1. If an `.fs_defs` file does not exist, create the file to define the structure of the 1.8.x file system. See *Managing Lustre for the Cray Linux Environment (CLE)* (S–0010).

   a. Use `umount` to unmount Lustre file system on all clients.

   b. Stop Lustre servers.

   c. Install Lustre 2.5 RPMs on all CLFS servers.

   d. Log in to the CLFS node. **Do NOT** start Lustre yet.

   e. If the `.fs_defs` file has not been installed, do so now, **being very careful not to start Lustre**. From the CIMS node, enter:

```
esms1# lustre_control install filename.fs_defs
```

Caution: The following upgrade procedures require `e2fsprogs` version `1.42.3.wc1` or later. Check the installed version by running `dumpe2fs` without any parameters. **Do NOT** proceed with these instructions if the version is not at least `1.42.3.wc1`.

2. Initial mount: Create object index (OI).

   a. Regenerate configuration logs on all targets. If the `write_conf` operation reports a disk device not available, repeat the `write_conf` command.

```
esms1# lustre_control write_conf -f fsname
esms1# lustre_control start -p -f fsname
```

b. To mount CDL or internal login node clients:

```
esms1# lustre_control mount_clients -f fsname
boot:~> lustre_control mount_clients -f fsname
```

To mount compute node clients:

```
boot:~> lustre_control mount_clients -c -f fsname
```

c. Verify that the file system can be accessed from clients. Use the `ls`, `touch`, `cat`, commands etc. on Lustre file system or another procedure to verify that a Lustre file system is operational.

3. `lfsck`: Add FIDs to inode attributes.

Caution: After the `dirdata` attribute is set on the MDT, 1.8.x servers will no longer be able to mount the file system. Cray strongly recommends that you set the `dirdata` attribute to get the best performance and complete functionality from the Lustre 2.5 file system.

a. Shutdown Lustre.

```
boot:~># lustre_control umount_clients -c -f fsname
boot:~># lustre_control umount_clients -f fsname

esms1# lustre_control umount_clients -f fsname
esms1# lustre_control stop -f fsname
```

b. Set `dirdata` attribute on the MDT. The state of the `dirdata` attribute can be checked before and after the `tune2fs` command by dumping the superblock of the MDT using `dumpe2fs`.

```
esms1# ssh MDS_node
mds# tune2fs -O dirdata MDT-device
mds# exit
esms1#

mds# dumpe2fs -h MDT-device | grep 'Filesystem features'
```

An example of attributes before the `tune2fs` command:

```
dumpe2fs 1.42.7.wc1 (12-Apr-2013)
[Filesystem features: has_journal ext_attr resize_inode dir_index filetype
sparse_super large_file uninit_bg quota]
```

An example of attributes after the `tune2fs` command:

```
dumpe2fs 1.42.7.wc1 (12-Apr-2013)
[Filesystem features: has_journal ext_attr resize_inode dir_index filetype
dirdata sparse_super large_file uninit_bg quota]
```

c. Start Lustre servers.

```
esms1# lustre_control start -f fsname
```

d. Update inode attributes.

```
esms1# ssh MDS_node
mds# lctl lfsck_start -M fsname-MDT0000 -t namespace
mds# exit
esms1#
```

The `lfsck` process performs periodic checkpoints so it can resume from where it left off if it is stopped or interrupted. `lfsck` progress can be monitored by watching the `lfsck_namespace` `/proc` file:

```
mds# lctl get_param mdd/fsname-MDT0000/lfsck_namespace
```

Before running the `lfsck -t namespace` command, the output looks like this:

```
mdd.fsname-MDT0000.lfsck_namespace=
name: lfsck_namespace
magic: 0xa0629d03
version: 2
status: init
flags:
param:
time_since_last_completed: N/A
time_since_latest_start: N/A
time_since_last_checkpoint: N/A
latest_start_position: N/A, N/A, N/A
last_checkpoint_position: N/A, N/A, N/A
first_failure_position: N/A, N/A, N/A
checked_phase1: 0
checked_phase2: 0
updated_phase1: 0
updated_phase2: 0
failed_phase1: 0
failed_phase2: 0
dirs: 0
M-linked: 0
nlinks_repaired: 0
lost_found: 0
success_count: 0
run_time_phase1: 0 seconds
run_time_phase2: 0 seconds
average_speed_phase1: 0 items/sec
average_speed_phase2: 0 objs/sec
real-time_speed_phase1: N/A
real-time_speed_phase2: N/A
current_position: N/A
```

These values will be updated while `lfsck` is running. Once it has completed, the `success_count` value will go up by 1.

4. (Optional) Enable quotas.

Caution: Some small regressions in metadata performance have been attributed to the automatic usage accounting. If quota enforcement is not needed, step 4 can be skipped. Be aware however that the default configuration for Lustre 2.5 enables automatic accounting. Disabling automatic accounting, although possible, is not a documented feature.

a. Stop the Lustre servers.

```
esms1# lustre_control stop -f fsname
```

b. Enable accounting and create space usage database. The `lustre_control set_quota_flag` operation executes `tunefs.lustre --quota` on each Lustre target in the specified file system. The `tunefs.lustre` command sets the quota feature flag in the superblock of the target and runs `e2fsck` to build the per-UID/GID disk usage database.

```
esms1# lustre_control set_quota_flag -f fsname
```

c. Enable enforcement.

```
esms1# ssh MGS-node
mgs# lctl <set|conf>_param fsname/quota/<ost|mdt>=<u|g|ug|none>
mgs# exit
```

5. (Optional) Perform an OI scrub. The upgrade process performs an OI scrub to create the FID to inode mappings automatically. The following commands start and stop OI scrub and are not required to complete the upgrade, but are included here for reference. These commands are run on the MDS node.

a. Start OI scrub.

```
mds# lctl lfsck_start -M fsname-MDT0000
```

b. Stop OI scrub.

```
mds# lctl lfsck_stop -M fsname-MDT0000
```

c. Tune OI scrub speed.

```
mds# lctl lfsck_start -M fsname-MDT0000 -s Max_Objects_Persecond
```

d. Monitor OI scrub status.

```
mds# lctl get_param osd-ldiskfs/fsname-MDT0000/oi_scrub
```

Example of output from a small test system after the scrub has completed:

```
name: OI_scrub
magic: 0x4c5fd252
oi_files: 64
status: completed
flags:
param:
time_since_last_completed: 711866 seconds
time_since_latest_start: 711916 seconds
time_since_last_checkpoint: 711866 seconds
latest_start_position: 12
last_checkpoint_position: 268435457
first_failure_position: N/A
checked: 628157
updated: 504
failed: 0
prior_updated: 0
noscrub: 0
igif: 0
success_count: 1
run_time: 49 seconds
average_speed: 12819 objects/sec
real-time_speed: N/A
current_position: N/A
```

6.3 Configure Lustre® File Systems on CLFS Nodes From the CIMS Node

This section describes how to configure Lustre file systems on CLFS nodes from the CIMS node.

Procedure 87. Configure Lustre file systems on CLFS nodes from the CIMS Node

1. Log in to the CIMS as `root`.
2. Copy the `example.fs_defs` file in `/opt/cray/esms/cray-lustre-control-XX/default/etc/` to `/opt/cray/esms/cray-lustre-control-XX/etc/scratch.fs_defs` and prevent it from being overwritten during software updates. Lustre file system names must be eight characters or less. This procedure creates a file system called `scratch`.

```
esms1# mkdir -p /opt/cray/esms/cray-lustre-control-XX/etc
esms1# cd /opt/cray/esms/cray-lustre-control-XX/etc
esms1# cp -p /opt/cray/esms/cray-lustre-control-XX/default/etc/example.fs_defs scratch.fs_defs
```

3. Display the Lustre network identifier (NID) map information to include in the `scratch.fs_defs` file.

```
esms1# lustre_control dump_nid_map -w esfs-mds00[1-2],esfs-oss00[1-4]
Performing 'dump_nid_map' from esms1 at Thu Jan 31 14:47:49 CST 2013

Hostname to LNET nid mapping for the nodes:
esfs-mds001,esfs-mds002,esfs-oss001,esfs-oss002,esfs-oss003,esfs-oss004
nid_map: nodes=esfs-mds00[1-2]  nids=10.149.0.[1-2]@o2ib
nid_map: nodes=esfs-oss00[1-4]  nids=10.149.0.[3-6]@o2ib
```

4. Configure device names in the `scratch.fs_defs` file. This can be done with either the disk device names configured earlier for multipath or the persistent device names via the method in this section.

   For a multipath configuration, device names should link to a logical alias under `/dev/mapper/`*alias*. Using the `/dev/mapper/`*alias* allows the target definition to be condensed considerably for large file systems using multipath. For example, a set of 12 LUNs whose alias contains 12 separate `"ost: dev=xxx node=xxx..."` lines can be represented as:

   ```
   ost: node=esfs-oss[1,2] dev=/dev/mapper/lun[0-11] index=[0-11]
   ```

   This assigns the even numbered LUNs to `esfs-oss1` and the odd numbered LUNs to `esfs-oss2`. This can be configured in the site node finalize script (`site.esf_finalize.sh`) either for the node, or the node category (such as `esfs-even-`*filesystem*). Multipath configuration device names can also be configured as a `dm-uuid` path name such as `/dev/disk/by-id/dm-uuid-mpath-`*wwid*.

   To configure device names in a non-multipath configuration, use a device mapper to assign names to the disk devices. The following command lists disks by ID.

```
esms1# ssh esfs-mds001
Last login: Fri Apr 19 11:08:03 2013 from esms1.cm.cluster
[root@esfs-mds001 by-id]# ls -l /dev/disk/by-id
total 0
lrwxrwxrwx 1 root root  9 Apr 18 21:14 scsi-3600508e000000000faef8330f46c9c09 -> ../../sda
lrwxrwxrwx 1 root root 10 Apr 18 21:14 scsi-3600508e000000000faef8330f46c9c09-part1 -> ../../sda1
lrwxrwxrwx 1 root root 10 Apr 18 21:14 scsi-3600508e000000000faef8330f46c9c09-part2 -> ../../sda2
lrwxrwxrwx 1 root root  9 Apr 18 16:43 scsi-3600a0b800026cfe400002dfd4becf544 -> ../../sdb
lrwxrwxrwx 1 root root  9 Apr 18 21:14 wwn-0x600508e000000000faef8330f46c9c09 -> ../../sda
lrwxrwxrwx 1 root root 10 Apr 18 21:14 wwn-0x600508e000000000faef8330f46c9c09-part1 -> ../../sda1
lrwxrwxrwx 1 root root 10 Apr 18 21:14 wwn-0x600508e000000000faef8330f46c9c09-part2 -> ../../sda2
lrwxrwxrwx 1 root root  9 Apr 18 16:43 wwn-0x600a0b800026cfe400002dfd4becf544 -> ../../sdb
```

   The persistent device name for `sdb` is `scsi-3600a0b800026cfe400002dfd4becf544`. Enter the value in the device configuration section of `scratch.fs_defs`. Either option in step 4 is a link that points to the `/dev/dm-`*x* device.

5. Edit the `scratch.fs_defs` file to set values.

```
esms1# vi /opt/cray/esms/cray-lustre-control-XX/etc/scratch.fs_defs
```

6. Modify the `fs_name`, `nid_map`, `mdt`, `mgt`, `ost`, and `stripe_count` settings in the `scratch.fs_defs` file.

   a. Enter the `fs_name:` (such as `scratch`).

```
# file system name - must be 8 characters or less
fs_name: scratch
```

   b. Enter the `nid_map` information obtained in step 3.

```
nid_map: nodes=esfs-mds00[1-2] nids=10.149.0.[2-3]@o2ib
nid_map: nodes=esfs-oss00[1-4] nids=10.149.0.[4-7]@o2ib
```

   c. Enter the `mdt` and `mgt` definitions and disk device name(s) for your system obtained from step 4. Include the failover node definition (`fo_node`) if configuring a failover system. If there is no failover node definition, remove this line. In this example, the metadata target (MDT) and management target (MGT) functions share the same CLFS node (`esfs-mds001`).

```
## MDT
## MetaData Target
mdt: node=esfs-mds001
     dev=/dev/mapper/mdt0
     fo_node=esfs-mds002

## MGT
## Management Target
mgt: node=esfs-mds001
     dev=/dev/mapper/mdt0
     fo_node=esfs-mds002
```

7. Enter the `ost` definitions and disk device name(s) for your system obtained from

step 4. Include the failover node definition (`fo_node`) if configuring a failover system. If there is no failover node definition, remove this line. Include the target index value to simplify management of a large number of targets.

```
## OST
## Object Storage Target(s)
ost: node=esfs-oss001
     dev=/dev/mapper/ost0
     fo_node=esfs-oss002
     index=0

ost: node=esfs-oss002
     dev=/dev/mapper/ost1
     fo_node=esfs-oss001
     index=1

ost: node=esfs-oss001
     dev=/dev/mapper/ost2
     fo_node=esfs-oss002
     index=2

ost: node=esfs-oss002
     dev=/dev/mapper/ost3
     fo_node=esfs-oss001
     index=3

ost: node=esfs-oss001
     dev=/dev/mapper/ost4
     fo_node=esfs-oss002
     index=4

ost: node=esfs-oss002
     dev=/dev/mapper/ost5
     fo_node=esfs-oss001
     index=5
```

8. Exit and save the `scratch.fs_defs` file and proceed to Configure Lustre® File Systems on MDS and OSS Nodes on page 262.

6.4 Configure Lustre® File Systems on MDS and OSS Nodes

Procedure 88. Configure Lustre file systems on MDS and OSS nodes

1. Log in to the CIMS node as `root` and install the Lustre file system definitions.

```
esms1# cd /opt/cray/esms/cray-lustre-control-XX/etc
esms1# lustre_control install scratch.fs_defs
Performing 'install' from esms1 at Mon Apr 22 14:38:42 CDT 2013
Parsing file system definitions file: scratch.fs_defs
Parsed file system definitions file: scratch.fs_defs
The 'scratch' file system definitions were successfully installed!
```

2. Format the Lustre file system (`scratch`). You must enter `y` to enable the `lustre_control` command to reformat the file system.

```
esms1# lustre_control reformat -f scratch
Performing 'reformat' from esms1 at Mon Apr 22 14:42:52 CDT 2013

About to reformat all targets for the following file system(s):
scratch

Continue? (y|n|q)
y
```

3. Start the Lustre file system `scratch`.

```
esms1# lustre_control start -p -f scratch
```

4. Check the status of `scratch`.

```
esms1# lustre_control status -f scratch
Performing 'status' from esms1 at Mon Apr 22 14:44:14 CDT 2013

File system: scratch
Device              Host         Mount           OST Active     Recovery Status
MGS                 esfs-mds001  Mounted         N/A            Unknown
MGS*                esfs-mds002  Unmounted       N/A            N/A
scratch-MDT0000     esfs-mds001  Mounted         N/A            INACTIVE
scratch-MDT0000*    esfs-mds002  Unmounted       N/A            N/A
scratch-OST0000     esfs-oss1    Mounted         Active         INACTIVE
scratch-OST0000*    esfs-oss2    Unmounted       Active         N/A
.
.
.
```

6.5 Configure Lustre® Monitoring Tool (LMT) and Cerebro on the CIMS Node

Refer to the man pages for `ltop`(1), `lmt.conf`(5), `lmtinit`(5), `lmtsh`(8), and `lmt_agg.cron`(8) for more information.

Refer to the Cerebro man pages: `cerebro_module`(3), `cerebro_module_devel`(3), `cerebro.conf`(5), `cerebro`(7), `cerebrod`(8), `cerebro-stat`(8), and `cerebro-admin`(8).

6.5.1 Configure Cerebro

This configuration is explained in terms of the LMT server (CIMS node) and the LMT agents (CLFS nodes). The `lmt-server` package is installed on the CIMS node. The `lmt-server-agent` package is installed on CLFS nodes (on an ESF software image).

The Cerebro configuration file, `/etc/cerebro.conf`, contains comments that describe the default settings. You can also view this information in the `man` page for `cerebro.conf`.

Procedure 89. Configure Cerebro

1. Log in to the CIMS node as `root`.
2. Edit the `/etc/cerebro.conf` file on the CIMS node.

```
esms1# vi /etc/cerebro.conf
```

3. Make sure it contains the following lines. This configuration causes the CIMS node to listen for Cerebro messages on its IP address without sending any of its own.

```
cerebrod_speak off
cerebrod_listen on
cerebrod_listen_message_config LMT-SERVER-IP
```

4. Use the `chroot` shell to edit the software image used for CLFS node Lustre servers.

```
esms1# chroot /cm/images/ESF-XX-2.2.0-201401151643
[root@esms1 /]# vi /etc/cerebro.conf
```

5. Make sure the `/etc/cerebro.conf` file on the CLFS node Lustre servers contains the following lines. This causes the CLFS nodes to send Cerebro messages to the IP address of the CIMS node without listening for messages.

```
cerebrod_speak on
cerebrod_speak_message_config LMT-SERVER-IP
cerebrod_listen off
```

6. Exit the `chroot` shell.

```
[root@esms1 /]# exit
esms1#
```

The Cerebro daemon (`cerebrod`) can be configured to start automatically on the CIMS node and on the Lustre servers using `chkconfig`. To do so, run the following command on the CIMS node and on the CLFS node Lustre server software image.

```
esms1# chkconfig --level 235 cerebrod on
```

This causes `cerebrod` to start whenever `init` is invoked in run levels 2, 3, and 5. These are the run levels for multiuser mode, multiuser mode with networking, and multiuser mode with networking and X11, respectively. Turning on `cerebrod` with `chkconfig` does not start `cerebrod` immediately. It starts whenever the system is booted in run levels 2, 3, and 5. After completing the `chkconfig` commands on the CIMS and CLFS node software image start `cerebrod` manually as described below.

6.5.2 Start Cerebro and LMT

The `cerebrod` daemon can be started manually, if `chkconfig` has not been used to start it on boot, or if the system has not been restarted since the `chkconfig` was issued. To begin sending data into LMT, start the `cerebrod` on all CLFS nodes and the CIMS node as follows:

```
esms1# pdsh -w NodeList "/sbin/service cerebrod start"
esms1# /sbin/service cerebrod start
```

Example 22. Start `cerebrod` manually

```
esms1# pdsh -w esfs-mds[1,2],esfs-oss[1,2,3,4] "/sbin/service cerebrod start"
esms1# /sbin/service cerebrod start
```

6.5.3 Configure the MySQL Database for Cerebro and LMT

After LMT and Cerebro have been installed, the MySQL database must be configured and the `cerebrod` on the CIMS node must be restarted before data will be added to the database.

Procedure 90. Configuring the MySQL database for Cerebro and LMT

1. Log in to the CIMS node as `root`.
2. Edit the `mkusers.sql` file to change the password from `mypass` to the site password.

```
esms1# chmod 600 /usr/share/lmt/mkusers.sql
esms1# vi /usr/share/lmt/mkusers.sql
```

   - Edit the GRANT statements to grant privileges on only `filesystem_`*filesytem_name*`.*` where *filesystem* is the name of the file system. This grants permissions on the database for the file system being monitored.
   - Edit the password for `lwatchadmin` by changing `mypass` to the desired password. Also add a password for the `lwatchclient` user.

   Here is an example `mkusers.sql` script where the file system is named `scratch`, and the desired passwords for `lwatchclient` and `lwatchadmin` are `foo` and `bar`.

```
CREATE USER 'lwatchclient'@'localhost' IDENTIFIED BY 'foo';
GRANT SELECT ON filesystem_scratch.* TO 'lwatchclient'@'localhost';

CREATE USER 'lwatchadmin'@'localhost' IDENTIFIED BY 'bar';
GRANT SELECT,INSERT,DELETE  ON filesystem_scratch.* TO 'lwatchadmin'@'localhost';
GRANT CREATE,DROP           ON filesystem_scratch.* TO 'lwatchadmin'@'localhost';

FLUSH PRIVILEGES;
```

3. Enter the following command and type `root` password when prompted to create the LMT MySQL users.

```
esms1# mysql -u root -p < /usr/share/lmt/mkusers.sql
```

4. Change the permissions of `/etc/lmt/lmt.conf` so that only root has read access, then edit the file.

```
esms1# chmod 600 /etc/lmt/lmt.conf
esms1# vi /etc/lmt/lmt.conf
```

5. Edit the LMT configuration file (`/etc/lmt/lmt.conf`) to add the passwords for the users `lwatchclient` and `lwatchadmin`. Replace `nil` in the following line with the `lwatchclient` password in double quotation marks (`foo` in the example):

```
esms1# lmt_db_ropasswd = "foo"
```

6. Create the `/etc/lmt/rwpasswd` file, enter the `lwatchadmin` password in the file (`lmt.conf` reads the password from this file), and save the file. It should be accessible only by the `root` user.

```
esms1# touch /etc/lmt/rwpasswd
esms1# chmod 600 /etc/lmt/rwpasswd
esms1# vi /etc/lmt/rwpasswd
```

7. (Optional) A similar password file scheme can be used for the `lwatchclient` user if desired. To do so, replace the `lmt_db_ropasswd` line with the following:

```
f = io.open("/etc/lmt/ropasswd")
if (f) then
  lmt_db_ropasswd = f:read("*l")
  f:close()
else
  lmt_db_ropasswd = nil
end
```

8. (Optional) Complete this step if step 7 was performed. Otherwise, proceed to step 9.

   Create the `/etc/lmt/ropasswd` file, type the `lwatchclient` password in the file, and save the file. It should be accessible only by the `root` user. Creating separate files for both passwords enables `/etc/lmt/lmt.conf` to be readable by anybody, while protecting the passwords for `lwatchadmin` and `lwatchclient`.

```
esms1# touch /etc/lmt/ropasswd
esms1# chmod 600 /etc/lmt/ropasswd
esms1# vi /etc/lmt/ropasswd
```

9. Create the database for the file system being monitored.

```
esms1# lmtinit -a filesystem
```

10. Restart `cerebrod` on the CIMS node.

```
esms1# /sbin/service cerebrod restart
```

11. Verify that LMT is adding data to its MySQL database by using `lmtsh` to bring up the LMT shell. Then enter **`t`** to list tables. If the row count increases when you enter **`t`** again after 5 seconds, then LMT is configured properly.

```
esms1# lmtsh -f filesystem
```

12. Use the `ltop` command to display real time information about a Lustre file system.

```
esms1# ltop -f filesystem
```

6.5.4 Use LMT Aggregate Scripts to Manage the Data

There are two ways to view data provided by LMT. You can view live data with `ltop`, or you can view historical data from the MySQL database with `lmtsh`. Refer to the `man` pages for each command.

You can also access the MySQL database directly to view the data if you need more control over how the data is presented.

LMT provides scripts which aggregate data into the aggregate tables in the MySQL database. To run the aggregation scripts, enter the following:

Example 23. Use LMT aggregate scripts to manage the data

```
esms1# /usr/share/lmt/cron/lmt_agg.cron
```

This command may take some time to complete, but subsequent executions will be much faster. To see the tables which were populated by the aggregation scripts, use `lmtsh`. The aggregation script can be set up to run as a cron job if you would like the aggregated tables to be populated on a regular basis. Use **`crontab -e`** to enter the crontab editor, and enter the line below to set up a cron job:

Example 24. Setup LMT Cron Job

```
esms1# crontab -e
0 * * * * /usr/share/lmt/cron/lmt_agg.cron
```

LMT does not provide an automated utility for clearing old data from the MySQL database; this must be done manually using MySQL commands. For example, to clear all data from the `MDS_OPS_DATA` table which is older than October 4th at 15:00:00, run the following `mysql` command:

Example 25. Clear old Ddata from LMT MySQL database

```
esms1# mysql -p -e "use filesystem_filesystem;
delete MDS_OPS_DATA from
MDS_OPS_DATA inner join TIMESTAMP_INFO
on MDS_OPS_DATA.TS_ID=TIMESTAMP_INFO.TS_ID
where TIMESTAMP < '2013-10-04 15:00:00';"
```

6.5.5 Stop the Cerebro Service

To stop Cerebro from sending data to LMT, stop the Cerebro daemon (`cerebrod`) from running on all Lustre servers and the LMT server. *NodeList* in the following command can be `esfs-mds[1,2],esfs-oss[1,2,3,4]`.

```
esms1# pdsh -w NodeList "/sbin/service cerebrod stop"
esms1# /sbin/service cerebrod stop
```

If `cerebrod` has been turned on with `chkconfig`, it can also be turned off with `chkconfig` so that it does not start every time the system is booted. To turn off `cerebrod`, use:

```
esms1# chkconfig --level 235 cerebrod off
```

6.5.6 Delete the LMT MySQL Database

To delete the LMT MySQL database, enter the following command where *filesystem* is the name of the file system you would like to remove.

```
esms1# lmtinit -d filesystem
```

To remove the MySQL users added by LMT, run the following MySQL command:

```
esms1# mysql -u root -p -e "drop user 'lwatchclient'@'localhost'; drop
user 'lwatchadmin'@'localhost';"
```

6.5.7 Manage Cerebro with Bright

The `cerebrod` service can be started, stopped, and monitored using Bright. The following procedure adds the Cerebro service to the `esfs-mds1` node. This same procedure can be used to monitor the `cerebrod` service on the CIMS node.

Procedure 91. Managing Cerebro on a slave node with Bright

1. Log in to the CIMS node as `root` and start `cmsh`.

```
esms1# cmsh
[esms1]%
```

2. Switch to `device services` mode and select the `esfs-mds1` node.

```
[esms1]% device services esfs-mds1
[esms1->device[esfs-mds1]->services]%
```

3. Add and configure the `cerebrod` service.

```
[esms1->device[esfs-mds1]->services]% add cerebrod
[esms1->device*[esfs-mds1*]->services*[cerebrod*]]% show
Parameter                      Value
------------------------------ --------------------------
Autostart                      no
Belongs to role                no
Monitored                      no
Revision
Run if                         ALWAYS
Service                        cerebrod

[esms1->device*[esfs-mds1*]->services*[cerebrod*]]% set monitored on
[esms1->device*[esfs-mds1*]->services*[cerebrod*]]% set autostart on
[esms1->device*[esfs-mds1*]->services*[cerebrod*]]% commit
[esms1->device[esfs-mds1]->services[cerebrod]]%
```

4. To monitor the status of `cerebrod`, enter:

```
[esms1->device[esfs-mds1]->services[cerebrod]]% status
cerebrod                  [DOWN]
```

Procedure 92. Manage Cerebro for a category

1. Log in to the CIMS node as `root` and start `cmsh`.

```
esms1# cmsh
[esms1]%
```

2. Switch to `category services` mode and select the `esFS-MDS` category.

```
[esms1]% category services esFS-MDS
[esms1->category[esFS-MDS]->services]%
```

3. Configure the `cerebrod` services for the `esFS-MDS` category.

```
[esms1->category[esFS-MDS]->services]% add cerebrod
[esms1->category*[esFS-MDS*]->services*]% set autostart yes
[esms1->category*[esFS-MDS*]->services*]% set monitored yes
[esms1->category*[esFS-MDS*]->services*]% commit
esms1->category[esFS-MDS]->services]%
```

Refer to the *Bright Cluster Manager 6.1 Administrator Manual* PDF file on the CIMS node for more information about configuring services in Bright.

6.6 Mount a Lustre® File System on a CLE System

Procedure 93. Mount a Lustre file system on a CLE system

1. Add a `node_class` entry for LNET router nodes to `CLEinstall.conf`. The `node_class` entry will be added to `/etc/hosts` when `CLEinstall` is run. In this example, the LNET routers node IDs (NIDs) are `2` and `30`.

   a. Log in to the SMW as `root`.

   b. Edit the `/home/crayadm/install.`*xtrel*`/CLEinstall.conf` file so

that there will be an `lnet` class created with `nid 2` and `30` as members of that class. After running `CLEinstall`, the new class and its members will be in `/etc/opt/cray/sdb/node_classes` on the `bootroot` and `sharedroot` file systems.

```
smw#  vi /home/crayadm/install.xtrel/CLEinstall.conf

node_class[1]=lnet 2 30
```

c. Save the file and exit.

2. Run `CLEinstall` using the same command line options as if you were performing a software update. Running `CLEinstall` makes the configuration change to the `node_classes` file. After running `CLEinstall`, boot the CLE system before continuing to step 3.

3. Make a mount point in the default view of the `sharedroot`.

```
smw# ssh boot
boot# xtopview
default/:/ # mkdir -p /lus/scratch
```

4. Add the following parameter options to `/etc/modprobe.conf.local` for login nodes and all service nodes. Note that the line containing `options lnet networks=gni` is commented. The setting of `10.149.1.*` is the site IB network address, but the network must be within 10.149.0.0/16.

```
default:/ # vi /etc/modprobe.conf.local

#options lnet networks=gni
options lnet ip2nets="gni0 10.128.*.*; o2ib 10.149.*.*"
options lnet routes="gni0 10.149.1.[3,31]@o2ib; o2ib [2,30]@gni0"

### LNET options
options lnet check_routers_before_use=1
options lnet avoid_asym_router_failure=1
options lnet dead_router_check_interval=60
options lnet live_router_check_interval=60
options lnet router_ping_timeout=50
```

5. (Optional) Cray recommends the following parameters be set for CLFS nodes in `modprobe.conf.local`.

```
## o2iblnd parameters
# Increase
options ko2iblnd timeout=100

# Disable peer health
options ko2iblnd peer_timeout=0

# Decrease
options ko2iblnd keepalive=30

# Increase
options ko2iblnd credits=2048

# Increase
options ko2iblnd ntx=2048

# Increase
options ko2iblnd peer_credits=126

# Increase
options ko2iblnd concurrent_sends=63
```

6. Exit `xtopview`.

```
default:/ # exit
```

7. Create the `lnet` class-specialized files.

```
boot-p1# xtopview -c lnet
class/lnet:/ # ls -l /etc/modprobe.conf.local
lrwxrwxrwx 1 root root 45 Feb  1 08:53 /etc/modprobe.conf.local -> /.shared/base/default/etc/\
modprobe.conf.local
class/lnet:/ # xtspec -c lnet /etc/modprobe.conf.local
class/lnet:/ # ls -l /etc/modprobe.conf.local
lrwxrwxrwx 1 root root 48 Feb  1 09:06 /etc/modprobe.conf.local -> /.shared/base/class/lnet/\
etc/modprobe.conf.local
class/lnet:/ # vi  /etc/modprobe.conf.local
```

Add the local extensions to the `/etc/modprobe.conf.local` file. The two examples below assume that `ib0` on the LNET router nodes is connected to the DMP IB switch. If `ib1` is used, there is a different format.

For `ib0`:

```
### LNET interfaces
options lnet ip2nets="gni0 10.128.*.*; o2ib 10.149.*.*"
### LNET routes for esFS
options lnet routes="gni0 10.149.1.[3,31]@o2ib; o2ib [2,30]@gni0"
```

For `ib1`:

```
### LNET interfaces
options lnet ip2nets="gni0 10.128.*.*; o2ib(ib1) 10.149.*.*"
### LNET routes for esFS
options lnet routes="gni0 10.149.1.[3,31]@o2ib; o2ib [2,30]@gni0"
```

8. Create node-specialized files for all LNET routers. Repeat steps step 8.a through step 8.d for each LNET router.

   a. Run `xtopview` to get the node view of the `sharedroot` for NID `2`.

```
boot# xtopview -n 2
```

   b. Create and edit the interface file (example shows `ifcfg-ib0`) for NID `2` with IP address 10.149.1.3. This IP address is the IP address of the `esfs-mds001` node on the `ib-net`.

```
node/2:/ # touch /etc/sysconfig/network/ifcfg-ib0
node/2:/ # xtspec -n 2 /etc/sysconfig/network/ifcfg-ib0
node/2:/ # vi /etc/sysconfig/network/ifcfg-ib0
```

   An example `ifcfg-ib0` file follows:

```
BOOTPROTO='static'
IPADDR=10.149.1.3
NETMASK=255.255.0.0
STARTMODE='onboot'
USERCONTROL='no'
MTU=2044
IPOIB_MODE='connected'
```

   c. Create and edit the `/etc/sysconfig/infiniband` file.

```
node/2:/ # touch /etc/sysconfig/infiniband
node/2:/ # xtspec -n 2 /etc/sysconfig/infiniband
node/2:/ # vi /etc/sysconfig/infiniband
```

   Change the following two variables:

```
ONBOOT=no
SRP_LOAD=yes
```

   To

```
ONBOOT=yes
SRP_LOAD=no
```

   d. Exit `xtopview`.

```
node/2:/ # exit
boot#
```

9. Update the bootimage so that compute nodes mount the Lustre file system.

   a. From the SMW, update the bootimage `/etc/modprobe.conf`. This example uses the `p1` partition of a CLE system.

```
smw# cd /opt/xt-images/templates/default-p1
smw:/opt/xt-images/templates/default-p1 # vi etc/modprobe.conf
```

Make the following changes to the modprobe.conf file.

```
#options lnet networks=gni
### LNET options
options lnet check_routers_before_use=1
options lnet avoid_asym_router_failure=1
options lnet dead_router_check_interval=60
options lnet live_router_check_interval=60
options lnet router_ping_timeout=50

### LNET interfaces
options lnet ip2nets="gni0 10.128.*.*; o2ib 10.149.*.*"

### LNET routes for esFS
options lnet routes="gni0 10.149.1.[3,31]@o2ib; o2ib [2,30]@gni0"
```

b. Make a mount point for the bootimage.

```
smw:/opt/xt-images/templates/default-p1 # mkdir -p lus/scratch
```

c. Update /etc/fstab for bootimage.

```
smw:/opt/xt-images/templates/default-p1 # cd etc
smw:/opt/xt-images/templates/default-p1/etc # cp -p fstab fstab.orig
smw:/opt/xt-images/templates/default-p1/etc # vi fstab
```

Add the following line to /etc/fstab.

```
10.149.0.2@o2ib:10.149.0.3@o2ib:/scratch  /lus/scratch lustre rw,flock 0 0
```

Exit and save the /etc/fstab file.

d. Rebuild the bootimage. This example uses a script created for the BLUE system set.

```
smw# /var/opt/cray/install/shell_bootimage_BLUE.sh -c
```

10. This step requires that /etc/modprobe.d/cray.conf file exist in the CLFS node software image. You may need to create (touch) the /etc/modprobe.d/cray.conf file to create it in the CLFS node software image before proceding.

    a. Log in to the CIMS node as root.

    b. Use the chroot shell to modify the CLFS node software image (/cm/images/ESF-XX-2.2.0-201401151643 for example).

    ```
    esms1# cd /cm/images
    esms1# chroot ESF-XX-2.2.0-201401151643
    ```

c. Determine if the `cray.conf` file exists in `/etc/modprobe.d`.

```
[root@esms1 /]# cd /etc/modprobe.d
[root@esms1 /]# ls
blacklist-kvm.conf    cray-ib_srp.conf  cray-ost.conf   ib_ipoib.conf
blacklist.conf        cray-lnd.conf     dist-alsa.conf  ib_sdp.conf
bright-cmdaemon.conf  cray-lnet.conf    dist-oss.conf   mlx4_en.conf
cray-aliases.conf     cray-mlx4.conf    dist.conf       openfwwf.conf
[root@esms1 /]# touch cray.conf
[root@esms1 /]# ls
blacklist-kvm.conf    cray-lnd.conf   dist-alsa.conf  mlx4_en.conf
blacklist.conf        cray-lnet.conf  dist-oss.conf   openfwwf.conf
bright-cmdaemon.conf  cray-mlx4.conf  dist.conf
cray-aliases.conf     cray-ost.conf   ib_ipoib.conf
cray-ib_srp.conf      cray.conf       ib_sdp.conf
[root@esms1 /]# exit
```

11. Update the CLFS category finalize script for `esfs-odd-`*filesystem*, `esfs-even-`*filesystem*, and `esfs-failed-`*filesystem* categories.

    a. Edit the `site.esf_finalize.sh` script.

```
esms1# cd /opt/cray/esms/cray-es-finalize-scripts-XX/etc
esms1# vi site.esf_finalize.sh
```

    b. Add the entry before the `exit 0` line. This example uses the IP address 10.149.1.3 for NID 2 and 10.149.1.31 for NID 30.

```
echo "options lnet routes=\"gni0 10.149.1.[3,31]@o2ib\" >> /localdisk/etc/modprobe.d/cray.conf
```

    c. Exit and save the `site.esf_finalize.sh` script.

    d. Update each of the CLFS categories in Bright with the new finalize script (`esfs-odd-`*filesystem*, `esfs-even-`*filesystem*, and `esfs-failed-`*filesystem*.

```
esms1# cmsh
[esms1]% category
[esms1->category]% use esfs-even-filesystem
[esms1->category[esfs-even-filesystem]]% set finalizescript /opt/cray/esms/cray-es-finalize-scripts-XX/\
etc/site.esf_finalize.sh
[esms1->category*[esfs-even-filesystem*]]% commit
```

    e. Verify the finalize script.

```
[esms1->category[esfs-even-filesystem]]% get finalizescript
```

    f. Quit `cmsh`.

```
[esms1->category[esfs-even-filesystem]]% quit
```

12. Unmount the Lustre clients and stop the Lustre file system.

    a. Unmount CDL Lustre clients.

```
eslogin1# umount /lus/scratch
```

b. Stop Lustre file system.

```
esms1# lustre_control status -f scratch
esms1# lustre_control stop -f scratch
```

13. Reboot MDS and OSS node(s). Wait for each all MDS and OSS nodes to reboot before continuing. Use Bright `cmgui` to open remote consoles for each node to monitor to reboot process.

```
esms1# cmsh
esms1# device
esms1# foreach esfs-mds001,esfs-mds002,esfs-oss1,esfs-oss002,esfs-oss003,esfs-oss004 (reboot)
esfs-mds001: Reboot in progress ...
esfs-mds002: Reboot in progress ...
```

14. Exit `cmsh` and start the Lustre file system.

```
[esms1]% exit
esms1# lustre_control start -f scratch
Performing 'start' from esms1 at Wed Apr 24 16:06:55 CDT 2013
```

15. Either configure LDAP on MDS nodes or execute the following command on the `esfs-mds001` node after starting Lustre file system to disable upcall and make the MGS/MDS use the UID/GID supplied by the client. Refer to Configure LDAP on MDS Nodes on page 237 for more information about configuring LDAP after completing this procedure.

```
esms1# ssh esfs-mds001
esfs-mds001# lctl conf_param extlus-MDT0000.mdt.identity_upcall=NONE
```

16. Log out of the `esfs-mds001` node and return to the CIMS node.

```
esf1-mds001# exit
esms1#
```

17. Use `ssh` to log in to `eslogin1` and mount the Lustre client.

```
esms1# ssh eslogin1
eslogin1# mount -t lustre -o rw,flock,lazystatfs 10.149.0.2@o2ib:10.149.0.3@o2ib:/scratch
/lus/scratch
```

18. Reboot the CLE system. The compute nodes will mount the Lustre file system as specified in the boot image `/etc/fstab` file.

19. Mount the Lustre file system on the CLE login node. See *Managing Lustre for the Cray Linux Environment (CLE)*, (S–0010).

6.7 Configure a CDL Node to Mount an External Lustre® File System

This section describes how to configure a CDL node to mount a Lustre file system.

Procedure 94. Configure a CDL node to mount an external Lustre file system

Note: This procedure modifies the software image *imagename*, to mount a Lustre file system which is mounted as `/scratch` on the Lustre MDS server at IP address 10.149.0.1 to the local mount point on the CDL of `/lus/scratch`.

1. Make a mount point in the CDL software image.

```
esms1# chroot /cm/images/imagename
esms1:/> mkdir -p /lus/scratch
esms1:/> exit
```

2. Use the `cmsh imageupdate` command to update what is on the running node from *imagename*. The command performs a dry run, then use `-w` to perform the actual update.

```
esms1# cmsh
[esms1]% device use eslogin1
[esms1->device[eslogin1]]% imageupdate
Performing dry run (use synclog command to review result, then pass -w to perform real update)...
[esms1->device[eslogin1]]% imageupdate -w
Mon Mar  4 14:08:26 2013 [notice] esms1: Provisioning started: sending\
esms1:/cm/images/imagename to eslogin1:/, mode UPDATE, dry run = no
```

3. Start a remote console on `eslogin1`.

```
[esms1->device[eslogin1]]% rconsole
```

4. Start a new `cmsh` session and reboot `eslogin1` to add the new mount point.

```
esms1# cmsh
[esms1]% device use eslogin1
[esms1->device[eslogin1]]% reboot
```

5. After `eslogin1` reboots, login as `root` and enter the following commands to mount the file system:

```
esms1# ssh root@eslogin1
eslogin1# mount -t lustre -o rw,flock,lazystatfs 10.149.0.1@o2ib:/scratch /lus/scratch
```

6. Verify file system mounted.

```
eslogin1# mount | grep lustre
10.149.0.1@o2ib:/scratch on /lus/scratch type lustre (rw,flock,lazystatfs)

eslogin1# df /lus/scratch
Filesystem              1K-blocks      Used   Available     Use% Mounted on
10.149.0.1@o2ib:/scratch 15611116960    922104 14829205576   1%   /lus/scratch
```

6.8 Mount a Lustre® File System on a CDL Node

Procedure 95. Mount a Lustre file system on a CDL node

1. Log in to the CIMS node as `root`.

2. Use the `chroot` shell to make a mount point in the CDL software image (in this example `ESL-XC-2.2.0-201401160637`).

```
esms1# chroot /cm/images/ESL-XC-2.2.0-201401160637
esms1:/> mkdir -p /lus/scratch
esms1:/> exit
```

3. Update the CDL nodes with the modified `ESL-XC-2.2.0-201401160637` image.

   a. Start `cmsh` and switch to `device` mode.

```
esms1# cmsh
[esms1]% device
[esms1->device]%
```

   b. Update the image for the CDL node (in this example, `eslogin1`).

```
[esms1->device]% imageupdate -n eslogin1
Performing dry run (use synclog command to review result, then pass -w to perform real update)...
[esms1->device]%
...
[esms1->device]% imageupdate -w -n eslogin1
```

   c. Exit `cmsh`.

```
[esms1->device]% quit
```

4. Use SSH to log in to the CDL node (`eslogin1`).

```
esms1# ssh eslogin1
Last login: Fri Apr 15 11:28:06 2013 from esms1.cm.cluster
eslogin1#
```

5. Mount the file system on the CDL node (`eslogin1`). Add the following line to `/etc/fstab`, where 10.149.0.2 is the IP address of the `esfs-mds001` node on `ib-net`. The `scratch.fs_defs` also defines a failover MDS, so both the primary and failover MDS are included in the `fstab` entry:

```
10.149.0.2@o2ib:10.149.0.3@o2ib:/scratch  /lus/scratch lustre rw,flock,lazystatfs 0 0
```

```
eslogin1# mount -t lustre -o rw,flock,lazystatfs 10.149.0.2@o2ib:/scratch /lus/scratch
```

6. Verify that the `scratch` file system is mounted.

```
eslogin1# mount | grep lustre
10.149.0.1@o2ib:/scratch on /lus/scratch type lustre (rw,flock,lazystatfs)

eslogin1# df /lus/scratch
Filesystem              1K-blocks      Used   Available   Use% Mounted on
10.149.0.1@o2ib:/scratch 15611116960    922104 14829205576   1% /lus/scratch
```

Monitoring and Troubleshooting [7]

Bright Cluster Manager® (Bright) software enables administrators to monitor health check information and metrics. Health checks run periodically from the cluster management daemon (CMDaemon or `cmd`) and may run on either the CIMS or slave node or both. Health checks return a `PASS`, `FAIL` or `UNKNOWN` condition, and actions can be taken based on the return value. Metrics also run periodically from CMDaemon (`cmd`) and may run on either the CIMS, slave node, or both. Metrics return a numeric value, and actions can be taken based on crossing a threshold value.

Refer to Failover Features and Bright Monitoring on page 205 for information about how to install `esfsmon_` and `esfsmon_action` to monitor failover conditions.

Refer to the *Bright Cluster Manager 6.1 Administrator Manual* PDF file on the CIMS node for more information about Cray Data Management Platform (DMP) monitoring capabilities.

The monitoring system:

- Inspects monitoring data at preset levels to preserve resources
- Configures and gathers monitoring data for new resources
- Identifies current and past problems or abnormal behavior
- Analyzes trends that help an administrator predict likely future problems
- Handles problems by triggering alerts
- Taking action, if necessary, to improve the situation or to investigate further

Bright uses the term *Metric* to describe a property of a device that can be monitored and has a numeric value. Examples are 45.2 °C, any number like 1.23, or a value in bytes such as 12322343. Thresholds can be set so that when a the threshold is crossed (ion either direction), *Actions* are taken. *Actions* are stand-alone shell scripts that run when a monitoring condition is met.

Health checks are device states, that are returned when a health check script is periodically run on a node. Return values for the example are, `PASS`, `FAIL`, or `UNKNOWN`. An example of the health check feature follows:

- Determine if a hard drive has enough space available and returning a `PASS` status if it does
- Determine if an NFS® mount is accessible, and returning `FAIL` if it is not

- Determine if `CPUUser` is below 50%, and returning `PASS` if it is
- Determine if the `cmsh` binary is found, and returning `UNKNOWN` if it is not

Refer to Table 7 for alter level descriptions.

The `cmsh` monitoring mode is separated into 4 sections: `actions`, `healthchecks`, `metrics`, and `setup`. The `monitoring actions` mode enables control over the actions you have defined in shell scripts, or the built in actions such as `power off`.

Example 26. `cmsh` `monitoring` `actions` mode

```
esms1# cmsh
[esms1]% monitoring actions
[esms1->monitoring->actions]% list
Name (key) Command
---------------------- ------------------------------------------------
Drain node             <built in>
Power off               <built in>
Power on               <built in>
Power reset             <built in>
Reboot                  <built in>
SendEmail               <built in>
Shutdown                <built in>
Undrain node            <built in>
killprocess             /cm/local/apps/cmd/scripts/actions/killprocess.pl
remount                 /cm/local/apps/cmd/scripts/actions/remount
testaction              /cm/local/apps/cmd/scripts/actions/testaction
```

Example 27. `cmsh` `monitoring` `healthchecks` mode

```
esms1# cmsh
[esms1->monitoring->healthchecks]% list
name (key)         command
------------------ ------------------------------------------------------
DeviceIsUp         <built-in>
ManagedServicesOk  <built-in>
chrootprocess      /cm/local/apps/cmd/scripts/healthchecks/chrootprocess
cmsh               /cm/local/apps/cmd/scripts/healthchecks/cmsh
diskspace          /cm/local/apps/cmd/scripts/healthchecks/diskspace
exports            /cm/local/apps/cmd/scripts/healthchecks/exports
failedprejob       /cm/local/apps/cmd/scripts/healthchecks/failedprejob
failover           /cm/local/apps/cmd/scripts/healthchecks/failover
hardware-profile   /cm/local/apps/cmd/scripts/healthchecks/node-hardware-+
hpraid             /cm/local/apps/cmd/scripts/healthchecks/hpraid
interfaces         /cm/local/apps/cmd/scripts/healthchecks/interfaces
ipmihealth         /cm/local/apps/cmd/scripts/metrics/sample_ipmi
ldap               /cm/local/apps/cmd/scripts/healthchecks/ldap
lustre             /cm/local/apps/cmd/scripts/healthchecks/lustre
mounts             /cm/local/apps/cmd/scripts/healthchecks/mounts
mysql              /cm/local/apps/cmd/scripts/healthchecks/mysql
ntp                /cm/local/apps/cmd/scripts/healthchecks/ntp
oomkiller          /cm/local/apps/cmd/scripts/healthchecks/oomkiller
portchecker        /cm/local/apps/cmd/scripts/healthchecks/portchecker
rogueprocess       /cm/local/apps/cmd/scripts/healthchecks/rogueprocess
schedulers         /cm/local/apps/cmd/scripts/healthchecks/schedulers
smart              /cm/local/apps/cmd/scripts/healthchecks/smart
ssh2node           /cm/local/apps/cmd/scripts/healthchecks/ssh2node
swraid             /cm/local/apps/cmd/scripts/healthchecks/swraid
testhealthcheck    /cm/local/apps/cmd/scripts/healthchecks/testhealthcheck
```

Refer to the *Bright Cluster Manager 6.1 Administrator Manual* PDF file on the CIMS node for more information about each metric and descriptions for each.

Example 28. `cmsh` `monitoring` `metrics` mode

```
esms1# cmsh
[esms1->monitoring]% metrics
[esms1->monitoring->metrics]% list
Name (key)                  Command
--------------------------- ------------------------------------------------
AlertLevel                  <bulit-in>
Ambient_Temp                /cm/local/apps/cmd/scripts/metrics/sample_ipmi
.
.
.
```

7.1 Check Device Status

Use the following cmsh commands to check node and other device status in Bright.

```
esms1# cmsh
[esms1]% device status
KVM ................. [ UP ]
esms ................ [ UP ]
eth-01 .............. [ UP ]
ib-01 ............... [ UP ]
ib-02 ............... [ UP ]
mds01 ............... [ UP ]
mds02 ............... [ UP ]
oss01 ............... [ UP ]
oss02 ............... [ UP ]
oss03 ............... [ UP ]
oss04 ............... [ UP ]
[esms]%
```

7.2 Check Power Status

Use the following cmsh commands to check node power status, and the power status of other devices in Bright.

```
esms# cmsh
[esms]% device power status
No power control .... [ UNKNOWN ] KVM
No power control .... [ UNKNOWN ] esms
No power control .... [ UNKNOWN ] eth-01
No power control .... [ UNKNOWN ] ib-01
No power control .... [ UNKNOWN ] ib-02
ipmi0 ............... [ ON ] mds01
ipmi0 ............... [ ON ] mds02
ipmi0 ............... [ ON ] oss01
ipmi0 ............... [ ON ] oss02
ipmi0 ............... [ ON ] oss03
ipmi0 ............... [ ON ] oss04
[esms]%
```

7.3 Check Node Health Status

Use the following cmsh commands to check node health status, and the health status of other devices in Bright.

```
esms1# cmsh
[esms1]% device
[esms1->device]% showhealth
Device      AlertLevel Failed               Thresholds    Unknown
------------------ ---------- ------------------- -------------- ---------------
esmaint-net-switch  0
esms1               30         ManagedServicesOk
ib-switch-1         0
ipmi-net-switch     0
lake-esl            0
mds001              10         mounts, rogueprocess
```

```
mds002              40         DeviceIsUp
node001             no data
oss001              10         mounts, rogueprocess
oss002              10         mounts, rogueprocess
eslogin1            0
```

The `AltertLevel` entries are defined in Table 7:

Table 7. Health Check Alert Levels

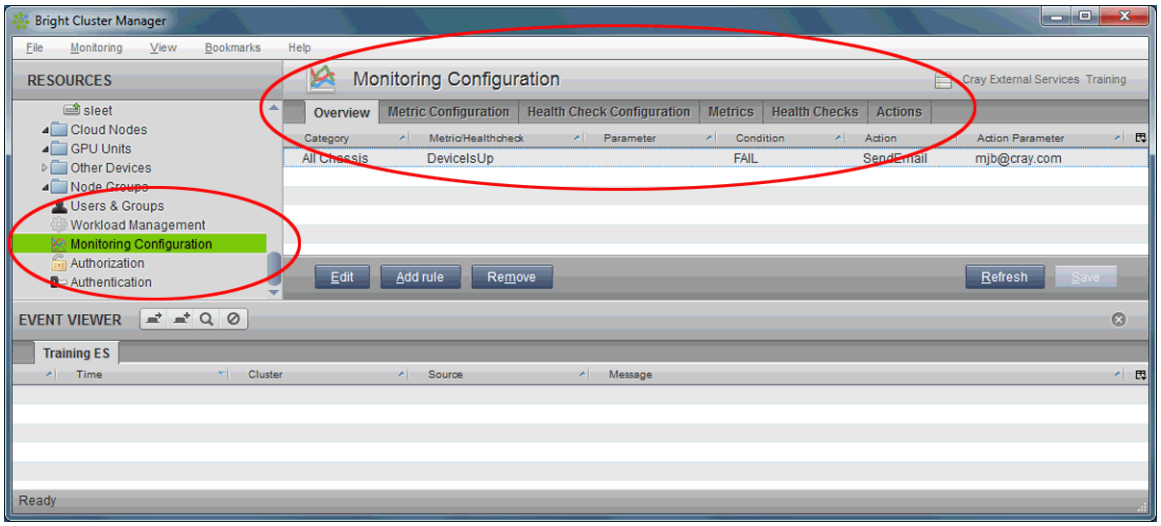
Value	Name	Description
0	Info	Informational Message
10	Notice	Normal, but significant condition
20	Warning	Warning conditions
30	Error	Error conditions
40	alert	take immediate action

```
esms1# cmsh
[esms1]% device
[esms1->device]% latesthealthdata mds001
Health Check           Severity   Value       Age (sec.) Info Message
---------------------- --------   -------     ---------  ------------------------
DeviceIsUp              0          PASS        45
ManagedServicesOk       0          PASS        165
mounts                  10         FAIL        645       defined mountpoint /proc has different+
rogueprocess            10         FAIL        645       /usr/sbin/sendmail.sendmail (smmsp) /*6+
ssh2node                0          PASS        645
[esms1->device]%
```

7.4 Monitoring Configuration with `cmgui`

The monitoring configuration for health checks, metrics, and triggered actions is configured in the **Monitoring Configuration** section under resources section of `cmgui`.

Figure 39. Bright Monitoring Configuration



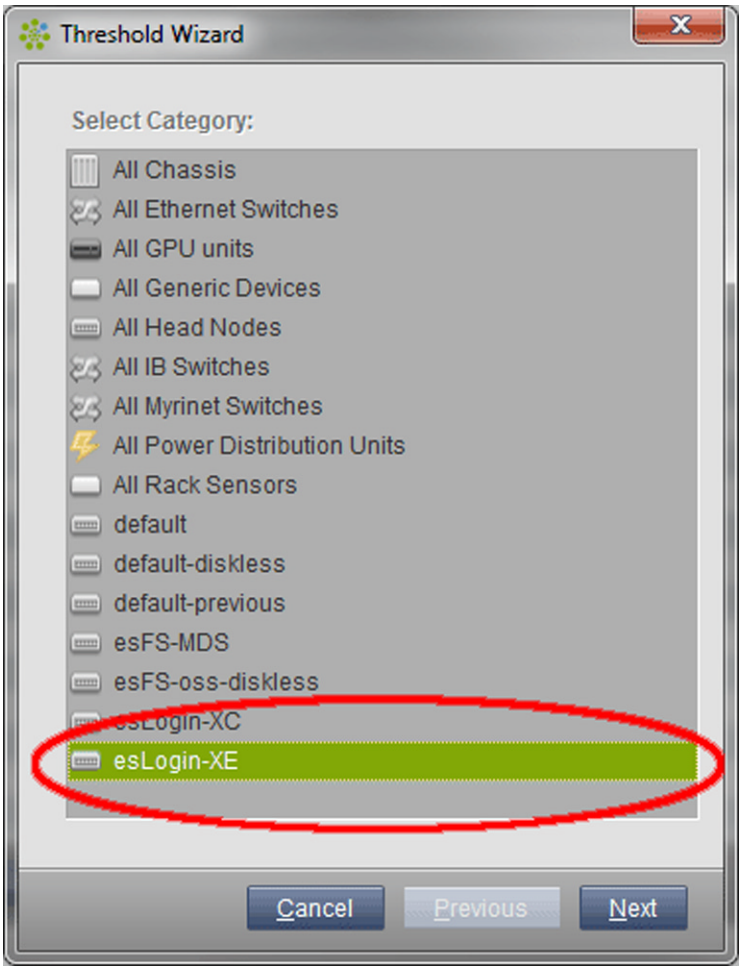
The Monitoring option in the menu bar of `cmgui` starts a visualization tool that enables you to monitor system behavior over periods of time. The monitoring framework enables you to monitor a condition, add an *Action* (an executable shell script), and then set up a threshold level that triggers the action.

Procedure 96. Monitoring configuration setup

This procedures configures a monitoring rule that alerts the administrator if an CDL node goes down for any reason.

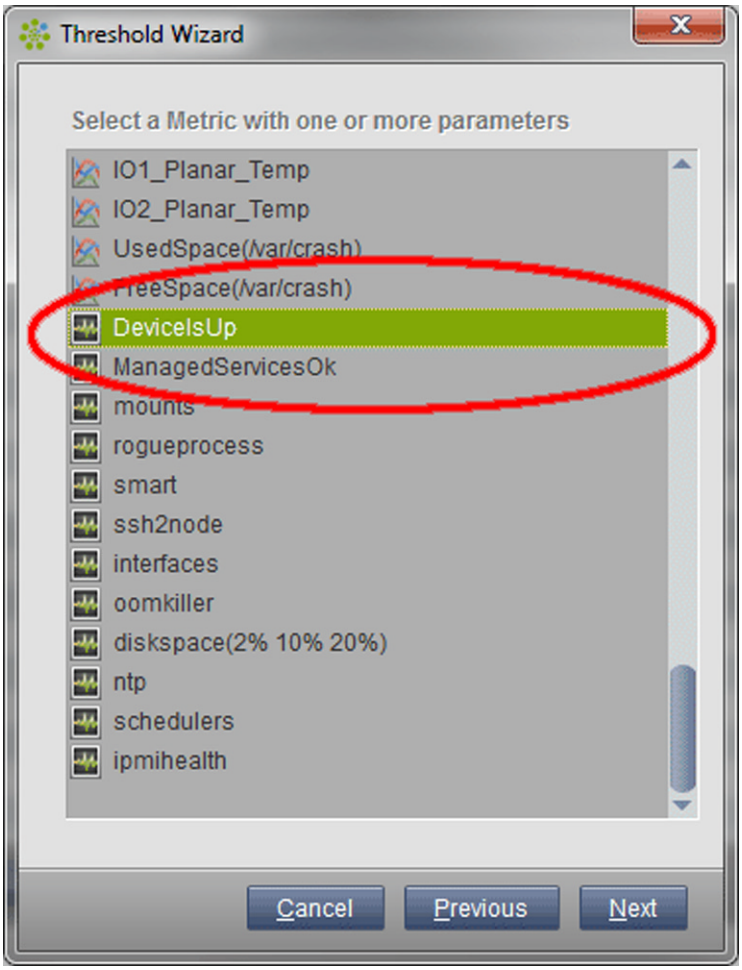
1. In `cmgui`, select **Monitoring Configuration** from the resource tree (refer to Figure 39.)
2. From the **Threshold Wizard**, select a device category (this example selects the `esLogin-XE` category).

Figure 40. Monitoring Configuration Category



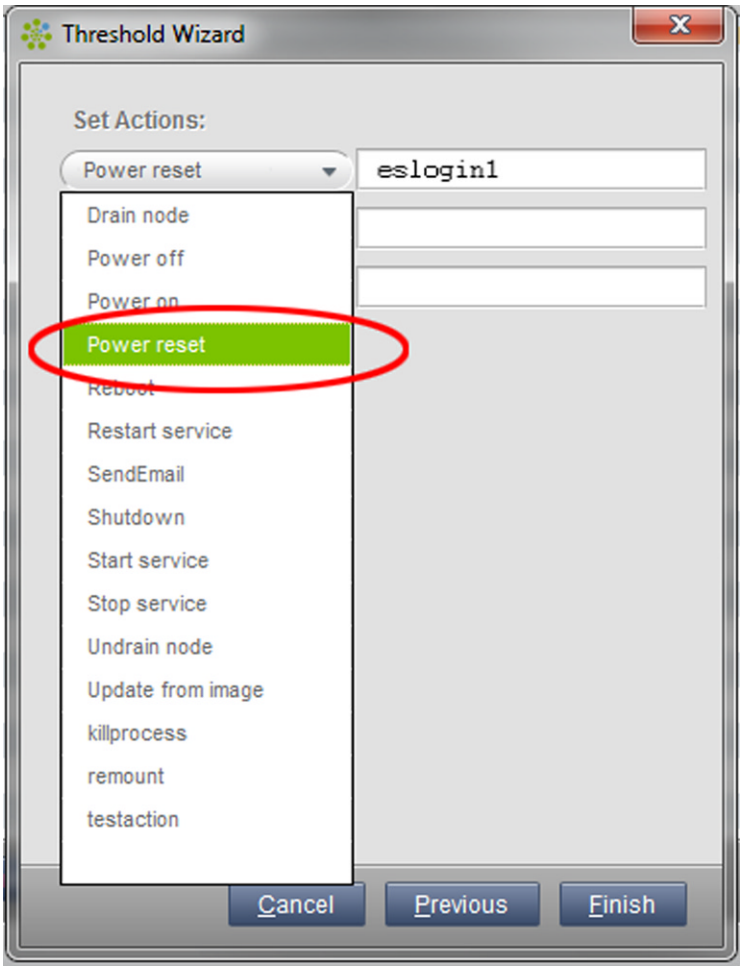
3. Click **Next**, then scroll down and locate the **DeviceIsUp** health check metric from the menu, and click **Next** again.

Figure 41. Monitoring Configuration Metric



4. In the **Set Actions** pulldown menu, select **Power Reset**, and enter the name of the slave node to reset (this example shows `eslogin1`). You can also include a **SendEmail** action, and enter an e-mail address for the administrator.

Figure 42. Monitoring Configuration Action



5. Click **Finish** to save the monitoring rule.

7.5 Check System Status

The CM software enables you to display power and device status. The following states indicate a normal operational status:

- `CLOSED` — The node is not being monitored by Bright Cluster Manager (Bright).
- `OPEN` — The node is being monitored by the Bright and is not in one of the following states.
- `DOWN` — The operating system is down and/or the CIMS node cannot communicate with CMDaemon (`cmd`) running on the node. This can be an expected state if it is intentionally entered by the administrator. If it in not intentionally in a `DOWN` state, expect a failure.

Under high loads during normal operation, the CLFS node state may toggle between UP and DOWN within 2-3 second intervals. This is not a failure, but indicates that the node was busy handling file system traffic during the time the CIMS node was requesting its state.

- INSTALLING — The Bright node-installer is provisioning the node during the boot process.
- INSTALLER_CALLINGINIT — The CM node-installer has handed over control to the local init process.
- UP — The OS is up and the CIMS node can manage the node.

The following states indicate a problem has occurred during the boot process:

- INSTALLER_FAILED — The Bright node-installer has detected an unrecoverable problem during the boot process or has taken too long to enter the UP state. Possible reasons for this state include:
  - Local hard disk not found.
  - Failure to start a network interface.
  - Previous state was INSTALLER_REBOOTING and the reboot took too long. Possible reasons for failure to reach the UP state in time include:
    - Failure to hand over control from the Bright node-installer to the local init process.
    - The local init process failed to start CMDaemon (cmd) or the cmd took too long to start if the latter, this state will go to UP when cmd starts.
- INSTALLER_UNREACHABLE — The CIMS node CMDaemon (cmd) can no longer ping the node. The node may have crashed while running the CM node-installer.
- INSTALLER_REBOOTING — The Bright node-installer may need to reboot a node to install a new kernel.
- INSTALLER_FAILED — The Bright node-installer failed or it took too long to long to enter the UP state.

Use the examples in the following procedures to check status of individual nodes or a group of nodes. You can check the power status of nodes individually, by list, range, category, or node group as shown in Procedure 97.

Procedure 97. Check power status

1. Log in to the CIMS node as root and run cmsh.

```
esms1# cmsh
[esms1]%
```

2. Type **device** at the cmsh prompt to enter device mode.

```
[esms1]% device
[esms1->device]%
```

a. To check power status of an individual node such as, eslogin01, type:

```
[esms1->device]% power -n eslogin01 status
```

b. To check power status of a list of nodes, such as eslogin01 to eslogin04 plus eslogin06, type:

```
[esms1->device]% power -n eslogin01..eslogin04,eslogin06 status
```

c. To check power status of all nodes in the eslogin category, type:

```
[esms1->device]% power -c eslogin status
```

d. To check power status of all nodes in the dm node group, type:

```
[esms1->device]% power -g dm status
```

3. You can check the device status of nodes individually, by list and range, by category or node group.

a. To check device status of an individual node such as, eslogin01, type:

```
[esms1->device]% status -n eslogin01
```

b. To check device status of a list of nodes, such as eslogin01 to eslogin04 plus eslogin06, type:

```
[esms1->device]% status -n eslogin01..eslogin04,eslogin06
```

c. To check device status of all nodes in the eslogin-XE category, type:

```
[esms1->device]% status -c eslogin-XE
```

d. To check device status of all nodes in the datamover node group, type:

```
[esms1->device]% status -g datamover
```

7.6 Monitoring Health and Metrics

- Refer to Switches and PDUs on page 39 for information about monitoring switches, RAID controllers, and other devices using Bright.
- Refer to Configure the LSI® MegaCLI™ RAID Utility on page 91 for information about monitoring CIMS node, CDL, or CLFS local RAID systems.
- Refer to Configure CLFS Failover (esfsmon 2.0.0) on page 203 for information about configuring esfsmon to monitor file system failover.
- Set E-mail Alerts when a Node Goes Down on page 292 describes how to configure a Bright health check to send an E-mail when a node goes down.

Procedure 98. Monitor system health and metrics

1. Log in to the CIMS node as `root` and start `cmsh`.

```
esms1# cmsh
```

2. Type **`device`** at the `cmsh` prompt to enter device mode and display the device mode prompt:

```
[esms1]% device
[esms1->device]%
```

3. Type **`latesthealthdata`** ***device*** **`-v`** to monitor system health and metrics:

```
[esms1->device]% latesthealthdata esfs-oss001 -v
Health Check                Severity Value           Age (sec.) Info Message
--------------------------- -------- --------------- ---------- ---------------------------------------
DeviceIsUp                  0        PASS            111
ManagedServicesOk           0        PASS            111
mounts                      0        PASS            1551
rogueprocess                0        PASS            351
smart                       0        PASS            1551       sda: Smart command failed
                                                                sdb: Smart command failed
                                                                sdc: Smart command failed
                                                                sdd: Smart command failed
                                                                sde: Smart command failed
                                                                sdf: Smart command failed
                                                                sdg: Smart command failed
                                                                sdh: Smart command failed
ssh2node                    0        PASS            1551
interfaces                  0        PASS            1551
oomkiller                   0        PASS            1551
diskspace:2% 10% 20%        0        PASS            1551
ntp                         0        PASS            351
schedulers                  0        PASS            1551
ipmihealth                  0        PASS            111
```

Procedure 99. Display the metric status

1. Log in to the CIMS node as `root` and start `cmsh`.

```
esms1# cmsh
```

2. Type **`device`** at the `cmsh` prompt to enter device mode and display the device mode prompt:

```
[esms1]% device
[esms1->device]%
```

3. Type **latestmetricdata** ***device*** **-v**:

```
 [esms1->device]% latestmetricdata esfs-oss001 -v
Metric                       Value           Age (sec.) Info Message
---------------------------- --------------- ---------- --------------------------------------
AlertLevel:max               0               106
AlertLevel:sum               0               106
BytesRecv:BOOTIF             332.35          166
BytesRecv:eth1               0               166
BytesRecv:eth2               0               166
BytesRecv:eth3               0               166
BytesRecv:ib0                8.4             166
BytesRecv:ib1                0               166
. . .
```

Procedure 100 shows you how to dump the health check data for a node. You can dump the collected health check data for a node for a specified period of time. Most health checks are logging the past 3000 samples. Available health checks for a node are given by the `latesthealthdata` command described above.

Procedure 100. Dump health check data

1. Log in to the CIMS node as `root` and start `cmsh`.

```
esms1# cmsh
```

2. Type **device** at the `cmsh` prompt to enter device mode and display the device mode prompt:

```
 [esms1]% device
[esms1->device]%
```

3. Enter **dumphealthdata** ***start-time end-time healthcheck device***. The following example dumps the `deviceisup` data for the past hour:

```
[esms1->device]% dumphealthdata -1h now deviceisup esfs-oss001
# From Wed Sep 11 13:32:10 2013 to Wed Sep 11 14:32:10 2013
Time                         Value           Info Message
---------------------------- --------------- ----------------------------------------
Wed Sep 11 13:32:10 2013   PASS
Wed Sep 11 14:32:00 2013   PASS
```

Procedure 101 shows you how to dump the metric data for a node. You can dump the collected metric data for a node for a specified period of time. Most metrics are logging the past 3000 samples. Available metrics for a node are given by the `latestmetricdata` command described above.

Procedure 101. Dump metric data

1. Log in to the CIMS node as `root` and start `cmsh`.

```
esms1# cmsh
```

2. Enter **device** at the `cmsh` prompt to enter device mode and display the device mode prompt:

```
[esms1]% device
[esms1->device]%
```

3. Enter **`dumpmetricdata`** ***start-time end-time healthcheck device***. The following example dumps the `deviceisup` data for the past hour:

```
[esms1->device]% dumpmetricdata -1h now FreeSpace:/var/crash eslogin1
# From Wed Sep 11 13:38:58 2013 to Wed Sep 11 14:38:58 2013
Time                        Value            Info Message
--------------------------- ---------------- ----------------------------------------
Wed Sep 11 13:38:58 2013   1.64826e+10
Wed Sep 11 14:38:58 2013   1.64826e+10
```

7.6.1 Set E-mail Alerts when a Node Goes Down

Procedure 102. Set e-mail alerts when a node goes down

1. Log in to the CIMS node as `root` and start `cmsh`.

```
esms1# cmsh
[esms1]%
```

2. Switch to `monitoring` mode.

```
[esms1]% monitoring
[esms1->monitoring]%
```

3. Switch to `setup` mode.

```
[esms1]% setup
[esms1->monitoring->setup]%
```

4. Enter `healthconf` and the node category name for the devices that you want to configure.

```
[esms1]% healthconf esLogin-XC
[esms1->monitoring->setup[esLogin-XC]->healthconf]%
```

5. Enter `list` to display the list of health checks.

```
[esms1->monitoring->setup[esLogin-XC]->healthconf]% list
HealthCheck              HealthCheck Param  Check Interval
------------------------ ------------------ ----------------------
DeviceIsUp                                  120
ManagedServicesOk                           120
diskspace                2% 10% 20%         1800
interfaces                                  1800
ipmihealth                                  120
mounts                                      1800
ntp                                         300
oomkiller                                   1800
rogueprocess                                600
schedulers                                  1800
smart                                       1800
ssh2node                                    1800
```

6. Enter `use deviceisup`.

```
[esms1->monitoring->setup[esLogin-XC]->healthconf]% use deviceisup
[esms1->monitoring->setup[esLogin-XC]->healthconf[DeviceIsUp]]%
```

7. Add the `SendEmail` action to the `DeviceIsUp` heath check.

```
[esms1->monitoring->setup[esLogin-XC]->healthconf[DeviceIsUp]]% set failactions SendEmail
```

8. Enter `commit` to save the changes.

```
[esms1->monitoring->setup*[esLogin-XC*]->healthconf*[DeviceIsUp*]]% commit
[esms1->monitoring->setup[esLogin-XC]->healthconf[DeviceIsUp]]%
```

7.7 Log Files

All `syslog` traffic from the DMP slave nodes is forwarded to `/var/log/messages` on the CIMS node.

In addition to `syslog`, Bright maintains an event log in its database (refer to View the Event Log on page 293.) Ensure that CMDaemon (`cmd`) is running on the CIMS node and on the suspect node if cluster management operations are malfunctioning.

Other log files of interest are:

- `/var/log/messages` — Slave node `syslog` messages.
- `/var/log/cmdaemon` — Bright CMDaemon (`cmd`) messages. Check this log if there are system management problems.
- `/var/log/node-installer` — Node Installer messages. Check this log if there are boot problems.
- `/var/log/conman/` — Slave node console messages.
- `/var/adm/cray/logs` — Software installation logs.
- Bright event log. Stored in the Bright database and accessed using `events` command in `cmsh` or event viewer when using the `cmgui`.

7.8 Bright Logs

7.8.1 View the Event Log

Procedure 103 shows you how to view the event log.

Procedure 103. View the event log

1. Log in to the CIMS node as `root` and start `cmsh`.

```
esms1# cmsh
```

2. Type **events** followed by the number of events to display. Use events details *eventnum* to display details for a specific event.

```
[esms1]% events 50
Thu Aug 29 16:30:43 2013 [notice] esms1: Starting image directory removal: /cm/images/ESL-XE-1.1.1-kdump
Thu Aug 29 16:31:00 2013 [notice] esms1: Check 'chrootprocess' is in state PASS on esms1
Thu Aug 29 16:31:47 2013 [notice] esms1: Image directory removal succeeded for: /cm/images/ESL-XE-1.1.1-kdump
Fri Aug 30 08:29:14 2013 [notice] esms1: Service named was restarted
Fri Aug 30 08:42:04 2013 [warning] esms1: Service nfs died
Fri Aug 30 08:42:05 2013 [notice] esms1: Service nfs was restarted
Fri Aug 30 09:18:03 2013 [warning] lake-esl: Check 'ntp' is in state FAIL on lake-esl
For details type: events details 3118
Fri Aug 30 09:18:03 2013 [warning] oss001: Check 'ntp' is in state FAIL on oss001
. . .
esms1]% events details 3118
ntpd not synchronized to a time server
```

7.8.2 View the rsync Log

Procedure 104 shows you how to view the rsync log which records which changes were stored to a particular device during the last image update operation.

Procedure 104. View the rsync log

1. Log in to the CIMS node as root and start cmsh.

```
esms1# cmsh
```

2. Type **device** at the cmsh prompt to enter device mode and display the device mode prompt:

```
[esms1]% device
[esms1->device]%
```

3. Type **synclog** ***device***:

```
[esms1->device]% synclog lustre1-oss001
```

Configure BIOS for DELL™ R620/R720 CIMS Nodes [A]

This procedure describes how to change the system setup for the CIMS: the network connections, remote power control, and the remote console.

Procedure 105. Configure BIOS for Dell R620/R720 CIMS

This procedure includes detailed steps for the Dell R620/R720 server. Depending on your server model and version of BIOS configuration utility, there could be minor differences in the steps to configure your system. For more information, refer to the documentation for your Dell server.

When configuring a secondary CIMS, do not disable the embedded NIC in the BIOS settings on the secondary CIMS. The secondary CIMS needs to initially PXE boot from the primary CIMS to perform the cloning operation. Also, the secondary CIMS should have `Boot Sequence` set to `Integrated Nic Hard drive C:`, and then the DVD/Optical drive.

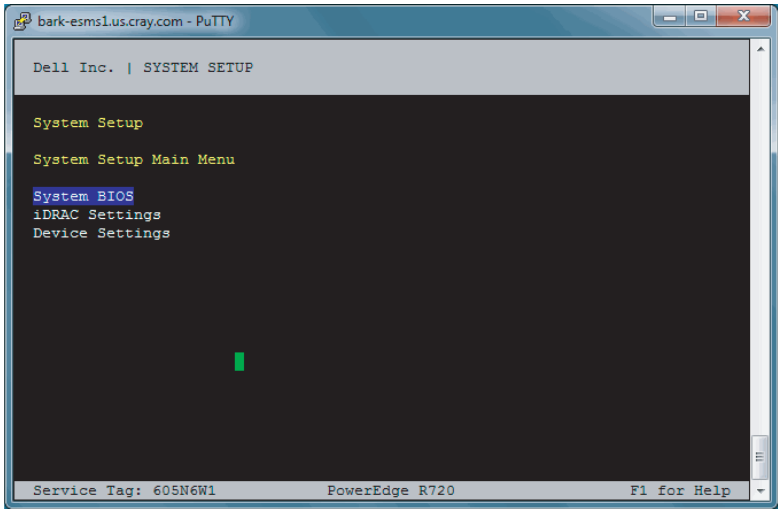
1. Watch as the system reboots. When the BIOS power-on self-test (POST) process begins, **quickly press the `F2` key** after the following messages appear in the upper-right of the screen.

   ```
       F2 = System Setup
     F10 = System Services
   F11 = BIOS Boot Manager
          F12 = PXE Boot
   ```

   When the `F2` keypress is recognized, the `F2 = System Setup` line changes to `Entering System Setup`.

   After the POST process completes and all disk and network controllers have been initialized, the **Dell System Setup** screen appears. The following submenus are available on the System Setup Main Menu:

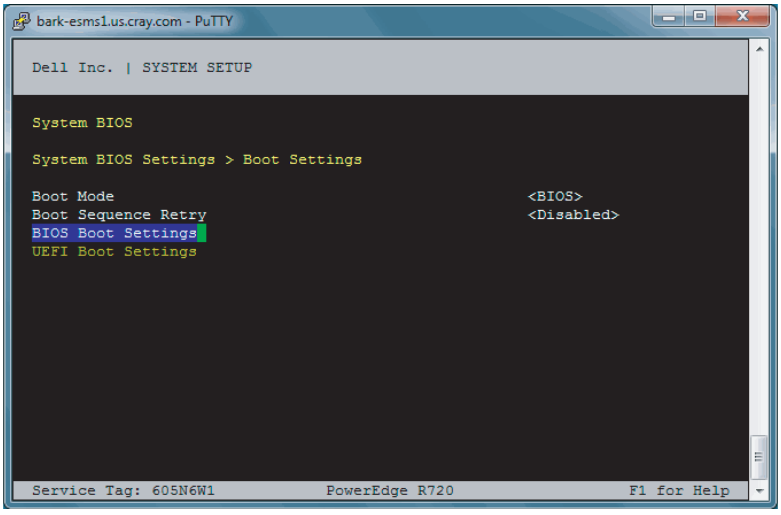
Figure 43. Dell R620/R720 BIOS Menu



Use the `Tab` key to move to different areas on the screen. To select an item, use the `up-arrow` and `down-arrow` keys to highlight the item, then press the `Enter` key. Press the `Escape` key to exit a submenu and return to the previous screen.

2. Change the System BIOS settings.

Figure 44. Dell R620/R720 BIOS Boot Settings



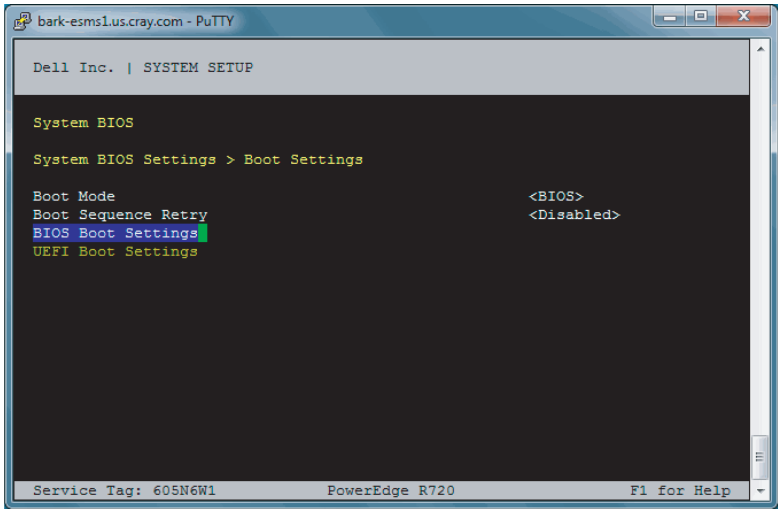
a. Select **System BIOS**, then press `Enter`.

b. Select **Boot Settings**, then press `Enter`.

c. Select **BIOS Boot Settings**, then press `Enter`.

d. Select **Boot Sequence**, then press `Enter` to view the boot settings.

Figure 45. Dell R620/R720 BIOS Boot Sequence



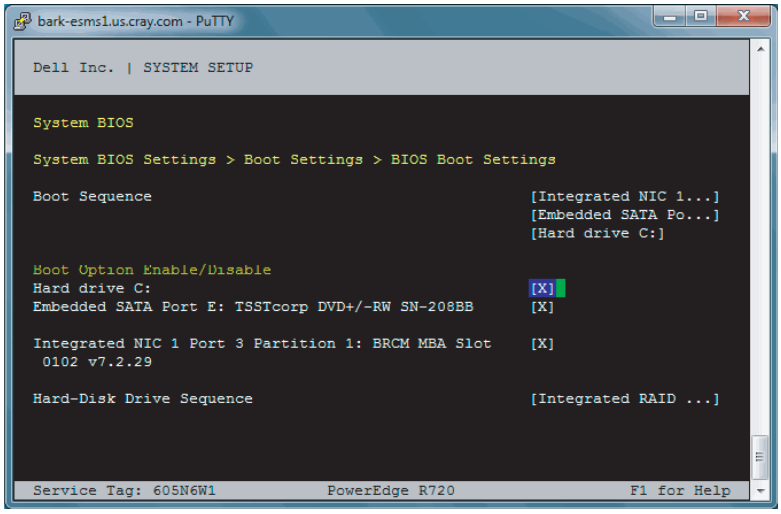
e. If creating a stand-alone CIMS, change the boot order in the pop-up window so that the optical drive appears first, then hard drive, then integrated NIC last.

   If creating an HA CIMS, do not disable the embedded NIC in the BIOS settings for the secondary CIMS. The secondary CIMS must initially PXE boot from the primary CIMS to perform the cloning operation. Set the secondary CIMS boot sequence to boot from the integrated NIC first, then hard drive, then DVD/optical drive last. See step 12.

   Tip: Use the `up-arrow` or `down-arrow` key to highlight an item, then use the `+` and `-` keys to move the item up or down.

f. Enable **Hard drive C:** under the **Boot Option/Enable/Disable** section.

Figure 46. Dell R620/R720 BIOS Boot Settings



g. Press `Enter` to return to the **BIOS Boot Settings** screen.

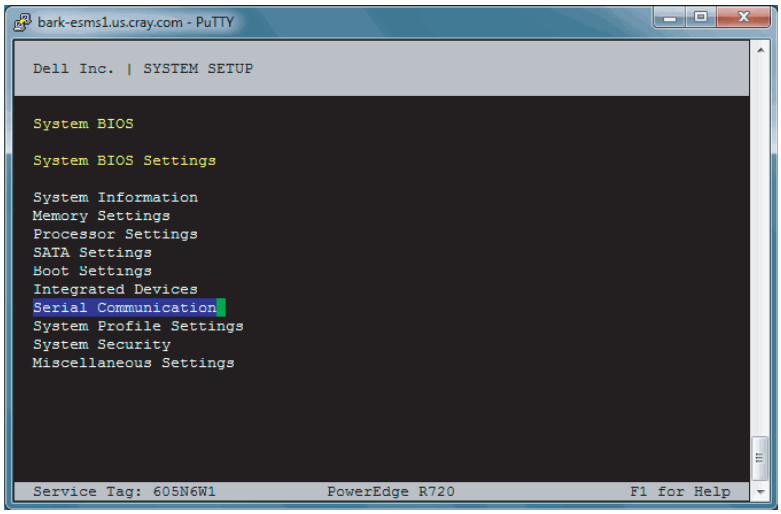
h. Press `Escape` to exit **BIOS Boot Settings**.

i. Press `Escape` to exit **Boot Settings** and return to the **System BIOS Settings** screen.

3. Change the serial communication settings.

   a. On the **System BIOS Settings** screen, select **Serial Communication**.

Figure 47. Dell R620/R720 BIOS Serial Communication Settings

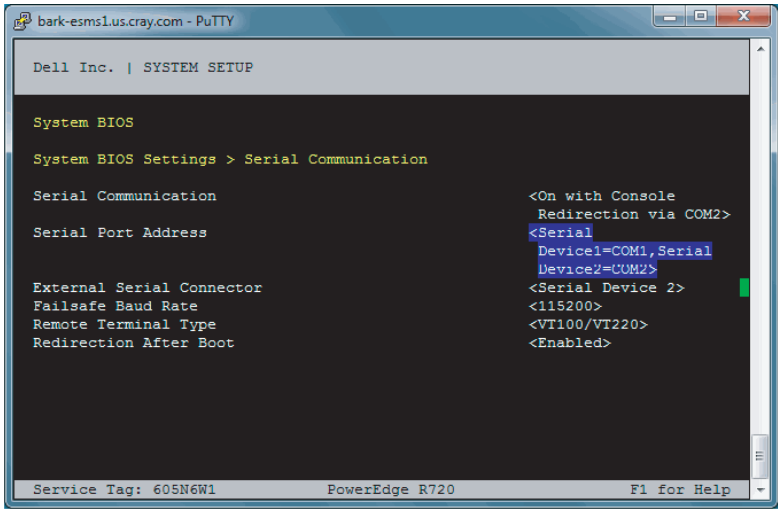


   b. On the **Serial Communication** screen, select **Serial Communication** and press `Enter`. A pop-up window displays the available options.

c. Select **On with Console Redirection via COM2**, then press `Enter`.

d. Select **Serial Port Address**, then select **Serial Device1=COM1, Serial Device2=COM2**, and press `Enter`.

Figure 48. Dell R620/R720 BIOS Serial Port Address Settings



e. Select **External Serial Connector**, then press `Enter`. A pop-up window displays the available options.

f. In the pop-up window, select **Remote Access Device**, then press `Enter` to return to the previous screen.

g. Select **Failsafe Baud Rate**, then press `Enter`. A pop-up window displays the available options.

h. In the pop-up window, select **115200**, then press `Enter` to return to the previous screen.

i. Press the `Escape` key to exit the **Serial Communication** screen.

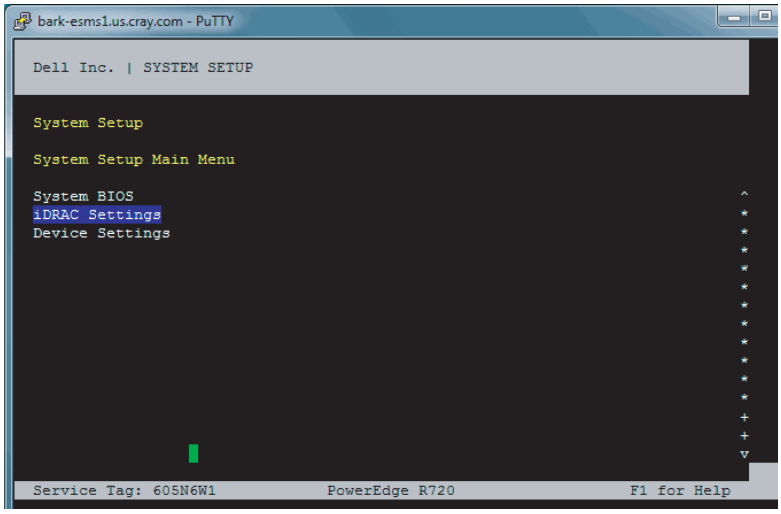
j. Press the `Escape` key to exit the **System BIOS Settings** screen.

k. A `"Settings have changed"` message appears. Select **Yes** to save your changes.

l. A `"Settings saved successfully"` message appears. Select **Ok**.

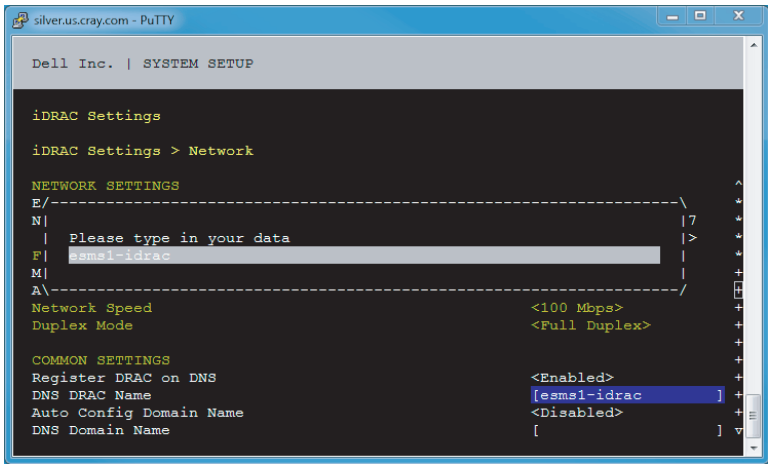
4. On the System Setup Main Menu, select **iDRAC Settings**, then press `Enter`.

Figure 49. Dell R620/R720 BIOS iDRAC Settings



5. Select **Network**, then press `Enter`. A long list of network settings is displayed.
6. Use the `down-arrow` key to scroll to **DNS DRAC Name** and press `Enter`.
7. Enter an iDRAC hostname that is similar to the CIMS node hostname. For example, `esms1-idrac`.

Figure 50. Dell R620/R720 BIOS iDRAC Name



8. Change the IPv4 settings.
   a. Use the `down-arrow` key to scroll to the **IPV4 SETTINGS** list.
   b. Ensure that IPv4 is enabled.
      1) If necessary, select **Enable IPV4** and press `Enter`.

2) In the pop-up window, select **<Enabled>**.

3) Press `Enter` to return to the previous screen.

c. Ensure that DHCP is disabled.

1) If necessary, select **Enable DHCP** and press `Enter`.

2) In the pop-up window, select **<Disabled>**.

3) Press `Enter` to return to the previous screen.

d. Change the IP address.

1) Select **IP Address**. A pop-up window opens for entering the new data.

2) In the pop-up window, enter the IP address of the iDRAC interface (`ipmi0`) for `site-admin-net` on the CIMS.

3) Press `Enter` to return to the previous screen.

e. Change the gateway.

1) Select **Gateway**. A pop-up window opens for entering the new data.

2) In the pop-up window, enter the appropriate value for the gateway of the `site-admin-net` network.

3) Press `Enter` to return to the previous screen.

f. Change the subnet mask.

1) Select **Subnet Mask**. A pop-up window opens for entering the new data.

2) In the pop-up window, enter the subnet mask for `site-admin-net` (such as `255.255.255.0`).

3) Press `Enter` to return to the previous screen.

g. Change the DNS server settings.

1) Select **Preferred DNS Server**. A pop-up window opens for entering the new data.

2) In the pop-up window, enter the IP address of the primary DNS server.

3) Press `Enter` to return to the previous screen.

4) Select **Alternate DNS Server**. A pop-up window opens for entering the new data.

5) In the pop-up window, enter the IP address of the alternate DNS server.

6) Press `Enter` to return to the previous screen.

9. Change the IPMI settings to enable the Serial Over LAN (SOL) console.

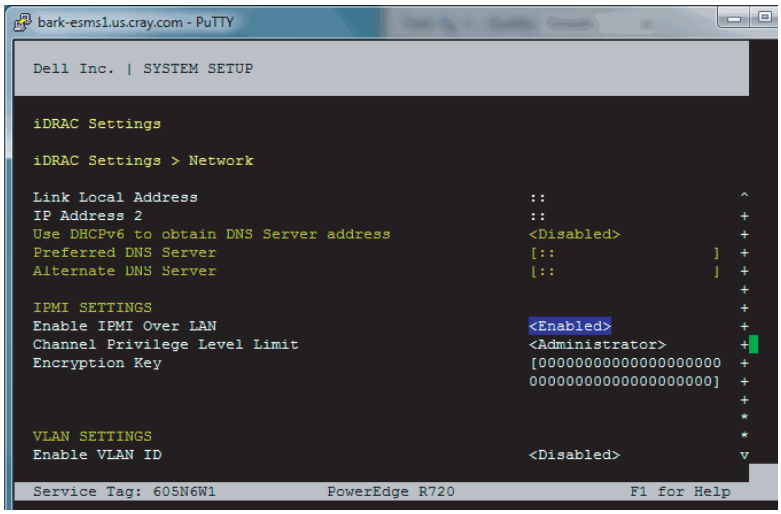
a. Use the `down-arrow` key to scroll to the **IPMI SETTINGS** list.

b. Ensure that IPMI over LAN is enabled.

1) If necessary, select **Enable IPMI over LAN**, then press `Enter`.

2) In the pop-up window, select **<Enabled>**.

Figure 51. Dell R620/R720 BIOS Enable IPMI Over LAN (SOL)



3) Press `Enter` to return to the previous screen.

c. Verify that **Channel Privilege Level Limit** is set to **Administrator**.

1) If necessary, select **Channel Privilege Level Limit** to display the pop-up window.

2) In the pop-up window, select **Administrator**.

3) Press `Enter` to return to the previous screen.

d. Press the `Escape` key to exit the Network screen and return to the **iDRAC Settings** screen.

10. On the **iDRAC Settings** screen, change the user configuration settings.

a. Use the `down-arrow` key to highlight **User Configuration**, then press `Enter`.

b. Confirm that **User Name** is **root**.

1) If necessary, select **User Name**. A pop-up window opens to let you enter the user name.

2) In the pop-up window, enter `root`.

3) Press `Enter` to return to the previous screen.

c. Select **Change Password**. A pop-up window opens to let you create a new password.

d. In the pop-up window, enter a new password.

e. In the next pop-up window, reenter the new password to confirm it.

f. Press the `Escape` key to exit the **User Configuration** screen.

11. Change the LCD configuration to show the hostname in the LCD display.

a. On the **iDRAC Settings** screen, use the `down-arrow` key to scroll down and highlight **LCD**, then press `Enter`.

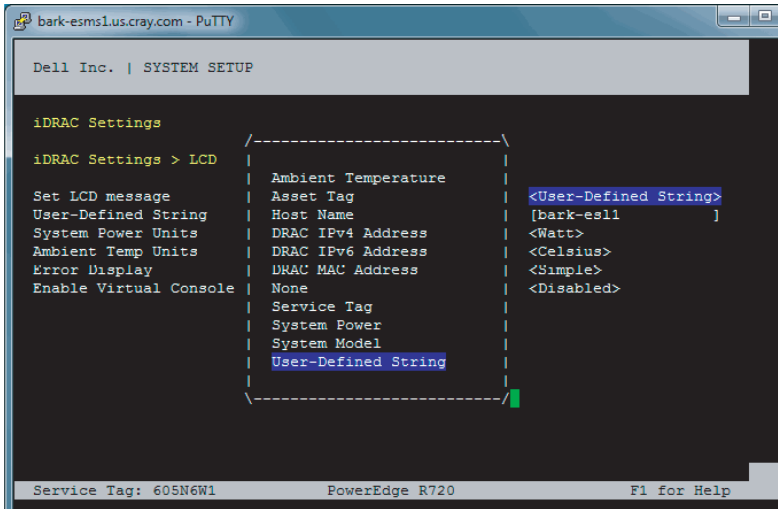
b. Select **Set LCD message**. A pop-up window opens (line 1).

c. In the pop-up window, select **User-Defined String**, then press `Enter`.

d. Select **User-Defined String**, then press `Enter`. A text pop-up window opens (line 2) for entering the new string.

e. In the text pop-up window, enter the CIMS hostname (such as `esms1`), then press `Enter`.

Figure 52. Dell R620/R720 BIOS iDRAC LCD Settings



f. Press the `Escape` key to exit the LCD screen.

g. Press the `Escape` key to exit the **iDRAC Settings** screen.

h. A `"Settings have changed"` message appears. Select **Yes**, then press `Enter` to save your changes.

i. A `"Settings saved successfully"` message appears. Select **Ok**, then press `Enter`. The main screen (System Setup Main Menu) appears.

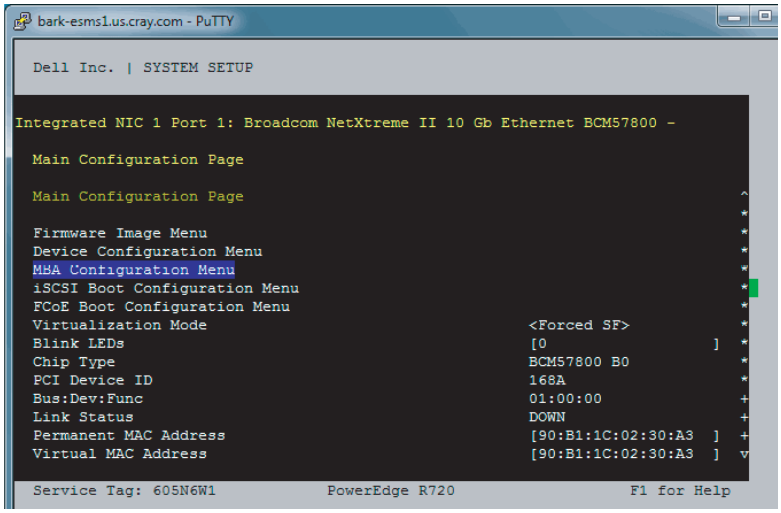
12. Disable the integrated NIC device by changing the setting for the integrated NIC on port 1 from **PXE** to **None**. If you are configuring a secondary CIMS in an HA configuration, set the integrated NIC port (on `esmaint-net`) to **PXE** so that it PXE boots from the primary CIMS.

    a. On the System Setup Main Menu, select **Device Settings**, then press `Enter`.

    b. On the Device Settings screen, select **Integrated NIC 1 Port 1: ...**, then press `Enter`.

    c. On the Main Configuration Page screen, select **MBA Configuration Menu**, then press `Enter`.

Figure 53. Dell R620/R720 BIOS MBA Configuration Settings



    d. On the MBA Configuration Menu screen, select **Legacy Boot Protocol**, then press `Enter`. A pop-up window displays the available options.

    e. In the pop-up window, use the `down-arrow` key to highlight **None**, then press `Enter`.

    f. Press the `Escape` key to exit the MBA Configuration Menu screen.

    g. Press the `Escape` key to exit the Main Configuration Page screen.

    h. Press the `Escape` key to exit the **Device Settings** screen.

    i. A `"Settings have changed"` message appears. Select **Yes**, then press `Enter` to save your changes.

    j. A `"Settings saved successfully"` message appears. Select **Ok**, then press `Enter`. The main screen (System Setup Main Menu) appears.

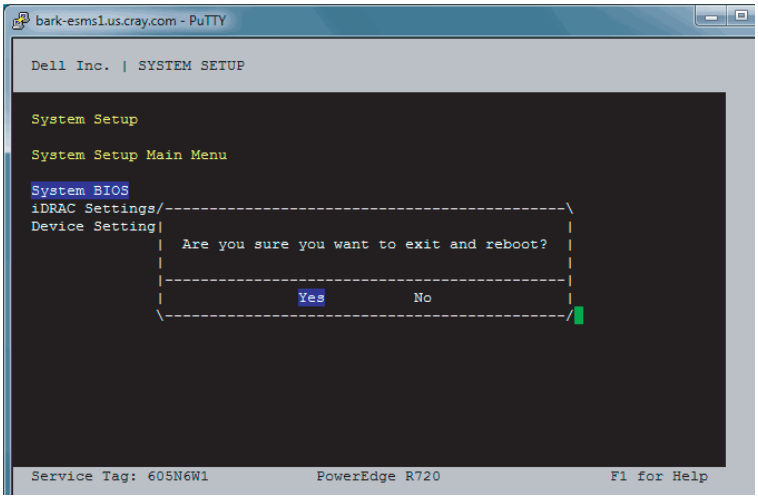
13. Insert the Bright software media in the optical (DVD) drive of the CIMS node.

    Important: The CIMS must boot from the Bright software media to configure and install the Bright software. While the memory-resident operating system is loaded, you must copy the XML configuration file from the Cray ESM software media and save it on the CIMS, then edit and save the file to `/root/cm` with your site-specific configuration settings. Be aware that changes to the XML configuration file will be lost if the file is not saved to a remote system before the CIMS is rebooted.

14. While viewing the System Setup Main Menu, press the `Escape` key to exit the Dell System Setup utility.

15. A message appears asking if you want to exit and reboot. Select **Yes**. The server will restart the boot process.

Figure 54. Dell R620/R720 System BIOS Settings



Configure BIOS for DELL™ R720 Slave Nodes [B]

Before you can configure a stand-alone or managed (using Bright Cluster Manager) slave node, you must change the BIOS and remote access controller (iDRAC) settings. The configuration settings are slightly different for managed and unmanaged slave nodes. This procedure may also apply to other DELL computers (R620) using a similar BIOS.

- Managed slave nodes must be cabled to both the CIMS node administration network (`esmaint-net`) and IPMI network (`ipmi-net`).
- A slave node running a workload manager such as Moab® or TORQUE must be cabled directly to the SDB on the Cray system over the `wlm-net` network.
- Slave nodes must be configured to provide IPMI Serial Over LAN (SOL) for remote console support.
- A managed slave node must be configured to PXE boot from the CIMS node (embedded NIC) before attempting to boot from the local disk.
- Unmanaged slave nodes must be cabled to customer administration and/or IPMI network.
- Unmanaged slave nodes must be configured to boot from the optical (DVD) media initially, then configured to boot from the hard disk.
- All DMP slave nodes should be configured stay powered off after a site power failure. Cray recommends that the CIMS node be powered on first and become operational before each slave node is powered on.

Use the following procedure to change the BIOS and iDRAC settings for a DELL R720 slave node.

Procedure 106. Configure a R720 slave node BIOS and iDRAC

In this utility, use the `Tab` key to move to different areas on the screen. To select an item, use the `up-arrow` and `down-arrow` keys to highlight the item, then press the `Enter` key. Press the `Escape` key to exit a submenu and return to the previous screen.

1. Power up the slave node. When the BIOS power-on self-test (POST) process begins, **quickly press the F2 key** after the following messages appear in the upper-right of the screen.

```
          F2 = System Setup
      F10 = System Services
  F11 = BIOS Boot Manager
           F12 = PXE Boot
```

   When the `F2` keypress is recognized, the `F2 = System Setup` line changes to `Entering System Setup`.

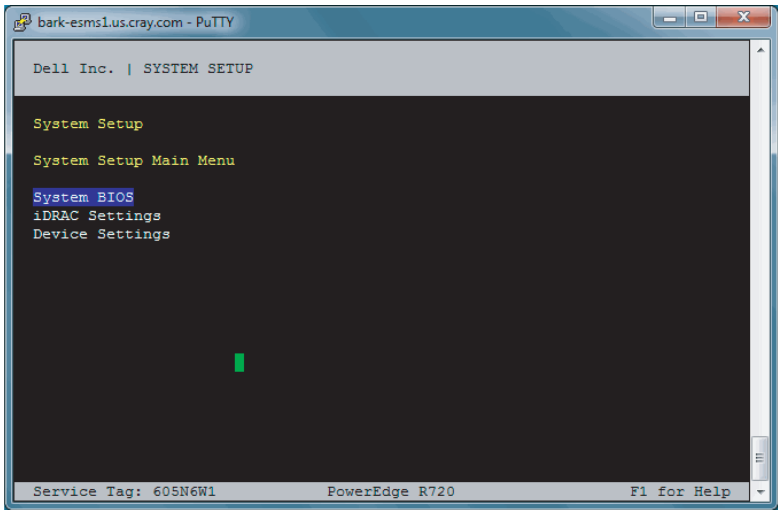
   After the POST process completes and all disk and network controllers have been initialized, the **Dell System Setup** screen appears. The following submenus are available:

```
System BIOS
iDRAC Settings
Device Settings
```

2. Change the system BIOS settings.

   a. Select **System BIOS**, then press `Enter`. See Figure 55.

Figure 55. Dell 720 System BIOS Settings



   b. Select **Boot Settings**, then press `Enter`.

   c. Select **BIOS Boot Settings**, then press `Enter`.

   d. Select **Boot Sequence**, then press `Enter` to view the boot settings.

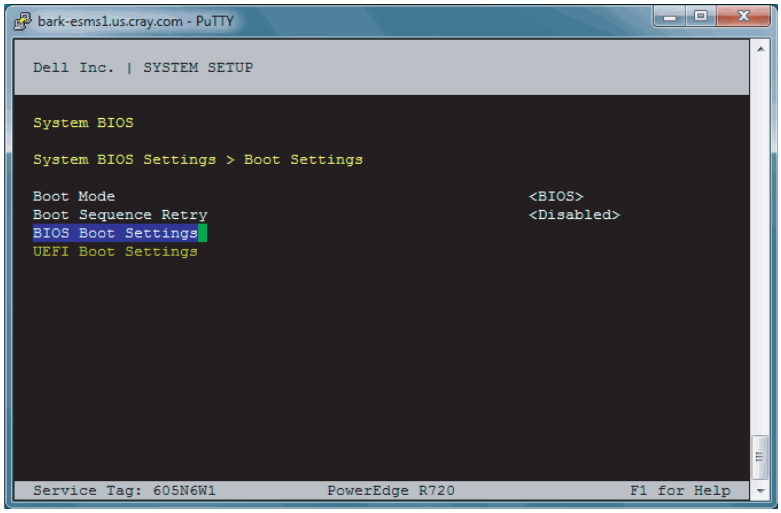
   e. Set the boot sequence for either an unmanaged or managed node:

1) **Managed slave nodes**: change the boot order so that the integrated NIC appears first, before the optical (DVD) drive. The hard drive should be last on the list. See Figure 56.

   Tip: Use the `up-arrow` or `down-arrow` key to highlight an item, then use the `+` and `-` keys to move the item up or down.

2) **Unmanaged slave nodes**: change the boot order so that the DVD is the first device in the sequence. After the operating system is installed, configure the boot sequence to boot from the hard drive first.

Figure 56. Dell 720 Boot Sequence BIOS Settings



f. For unmanaged nodes, be sure that **Hard Drive C:** and **Embedded SATA Port** are enabled under the **Boot Option Enable/Disable** section. See Figure 56. For managed nodes, be sure that the **Integrated NIC Port** is enabled so that the node can PXE boot from the CIMS (embedded NIC) before attempting to boot from the local disk.

g. Press `Enter` to return to the **BIOS Boot Settings** screen.

h. Press `Escape` to exit **BIOS Boot Settings**.

i. Press `Escape` to exit **Boot Settings** and return to the **System BIOS Settings** screen.

3. Change the serial communication settings.

   a. On the **System BIOS Settings** screen, select **Serial Communication**.

   b. On the **Serial Communication** screen, select **Serial Communication**. A pop-up window displays the available options.

   c. Select **On with Console Redirection via COM2**, then press `Enter`.

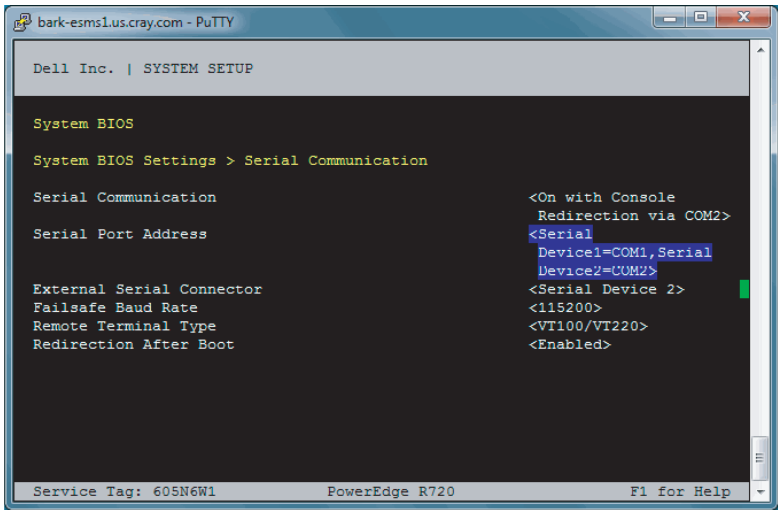
d. Verify that **Serial Port Address** is set to **Serial Device1=COM1, Serial Device2=COM2**.

Note: This setting enables the remote console. If this setting is incorrect, you cannot use a remote console to access the node.

1) If necessary, press `Enter` to display the available options.

2) Change the setting to **Serial Device1=COM1, Serial Device2=COM2**. See Figure 57.

Figure 57. Dell 720 Serial Device BIOS Settings



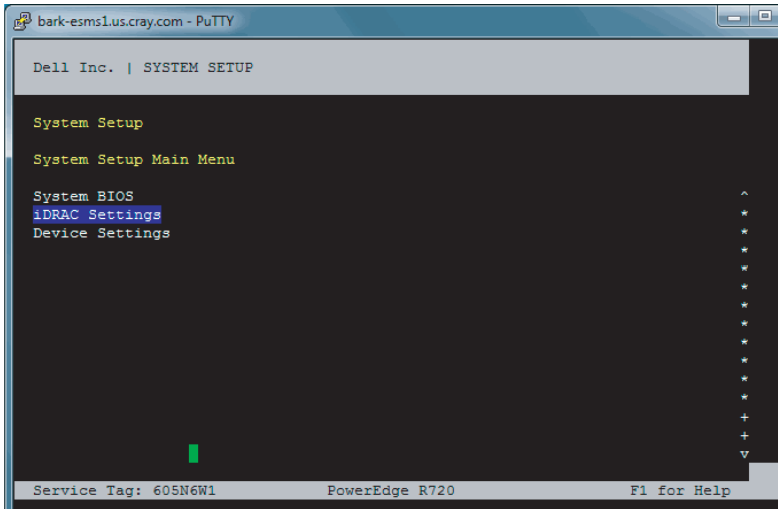
3) Press `Enter` to return to the **Serial Communication** screen.

e. Select **External Serial Connector**. A pop-up window displays the available options.

f. In the pop-up window, select **Remote Access Device**, then press `Enter` to return to the previous screen.

g. Select **Failsafe Baud Rate**. A pop-up window displays the available options.

Figure 58. Dell 720 Serial Communication BIOS Settings



h. In the pop-up window, select **115200**, then press `Enter` to return to the previous screen.

i. Press the `Escape` key to exit the **Serial Communication** screen.

j. Press the `Escape` key to exit the **System BIOS Settings** screen.

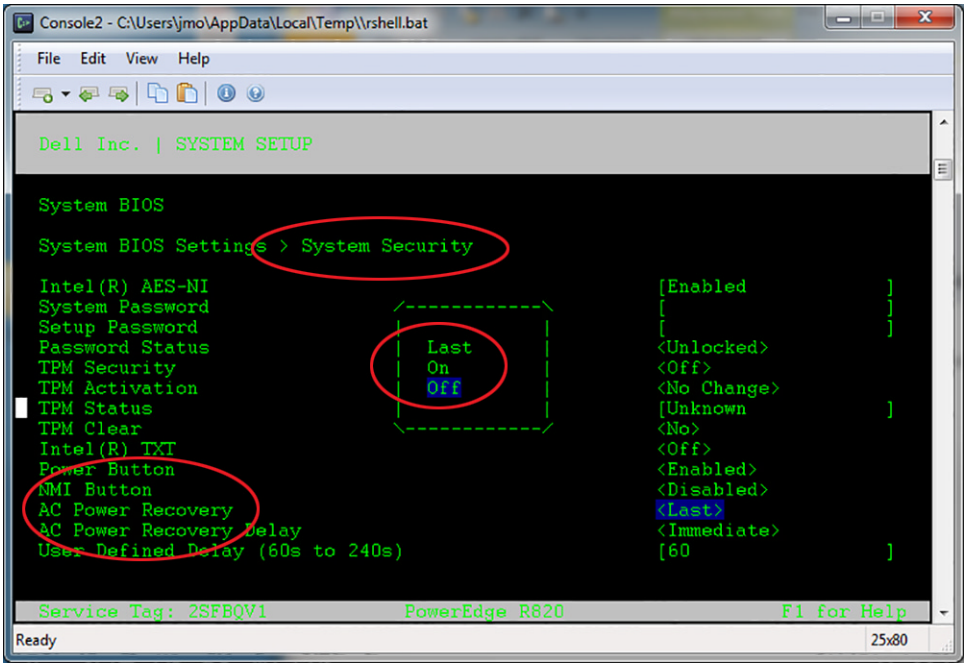
k. Press the `Escape` key to exit the **BIOS Settings** screen.

l. A `"Settings have changed"` message appears. Select **Yes** to save your changes.

m. A `"Settings saved successfully"` message appears. Select **OK**.

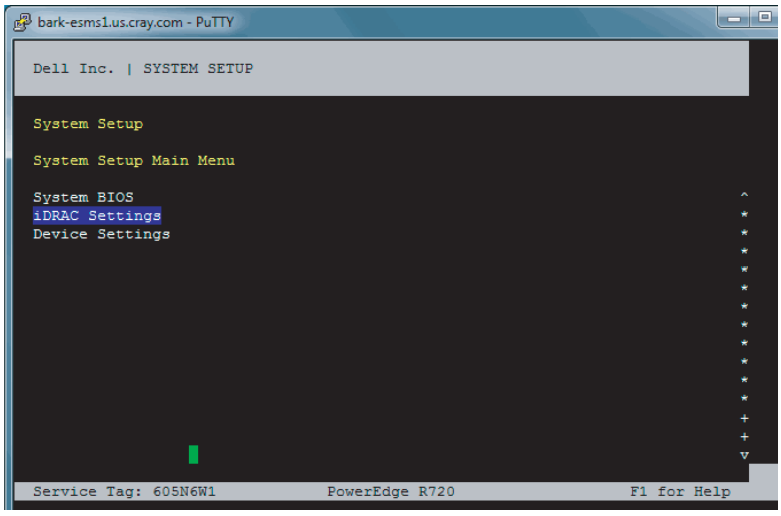
4. From the System BIOS settings menu, select **System Security**, and then **AC Power Recovery** to set the slave node to remain powered off after a system power failure. Cray recommends that the CIMS node power up and become operational before slave nodes.

Figure 59. Set Slave Node Auto-power On Setting to Off



5. On the System Setup Main Menu, select **iDRAC Settings**, then press `Enter`. See Figure 60.

Figure 60. Dell 720 iDRAC BIOS Settings



6. Select **Network**, then press `Enter`. A long list of network settings is displayed.
7. Change the IPMI settings to enable the Serial Over LAN (SOL) console.
   a. Use the `down-arrow` key to scroll to the **IPMI SETTINGS** list.

b. Ensure that **IPMI over LAN** (or **Enable IPMI over LAN**) is enabled.

1) If necessary, select **IPMI over LAN**, then press `Enter`.

2) In the pop-up window, select **<Enabled>**.

3) Press `Enter` to return to the previous screen.

c. Press the `Escape` key to exit the Network screen and return to the **iDRAC Settings** menu.

8. Change the LCD configuration to show the hostname in the LCD display.

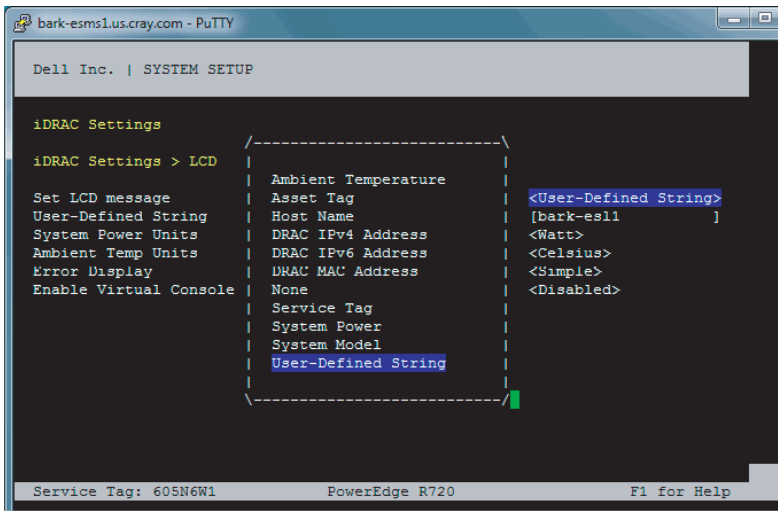
a. On the **iDRAC Settings** screen, use the `down-arrow` key to scroll down and highlight **LCD** (or **Front Panel Security**), then press `Enter`.

b. Select **Set LCD message**. A pop-up window opens.

c. In the pop-up window, select **User-Defined String**, then press `Enter`.

d. Select **User-Defined String** (again), then press `Enter`. A text pop-up window opens for entering the new string. See Figure 61.

Figure 61. Dell 720 iDRAC BIOS LCD Settings



e. In the text pop-up window, enter the hostname (such as `eslogin1`), then press `Enter`.

f. Press the `Escape` key to exit the LCD screen.

g. Press the `Escape` key to exit the Network screen.

h. Press the `Escape` key to exit the **iDRAC Settings** screen.

i. A `"Settings have changed"` message appears. Select **Yes**, then press `Enter` to save your changes.

j. A `"Settings saved successfully"` message appears. Select **OK**, then press `Enter`.

9. Proceed to step 10 if you are configuring an unmanaged node. If you are configuring a managed node, change the device settings so that the node can PXE boot from the CIMS administration network (`esmaint-net`).

   a. On the **System Setup** main menu, select **Device Settings**, then press `Enter`.

   b. In the **Device Settings** window, select **Integrated NIC 1 Port *N* ...**, then press `Enter`. The Main Configuration Page opens.

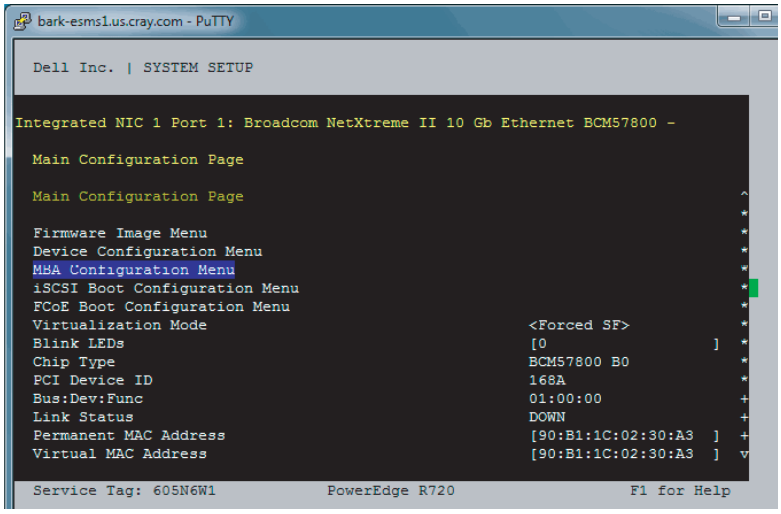
      Tip: Choose the NIC port number that corresponds to the Ethernet port for the `esmaint-net` network.

      - If `esmaint-net` uses the first Ethernet port (`eth0`), select **Integrated NIC 1 Port 1 ...**
      - If `esmaint-net` uses the third Ethernet port (`eth2`), select **Integrated NIC 1 Port 3 ...**

      Note: PXE booting must be disabled for the other three Ethernet ports.

   c. On the **Main Configuration Page** screen, select **MBA Configuration Menu**, then press `Enter`. See Figure 62.

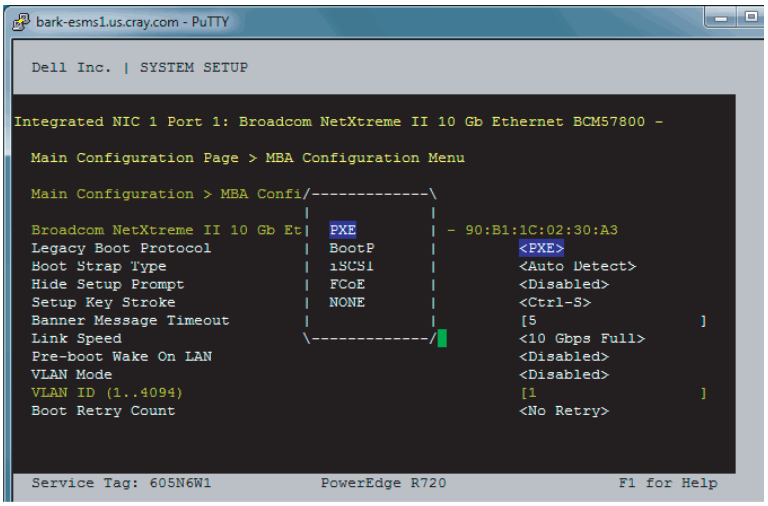
Figure 62. Dell 720 MBA Configuration Menu BIOS Settings



   d. On the MBA Configuration Menu screen, select **Legacy Boot Protocol**, then press `Enter`. A pop-up window displays the available options.

   e. In the pop-up window, use the `down-arrow` key to highlight **PXE**, then press `Enter`. See Figure 63.

Figure 63. Dell 720 Legacy Boot Protocol BIOS Settings



f. Press the `Escape` key to exit the MBA Configuration Menu screen.

g. Press the `Escape` key to exit the Main Configuration Page screen.

h. Verify that **Legacy Boot Protocol** is set to **None** for the other three Ethernet ports. If necessary, repeat step 9.b through step 9.g to change the setting for these three ports.

i. Press the `Escape` key to exit the **Device Settings** screen.

j. A `"Settings have changed"` message appears. Select **Yes**, then press `Enter` to save your changes.

k. A `"Settings saved successfully"` message appears. Select **OK**, then press `Enter`. The main screen (System Setup Main Menu) appears.

10. Save your changes and exit.

    a. Press `Escape` to exit the **System Setup Main Menu**.

    b. The utility displays the message `"Are you sure you want to exit and reboot?"` Select **Yes**.

Continue configuring the managed node in Bright or boot the unmanaged slave node from the stand-alone ESL media.

Configure BIOS for DELL™ R815 Managed CDL Nodes [C]

Before configuring a managed CDL node with Bright Cluster Manager® (Bright), you must change the BIOS and Dell remote access controller (iDRAC) settings. The following configuration is required for each CDL node.

- A CDL node must be cabled to both the CIMS administration network (`esmaint-net`) and IPMI network (`ipmi-net`).
- A CDL node using a workload manager such as TORQUE or Moab®, must be cabled directly to the SDB on the Cray system over the `wlm-net`.
- A CDL node must be configured to provide IPMI Serial Over LAN (SOL) for remote console support.
- A CDL node must be configured to PXE boot from the CIMS (embedded NIC) before attempting to boot from the local disk.
- All DMP slave nodes should be configured stay powered off after a site power failure. Cray recommends that the CIMS node be powered on first and become operational before each slave node is powered on.

Use the following procedure to change the BIOS and iDRAC settings for a CDL node.

Procedure 107. Configure a Dell R815 slave node BIOS and iDRAC

Note: This procedure shows specific steps for a Dell R815 system. See Procedure 106 for Dell R720 BIOS set up procedure.

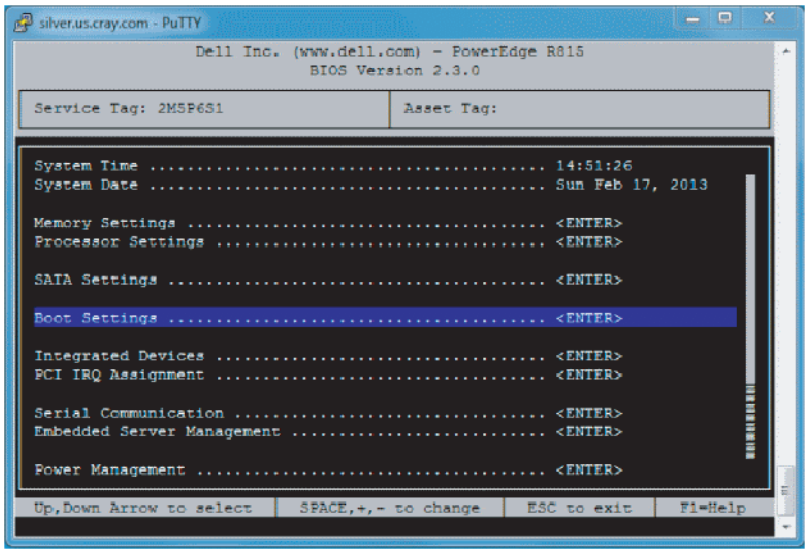
1. Power up the slave node. When the BIOS power-on self-test (POST) process begins, **quickly press the F2 key** after the following messages appear in the upper-right of the screen.

```
          F2 = System Setup
      F10 = System Services
F11 = BIOS Boot Manager
          F12 = PXE Boot
```

   When the `F2` keypress is recognized, the `F2 = System Setup` line changes to `Entering System Setup`.

2. Select **Boot Settings**, then press `Enter`.

Figure 64. Dell 815 Boot Settings Menu



a. Select **Boot Sequence**, then press `Enter` to view the boot settings.

b. In the pop-up window, change the boot order so that the integrated NIC appears first, before the optical (DVD) drive. The hard drive should be last on the list.

Figure 65. Dell 815 Boot Sequence Menu

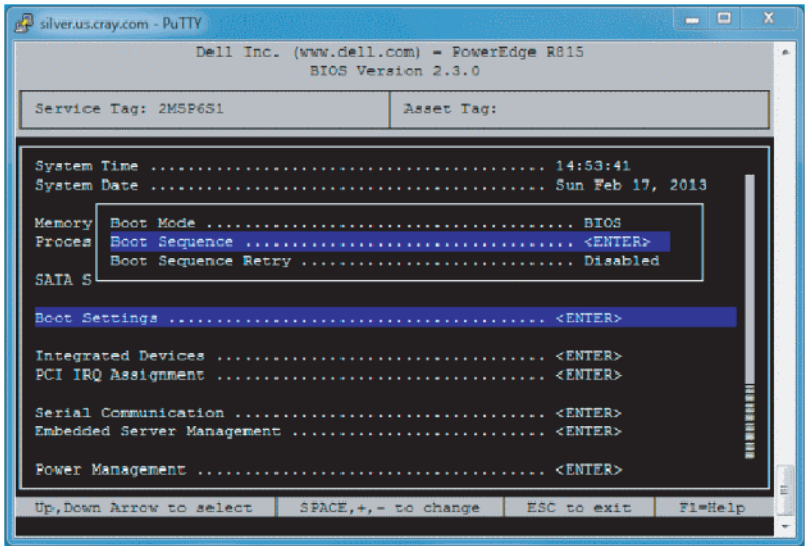
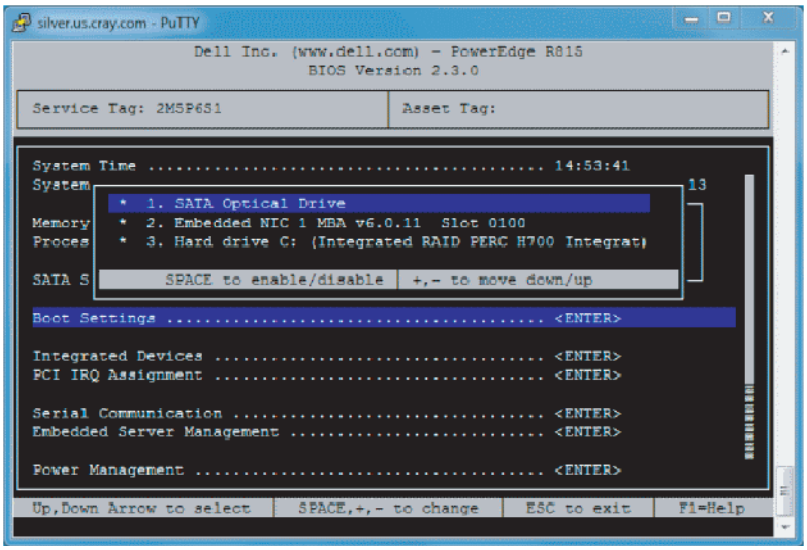


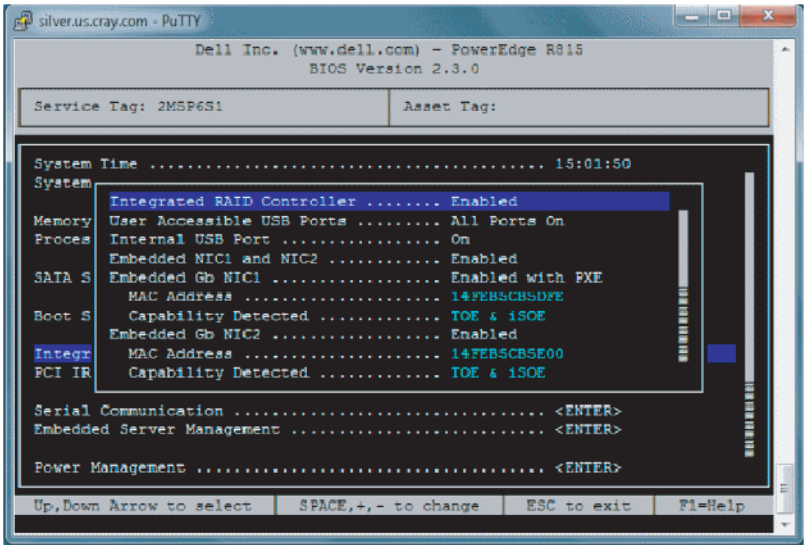
Figure 66. Dell 815 Boot Sequence Settings



c. Press `Enter` to return to the **BIOS Boot Settings** screen.

3. Press `Esc` to return to the System Setup Menu, scroll down and select **Integrated Devices**.

Figure 67. Dell 815 Integrated Devices (NIC) Settings



a. Set **Embedded NIC 1** to **Enabled with PXE**.

b. Set **Embedded Gb NIC 2** to **Enabled**.

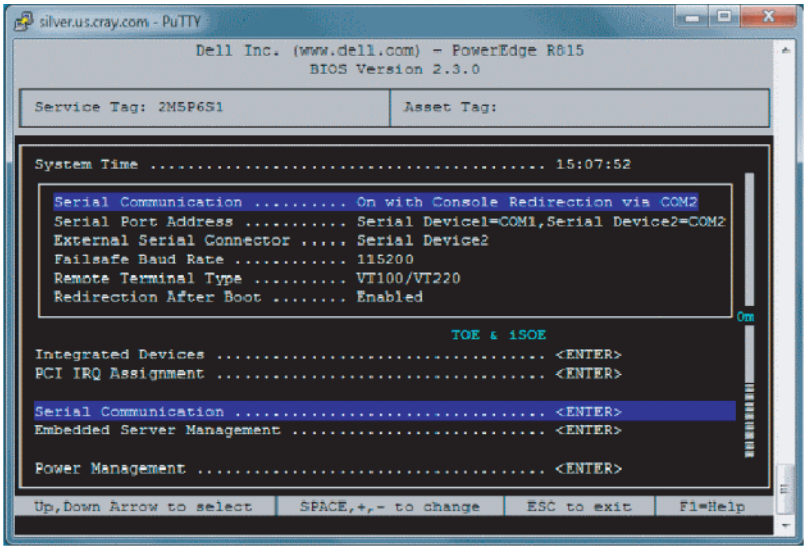
c. Scroll down and set **Embedded NIC 3** to **Enabled**.

d. Set **Embedded Gb NIC 4** to **Enabled**.

e. Press `Esc` to return to the System Settings Menu.

4. Change the serial communication settings.

Figure 68. Dell 815 Serial Communication BIOS Settings



a. Select **Serial Communication**.

b. Select **Serial Communication** and set it to **On** with **Console Redirection via COM2**.

c. Select **Serial Port Address** and set it to **Serial Device=COM1**, **Serial Device2=COM2**.

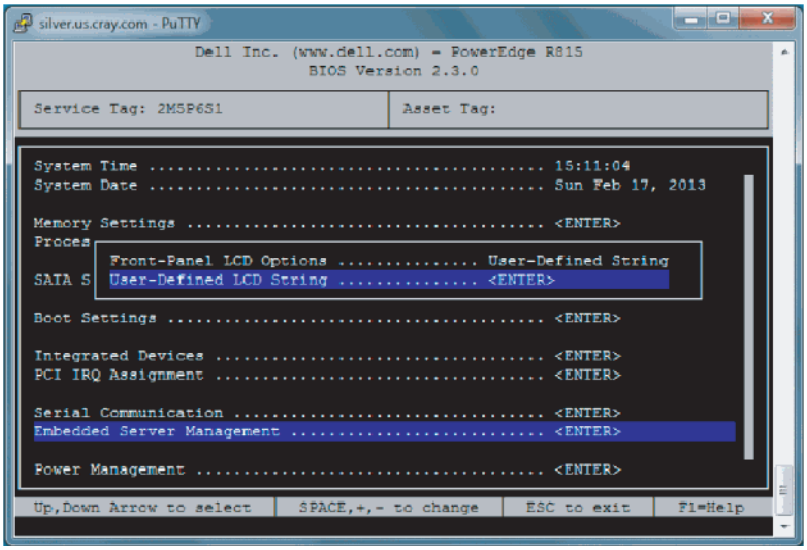
d. Select **External Serial Connector**, and set it to **Remote Access Device**.

e. Set **Failsafe Baud Rate** to **115200**.

f. Press `Esc` to return to the **System Setup Menu**.

5. Select **Embedded Server Management**.

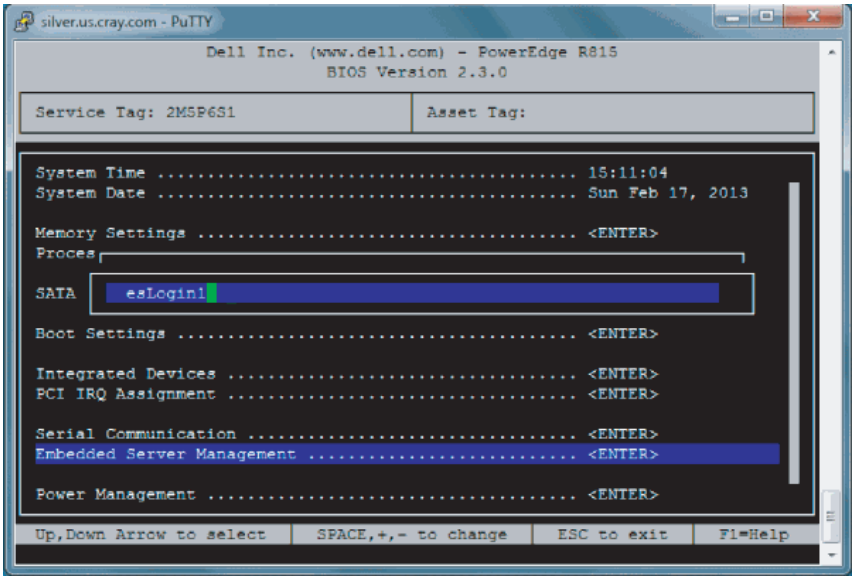
Figure 69. Dell 815 Embedded Server Management Settings



a. Set **Front-Panel LCD Options** to **User-Defined LCD String**.

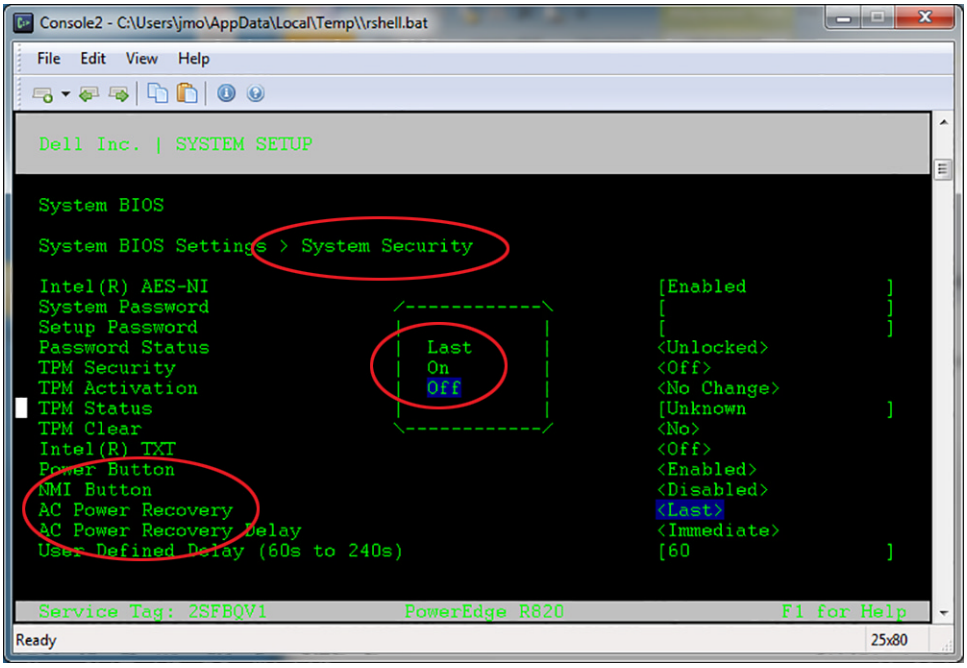
b. Set **User-Defined LCD String** to your login hostname, such as `eslogin1`.

Figure 70. Dell 815 User-defined LCD String Settings



6. From the System BIOS settings menu, select **System Security**, and then **AC Power Recovery** to set the slave node to remain powered off after a system power failure. Cray recommends that the CIMS node power up and become operational before slave nodes.

Figure 71. Set Slave Node Auto-power On Setting to Off



7. Save your changes and exit.

   a. Press `Escape` to exit the System Setup Main Menu.

   b. The utility displays the prompt "Are you sure you want to exit and reboot?" Select **Yes**.

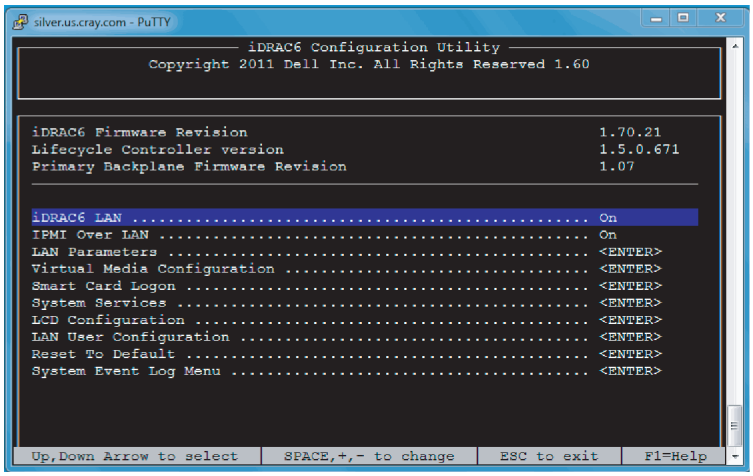
8. When the system reboots, press `Ctrl-E` to configure the iDRAC port settings.

```
www.dell.com

iDRAC6 Configuration Utility 1.60
Copyright 2011 Dell Inc. All Rights Reserved
Four 2.10 GHz Twelve-core Processors, L2/L3 Cache: 6 MB/10 MB
iDRAC6 FirmwareaRevisionHversion: 1.70.21
.
.
.'
 IPv4 Stack      : Enabled
 IP Address      :  10.148. 0 . 2
 Subnet mask     : 255.255. 0 . 0
 Default Gateway :  0 . 0 . 0 . 0
Press <Ctrl-E> for Remote Access Setup within 5 sec......
```

   a. Set the **iDRAC6 LAN** to **ON**.

   b. Set **IPMI Over LAN** to **ON**.

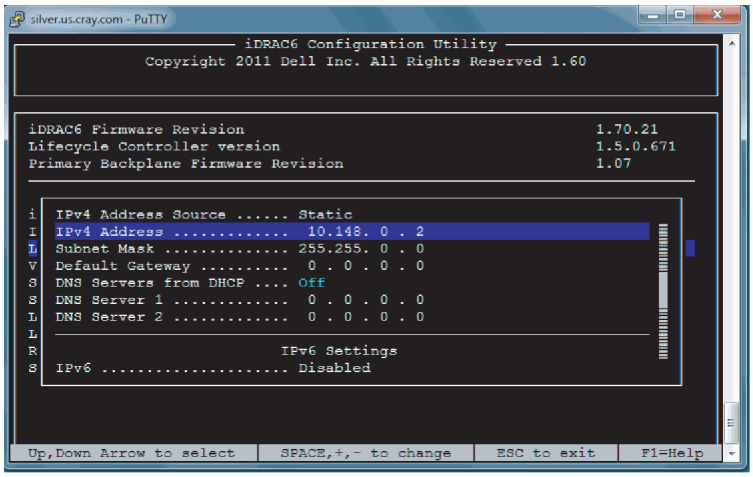
Figure 72. Dell 815 DRAC LAN Parameters Settings



c. Select **LAN Parameters** and press `Enter`. Set the IPv4 address to next available IP address on the `esmaint-net` network (10.148.0.x).

d. Press `Esc` to return to the iDRAC6 menu, and `Esc` to exit and save.

Figure 73. Dell 815 DRAC IPv4 Parameter Settings



Configure BIOS for a DELL™ R720 Managed CLFS Nodes [D]

Before you can configure a managed CLFS node with Bright Cluster Manager® (Bright), you must change the BIOS and Dell remote access controller (iDRAC) settings. The following configuration is required for each CLFS node.

- A CLFS node must be cabled to both the CIMS administration network (`esmaint-net`) and IPMI network (`ipmi-net`).
- A CLFS node using a workload manager such as TORQUE or Moab®, must be cabled directly to the SDB on the Cray system over the `wlm-net`.
- A CLFS node must be configured to provide IPMI Serial Over LAN (SOL) for remote console support.
- Processor hyper-threading (the "Logical Processor" setting) must be disabled on CLFS nodes.
- All DMP slave nodes should be configured remain powered off after a site power failure. Cray recommends that the CIMS node be powered on first and become operational before each slave node is powered on.

Use the following procedure to change the BIOS and iDRAC settings for a CLFS node.

Procedure 108. Configure a R720 CLFS node BIOS and iDRAC

Note: This procedure shows specific steps for a Dell 720 system.

1. Power up the slave node. When the BIOS power-on self-test (POST) process begins, **quickly press the F2 key** after the following messages appear in the upper-right of the screen.

```
          F2 = System Setup
      F10 = System Services
F11 = BIOS Boot Manager
          F12 = PXE Boot
```

   When the `F2` keypress is recognized, the `F2 = System Setup` line changes to `Entering System Setup`.

After the POST process completes and all disk and network controllers have been initialized, the **Dell System Setup** screen appears. The following submenus are available:

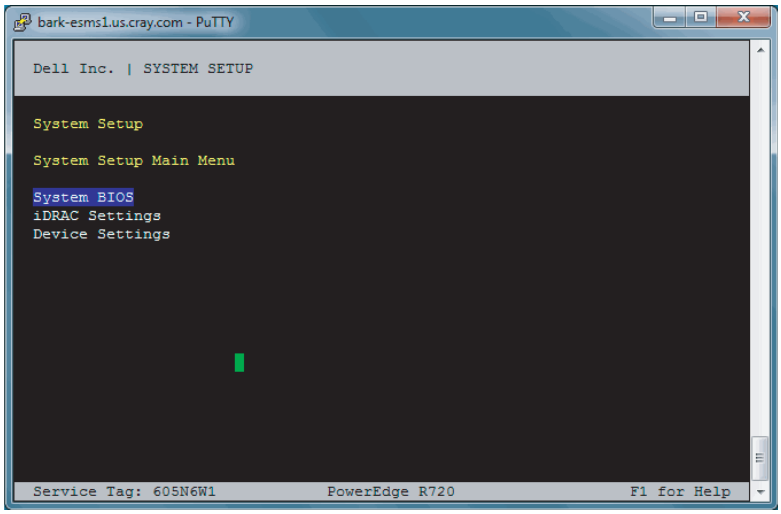
```
System BIOS
iDRAC Settings
Device Settings
```

Note: In this utility, use the `Tab` key to move to different areas on the screen. To select an item, use the `up-arrow` and `down-arrow` keys to highlight the item, then press the `Enter` key. Press the `Escape` key to exit a submenu and return to the previous screen.

2. Change the system BIOS settings.

   a. Select **System BIOS**, then press `Enter`. See Figure 74.

Figure 74. Dell 720 System BIOS Settings



   b. Select **Boot Settings**, then press `Enter`.

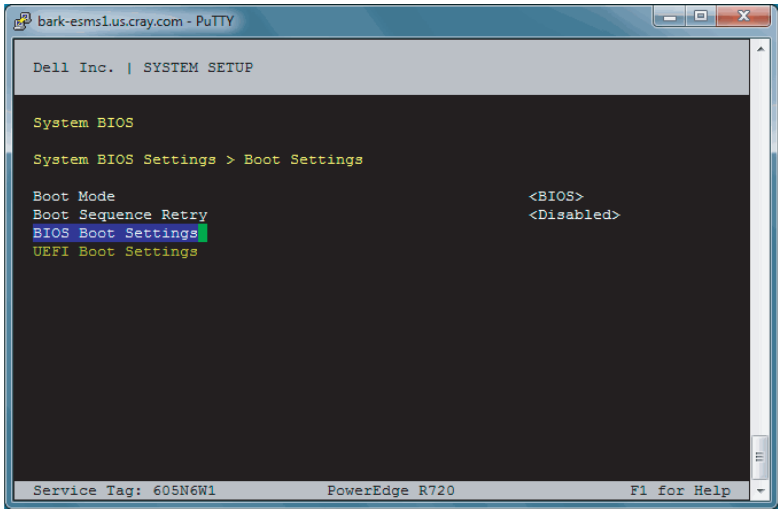
   c. Select **BIOS Boot Settings**, then press `Enter`.

   d. Select **Boot Sequence**, then press `Enter` to view the boot settings.

   e. In the pop-up window, change the boot order so that the integrated NIC appears first, before the optical (DVD) drive. The hard drive should be last on the list. See Figure 75.

      Tip: Use the `up-arrow` or `down-arrow` key to highlight an item, then use the `+` and `-` keys to move the item up or down.

Figure 75. Dell 720 Boot Sequence BIOS Settings



f. Be sure that **Hard Drive C:** is enabled under the **Boot Option Enable/Disable** section.

g. Press `Enter` to return to the **BIOS Boot Settings** screen.

h. Press `Escape` to exit **BIOS Boot Settings**.

i. Press `Escape` to exit **Boot Settings** and return to the **System BIOS Settings** screen.

3. On the **System BIOS Settings** screen, select **Processor Settings** and press `Enter`.

   a. Select **Logical Processor**, and press enter. Verify that **Logical Processor** is set to **Disabled**.

   b. Press `Escape` to exit **Processor Settings**.

4. Change the serial communication settings.

   a. On the **System BIOS Settings** screen, select **Serial Communication**.

   b. On the **Serial Communication** screen, select **Serial Communication**. A pop-up window displays the available options.

   c. Select **On with Console Redirection via COM2**, then press `Enter`.

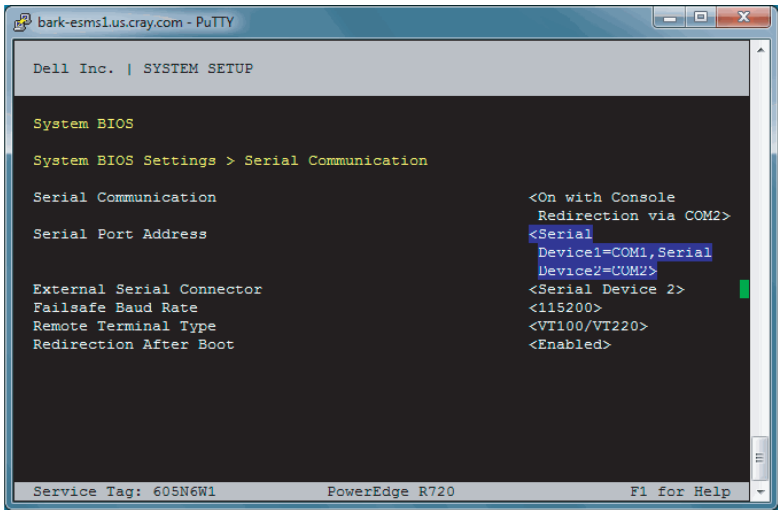
   d. Verify that **Serial Port Address** is set to **Serial Device1=COM1, Serial Device2=COM2**.

      Note: This setting enables the remote console. If this setting is incorrect, you cannot use a remote console to access the CLFS node.

      1) If necessary, press `Enter` to display the available options.

2) Change the setting to **Serial Device1=COM1, Serial Device2=COM2**. See Figure 76.

Figure 76. Dell 720 Serial Device BIOS Settings



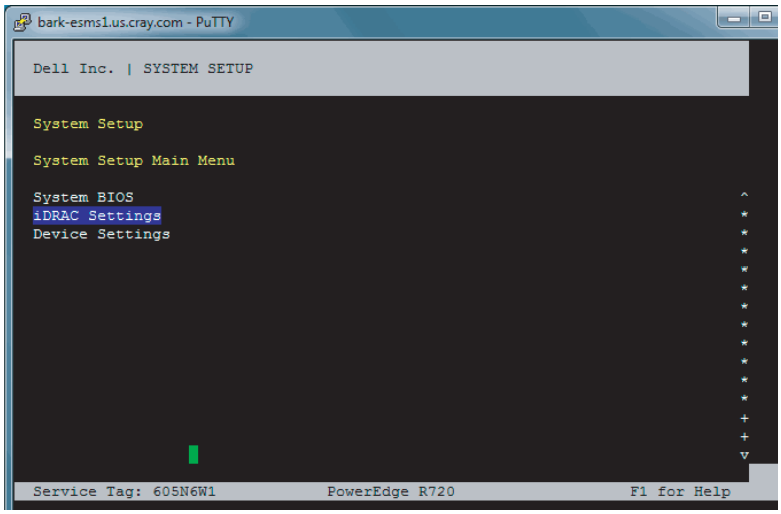
3) Press `Enter` to return to the **Serial Communication** screen.

e. Select **External Serial Connector**. A pop-up window displays the available options.

f. In the pop-up window, select **Remote Access Device**, then press `Enter` to return to the previous screen.

g. Select **Failsafe Baud Rate**. A pop-up window displays the available options.

Figure 77. Dell 720 Serial Communication BIOS Settings



h. In the pop-up window, select **115200**, then press `Enter` to return to the previous screen.

i. Press the `Escape` key to exit the **Serial Communication** screen.

j. Press the `Escape` key to exit the **System BIOS Settings** screen.

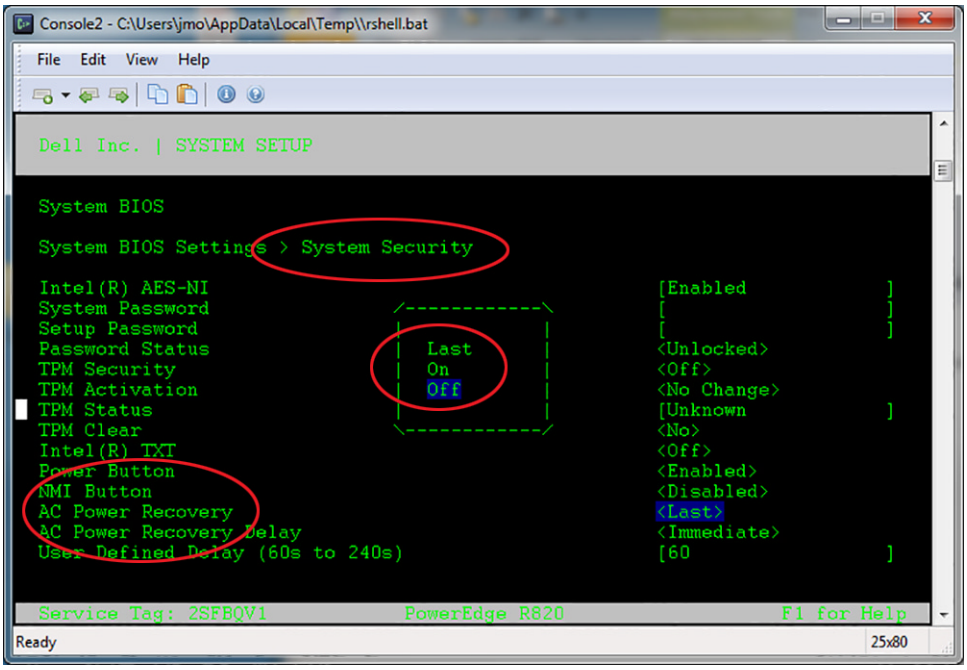
k. Press the `Escape` key to exit the **BIOS Settings** screen.

l. A `"Settings have changed"` message appears. Select **Yes** to save your changes.

m. A `"Settings saved successfully"` message appears. Select **OK**.

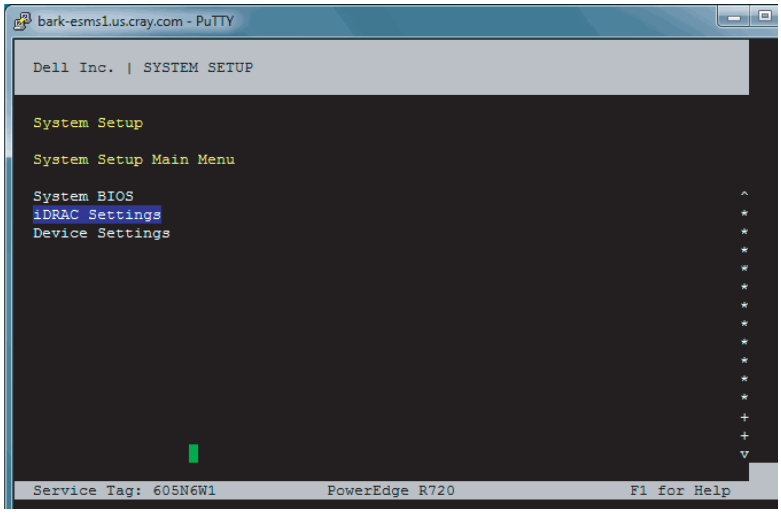
5. From the System BIOS settings menu, select **System Security**, and then **AC Power Recovery** to set the slave node to remain powered off after a system power failure. Cray recommends that the CIMS node power up and become operational before slave nodes.

Figure 78. Set Slave Node Auto-power On Setting to Off



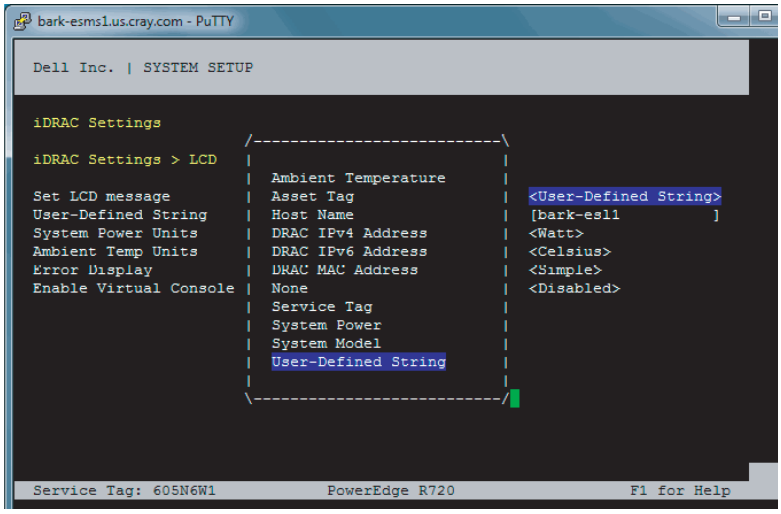
6. On the System Setup Main Menu, select **iDRAC Settings**, then press `Enter`. See Figure 79.

Figure 79. Dell 720 iDRAC BIOS Settings



7. Select **Network**, then press `Enter`. A long list of network settings is displayed.
8. Change the IPMI settings to enable the Serial Over LAN (SOL) console.
   a. Use the `down-arrow` key to scroll to the **IPMI SETTINGS** list.
   b. Ensure that **IPMI over LAN** is enabled.
      1) If necessary, select **IPMI over LAN**, then press `Enter`.
      2) In the pop-up window, select **<Enabled>**.
      3) Press `Enter` to return to the previous screen.
   c. Press the `Escape` key to exit the Network screen and return to the **iDRAC Settings** menu.
9. Change the LCD configuration to show the hostname in the LCD display.
   a. On the **iDRAC Settings** screen, use the `down-arrow` key to highlight **LCD**, then press `Enter`.
   b. Select **Set LCD message**. A pop-up window opens.
   c. In the pop-up window, select **User-Defined String**, then press `Enter`.
   d. Select **User-Defined String** (again), then press `Enter`. A text pop-up window opens for entering the new string. See Figure 80.

Figure 80. Dell 720 iDRAC BIOS LCD Settings



e. In the text pop-up window, enter the CLFS hostname (such as `esfs-mds001`), then press `Enter`.

f. Press the `Escape` key to exit the LCD screen.

g. Press the `Escape` key to exit the Network screen.

h. Press the `Escape` key to exit the **iDRAC Settings** screen.

i. A `"Settings have changed"` message appears. Select **Yes**, then press `Enter` to save your changes.

j. A `"Settings saved successfully"` message appears. Select **OK**, then press `Enter`.

10. Change the device settings so that the CLFS node can PXE boot on the CIMS administration network (`esmaint-net`).

    a. On the System Setup Main Menu, select **Device Settings**, then press `Enter`.

    b. In the **Device Settings** window, select **Integrated NIC 1 Port *N* ...**, then press `Enter`. The Main Configuration Page opens.

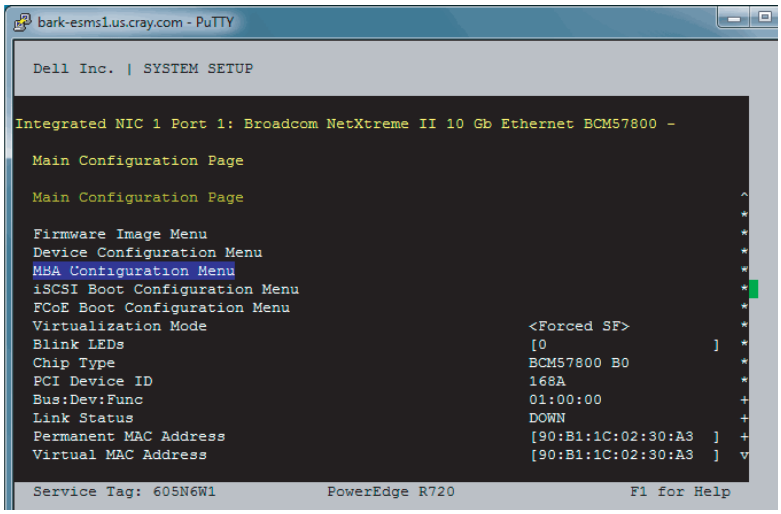
    Tip: Choose the NIC port number that corresponds to the Ethernet port for the `esmaint-net` network.

    - If `esmaint-net` uses the first Ethernet port (`eth0`), select **Integrated NIC 1 Port 1 ...**.
    - If `esmaint-net` uses the third Ethernet port (`eth2`), select **Integrated NIC 1 Port 3 ...**.

    Note: PXE booting must be disabled for the other three Ethernet ports.

c. On the Main Configuration Page screen, select **MBA Configuration Menu**, then press `Enter`. See Figure 81.

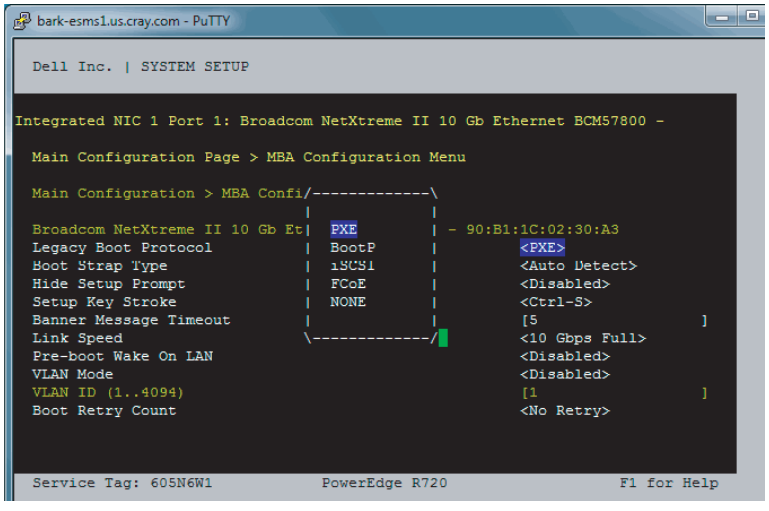
Figure 81. Dell 720 MBA Configuration Menu BIOS Settings



d. On the MBA Configuration Menu screen, select **Legacy Boot Protocol**, then press `Enter`. A pop-up window displays the available options.

e. In the pop-up window, use the `down-arrow` key to highlight **PXE**, then press `Enter`. See Figure 82.

Figure 82. Dell 720 Legacy Boot Protocol BIOS Settings



f. Press the `Escape` key to exit the MBA Configuration Menu screen.

g. Press the `Escape` key to exit the Main Configuration Page screen.

h. Verify that **Legacy Boot Protocol** is set to **None** for the other three Ethernet ports. If necessary, repeat step 10.b through step 10.g to change the setting for these three ports.

i. Press the `Escape` key to exit the **Device Settings** screen.

j. A `"Settings have changed"` message appears. Select **Yes**, then press `Enter` to save your changes.

k. A `"Settings saved successfully"` message appears. Select **OK**, then press `Enter`. The main screen (System Setup Main Menu) appears.

11. Save your changes and exit.

a. Press `Escape` to exit the System Setup Main Menu.

b. The utility displays the prompt "Are you sure you want to exit and reboot?" Select **Yes**.

Use Bright on the CIMS to configure a software image, networks, and other software for the CLFS node.