PRIMECLUSTER

Cluster Foundation (CF)
Configuration and Administration Guide 4.3
(Linux)

Edition October 2015

Copyright and Trademarks

Linux is a trademark or registered trademark of Mr. Linus Torvalds in the United States and other countries.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

EMC, PowerPath, and Symmetrix are registered trademarks of EMC Corporation.

TimeFinder and SRDF are trademarks of EMC Corporation.

PRIMECLUSTER is a registered trademark of Fujitsu Limited.

All other hardware and software names used are trademarks of their respective companies.

Export Controls

Exportation/release of this document may require necessary procedures in accordance with the regulations of your resident country and/or US export control laws.

Requests

- No part of this documentation may be reproduced or copied without permission of FUJITSU LIMITED.
- The contents of this documentation may be revised without prior notice.

Contents

1 Preface

The Cluster Foundation (CF) provides a comprehensive base of services that user applications and other PRIMECLUSTER services need to administrate and communicate in a cluster. These services include the following:

- Internode communications
- Node state management
- Cluster-wide configuration information
- Management and administration
- Distributed lock management

In addition, the foundation provides the following optional services:

- RCFS is a cluster-wide file share service
- RCVM is a cluster-wide volume management service

This document assumes that the reader is familiar with the contents of the PRIMECLUSTER *Concepts Guide* and that the PRIMECLUSTER software has been installed as described in the *Software Release Guide PRIMECLUSTER* and the *PRIMECLUSTER Installation Guide*.

1.1 Contents of this manual

This manual is organized as follows:

- The Chapter "Cluster Foundation" describes the administration and configuration of the Cluster Foundation.
- The Chapter "CF Registry and Integrity Monitor" discusses purpose and physical characteristics of the CF synchronized registry, and it discusses the purpose and implementation of the Cluster Integrity Monitor (CIM).
- The Chapter "Cluster resource management" discusses the database which is a synchronized clusterwide database holding information specific to several PRIMECLUSTER products.
- The Chapter "GUI administration" describes the administration features in the CF portion of the Cluster Admin graphical user interface (GUI).

- The Chapter "LEFTCLUSTER state" discusses the LEFTCLUSTER state, describes this state in relation to the other states, and discusses the different ways a LEFTCLUSTER state is caused.
- The Chapter "CF topology table" discusses the CF topology table as it relates to the CF portion of the Cluster Admin GUI.
- The Chapter "Shutdown Facility" describes the components and advantages of PRIMECLUSTER SF and provides administration information.
- The Chapter "CF over IP" discusses CF communications based on the use of interconnects.
- The Chapter "Diagnostics and troubleshooting" provides help for trouble-shooting and problem resolution for PRIMECLUSTER Cluster Foundation.
- The Chapter "CF messages and codes" provides a listing of messages and codes.
- The Chapter "Manual pages" lists the manual pages for PRIMECLUSTER.
- The Chapter "Release information" lists the manual pages for PRIME-CLUSTER.

1.2 PRIMECLUSTER documentation

The documents listed below provide details about PRIMECLUSTER products.

Books can be ordered via the Internet shop http://manualshop.ts.fujitsu.com/.

- "PRIMECLUSTER Concepts Guide"—Provides conceptual details on the PRIMECLUSTER family of products.
- "PRIMECLUSTER Installation and Administration Guide"—Provides instructions for installing and upgrading PRIMECLUSTER products on Linux.
- "PRIMECLUSTER Reliant Monitor Services (RMS) with Wizard Tools Configuration and Administration Guide"—Provides instructions for config-uring and administering PRIMECLUSTER Reliant Monitor Services using the Wizard Tools interface.
- "PRIMECLUSTER Reliant Monitor Services (RMS) Reference Guide"—Describes diagnostic procedures to solve RMS configuration problems, including how to view and interpret RMS log files. Provides a list of all RMS error messages with a probable cause and suggested action for each condition.

- "PRIMECLUSTER Global Disk Services Configuration and Administration Guide"—Provides information on configuring and administering Global Disk Services (GDS).
- "PRIMECLUSTER Global File Services Configuration and Administration Guide"—Provides information on configuring and administering Global File Services (GFS).
- "PRIMECLUSTER Global Link Services Configuration and Administration Guide: Redundant Line Control Function"—Provides information on configuring and administering the redundant line control function for Global Link Services (GLS).
- "PRIMECLUSTER Web-Based Admin View Operation Guide"—Provides information on using the Web-Based Admin View management GUI.
- "PRIMECLUSTER Messages"—Provides how to set the environment for PRIMECLUSTER. Messages output during system operation are also provided.
- "Software Release Guide" and "Installation Guide"—This document provides late-breaking information about installation, configuration, and operations for PRIMECLUSTER.

1.3 Conventions

In order to standardize the presentation of material, this manual uses a number of notational, typographical, and syntactical conventions.

1.3.1 Notation

This manual uses the following notational conventions.

1.3.1.1 Prompts

Command line examples that require system administrator (or root) privileges to execute are preceded by the system administrator prompt, the hash sign (#). Entries that do not require system administrator rights are preceded by a dollar sign ($).

In some examples, the notation *node*# indicates a root prompt on the specified node. For example, a command preceded by fuji3# would mean that the command was run as user root on the node named fuji3.

1.3.1.2 The keyboard

Keystrokes that represent nonprintable characters are displayed as key icons such as [Enter] or [F1]. For example, [Enter] means press the key labeled *Enter*; [Ctrl-b] means hold down the key labeled *Ctrl* or *Control* and then press the [B] key.

1.3.1.3 Typefaces

The following typefaces highlight specific elements in this manual.

Typeface	Usage
Constant Width	Computer output and program listings; commands, file names, manual page names and other literal programming elements in the main body of text.
Italic	Variables in a command line that you must replace with an actual value. May be enclosed in angle brackets to emphasize the difference from adjacent text; for example, <*nodename*>RMS; unless directed otherwise, you should not enter the angle brackets. The name of an item in a character-based or graphical user interface. This may refer to a menu item, a radio button, a checkbox, a text input box, a panel, or a window title.
Bold	Items in a command line that you must type exactly as shown.

Typeface conventions are shown in the following examples.

1.3.1.4 Example 1

Several entries from an /etc/passwd file are shown below:

```
bin:x:1:1:bin:/bin:/bin/bash
daemon:x:2:2:daemon:/sbin:/bin/bash
lp:x:4:7:lp daemon:/var/spool/lpd:/bin/bash
```

1.3.1.5 Example 2

To use the `cat` command to display the contents of a file, enter the following command line:

$ **cat** *file*

1.3.2 Command syntax

The command syntax observes the following conventions.

Symbol	Name	Meaning
[ ]	Brackets	Enclose an optional item.
{ }	Braces	Enclose two or more items of which only one is used. The items are separated from each other by a vertical bar (|).
|	Vertical bar	When enclosed in braces, it separates items of which only one is used. When not enclosed in braces, it is a literal element indicating that the output of one program is piped to the input of another.
( )	Parentheses	Enclose items that must be grouped together when repeated.
...	Ellipsis	Signifies an item that may be repeated. If a group of items can be repeated, the group is enclosed in parentheses.

1.4 Important notes and cautions

Important

Indicates important information.

Caution

Indicates a situation that can cause harm to data.

Note

Indicates information that needs special attention.

1.5 Abbreviations

Oracle Solaris might be described as Solaris, Solaris Operating System, or Solaris OS.

1.6 Revision history

Revision	Location	Manual code
Deleted the explanations of the following commands: - hvdist - hvgdmake - hvrclev - hvreset - hvthrottle	12.8	J2UZ-5284-04ENZ0(01)
Changed the description about the cluster Resource Database.	4.1	J2UZ-5284-04ENZ0(02)

2 Cluster Foundation

This chapter describes the administration and configuration of the Cluster Foundation (CF).

This chapter discusses the following:

- The Section "CF, CIP, and CIM configuration" describes CF, Cluster Interconnect Protocol (CIP) and Cluster Integrity Monitor (CIM) configuration that must be done prior to other cluster services.
- The Section "CIP configuration file" describes the format of the CIP configuration file.
- The Section "Cluster Configuration Backup and Restore (CCBR)" details a method to save and restore PRIMECLUSTER configuration information.

2.1 CF, CIP, and CIM configuration

You must configure CF before any other cluster services, such as Reliant Monitor Services (RMS). CF defines which nodes are in a given cluster. In addition, after you configure CF and CIP, the Shutdown Facility (SF) and RMS can be run on the nodes.

The Shutdown Facility (SF) is responsible for node elimination. This means that even if RMS is not installed or running in the cluster, missing CF heartbeats will cause SF to eliminate nodes.

You can use the Cluster Admin CF Wizard to easily configure CF, CIP, and CIM for all nodes in the cluster, and you can use the Cluster Admin SF Wizard to configure SF.

A CF configuration consists of the following main attributes:

- Cluster name—This can be any name that you choose as long as it is 31 characters or less per name and each character comes from the set of printable ASCII characters, excluding white space, newline, and tab characters. Cluster names are always mapped to upper case.
- Set of interfaces on each node in the cluster used for CF networking—An Ethernet device on the local node is one example of an interface.
- CF node name—By default, in Cluster Admin, the CF node names are the same as the Web-Based Admin View names; however, you can use the CF Wizard to change them. CF node names are converted to lower case.

The dedicated network connections used by CF are known as interconnects. They typically consist of some form of high speed networking such as 100 MB or Gigabit Ethernet links. These interconnects must meet the following requirements if they are to be used for CF:

- The network links used for interconnects must have low latency and low error rates. This is required by the CF protocol. Private switches and hubs will meet this requirement. Public networks, bridges, and switches shared with other devices may not necessarily meet these requirements, and their use is not recommended.

  It is recommended that each CF interface be connected to its own private network with each interconnect on its own switch or hub.

- The interconnects should not be used on any network that might experience network outages of 5 seconds or more. A network outage of 10 seconds will, by default, cause a route to be marked as DOWN. cfset(1M) can be used to change the 10 second default. Refer to the Section "cfset" for additional information.

  Since CF automatically attempts to bring up downed interconnects, the problem with split clusters only occurs if all interconnects experience a 10-second outage simultaneously. Nevertheless, CF expects highly reliable interconnects.

CF can also be run over IP. Any IP interface on the node can be chosen as an IP device, and CF will treat this device much as it does an Ethernet device. However, all the IP addresses for all the cluster nodes on that interconnect must be on the same IP subnetwork, and their IP broadcast addresses must be the same (refer to the Chapter "CF over IP" for more information).

The IP interfaces used by CF must be completely configured by the System Administrator before they are used by CF. You can run CF over both Ethernet devices and IP devices.

Higher level services, such as RMS, SF, Global File Services (GFS), and so forth, will not notice any difference when CF is run over IP.

You should carefully choose the number of interconnects you want in the cluster before you start the configuration process. If you decide to change the number of interconnects after you have configured CF across the cluster, you can either bring down CF on each node to do the reconfiguration or use the cfrecon command. Bringing down CF requires that higher level services, like RMS, SF, and applications, be stopped on that node, so the reconfiguration process is neither trivial nor unobtrusive. Using the cfrecon command will lead to temporary asymmetrical CF configuration.

Your configuration should specify at least two interconnects to avoid a single point of failure in the cluster.

Before you begin the CF configuration process, ensure that all of the nodes are connected to the interconnects you have chosen and that all of the nodes can communicate with each other over those interconnects. For proper CF configuration using Cluster Admin, all of the interconnects should be working during the configuration process.

CIP configuration involves defining virtual CIP interfaces and assigning IP addresses to them. Up to eight CIP interfaces can be defined per node. These virtual interfaces act like normal TCP/IP interfaces except that the IP traffic is carried over the CF interconnects. Because CF is typically configured with multiple interconnects, the CIP traffic will continue to flow even if an interconnect fails. This helps eliminate single points of failure as far as physical networking connections are concerned for intracluster TCP/IP traffic.

Except for their IP configuration, the eight possible CIP interfaces per node are all treated identically. There is no special priority for any interface, and each interface uses all of the CF interconnects equally. For this reason, many system administrators may chose to define only one CIP interface per node.

To ensure that you can communicate between nodes using CIP, the IP address on each node for a specific CIP interface should use the same subnet. Besides, if you use an IPv6 address, use the IPv6 address assigned to the CIP interface for communications. Communications using the link local address are not available.

CIP traffic is really intended only to be routed within the cluster. The CIP addresses should not be used outside of the cluster. Because of this, you should use addresses from the non-routable reserved IP address range.

For the IPv4 address, Address Allocation for Private Internets (RFC 1918) defines the following address ranges that are set aside for private subnets:

Subnets(s)	Class	Subnetmask
10.0.0.0	A	255.0.0.0
172.16.0.0 ... 172.31.0.0	B	255.255.0.0
192.168.0.0 ... 192.168.255.0	C	255.255.255.0

For the IPv6 address, the range where Unique Local IPv6 Unicast Addresses (RFC 4193) defined with the prefix FC00::/7 is used as the address (Unique Local IPv6 Unicast Addresses) which can be allocated freely within the private network.

For CIP nodenames, it is strongly recommended that you use the following convention for RMS:

*cfname*RMS

cfname is the CF name of the node and RMS is a literal suffix. This will be used for one of the CIP interfaces on a node. This naming convention is used in the Cluster Admin GUI to help map between normal node names and CIP names. In general, you only need to configure one CIP interface per node.

In the CIP configuration, CIP names are stored in /etc/hosts. /etc/nsswitch.conf(4) should be set to use files as the first criteria when looking up nodes.

The recommended way to configure CF, CIP and CIM is to use the Cluster Admin GUI. You can use the CF/CIP Wizard in the GUI to configure CF, CIP, and CIM on all nodes in the cluster in just a few screens. Before running the wizard, however, you must complete the following steps:

1. CF/CIP, Web-Based Admin View, and Cluster Admin should be installed on all nodes in the cluster.
2. If you are running CF over Ethernet, then all of the interconnects in the cluster should be physically attached to their proper hubs or networking equipment and should be working.
3. If you are running CF over IP, then all interfaces used for CF over IP should be properly configured and be up and running. See Chapter "CF over IP" for details.
4. Web-Based Admin View configuration must be done. Refer to "2.4.1 Management server configuration" in the *Web-Based Admin View Operation Guide* for details.

In the *cf* tab in Cluster Admin, make sure that the CF driver is loaded on that node. Press the *Load Driver* button if necessary to load the driver. Then press the *Configure* button to start the CF Wizard.

The CF/CIP Wizard is invoked by starting the GUI on a node where CF has not yet been configured. When this is done, the GUI automatically brings up the CF/CIP Wizard in the *cf* tab of the GUI. You can start the GUI by entering the following URL with a browser running the correct version of the Java plug-in:

http://*management_server***:8081/Plugin.cgi**

management_server is the primary or secondary management server you configured for this cluster. Refer to "4.3.3.1 Initial setup of the operation management server" in the "PRIMECLUSTER *Installation and Administration Guide*" for details on configuring the primary and secondary management servers. Refer to "3.1.2 Prerequisite client environment" in the "PRIMECLUSTER *Web-Based Admin View Operation Guide*" on which browsers and Java plug-ins are required for the Cluster Admin GUI.

In PRIMECLUSTER, it is recommended that you configure the administrative LAN and cluster interconnects on different NICs. However, if you cannot make such a configuration due to restrictions on hardware in a KVM environment, the configuration which shares the administrative LAN and cluster interconnects on the same NIC is also supported.

In the configuration which shares the administrative LAN and cluster interconnects on the same NIC, you must conform all the following conditions for network and GLS:

- Make two NICs redundant by GLS Virtual NIC mode on the Host OS.
- Create the necessary number of the VLAN interfaces for the Host OS, the administrative LAN for the Host OS, public LAN, and cluster interconnects on the virtual interface.
- Create cluster interconnects for the Host OS and guest OS on their VLAN interfaces. They are not made redundant on the cluster interconnect side.
- For the public LAN, create Gls resources on the guest OS and RMS on the guest OS monitors them.

This configuration requires the CF configuration by CLI. For the configuration method, see "Example of CF configuration by CLI".

In this configuration, there are the following notes:

Note

- Availability in the event of a double failure of network switch
  If both network switches where two NICs are connected fail, the administrative LAN, public LAN, and cluster interconnects will enter the fault state. In this state, the Host OS and guest OS cannot be forcibly stopped and no switchover of applications occur.
  Note that if a double failure occurs on the NIC of a server, switchover of applications occurs because they can be forcibly stopped from the other server.

- Restriction on the timeout value of cluster interconnects
  In GLS Virtual NIC mode, it takes 20 seconds to switch a path. On the other hand, the time to detect the failure of cluster interconnects is 10 seconds (default value). Therefore, with the default value, the failure of cluster interconnects will be detected first if one NIC failure occurs. To solve this problem, change the timeout value (CLUSTER_TIMEOUT) to 40 seconds for the Host OS and 30 seconds for the guest OS.
  By this setting change, the time to detect failures of cluster interconnects will be longer (from 10 seconds to 40 seconds).

- Cluster switchover due to overload of the public LAN
  If a communication timeout which is more than 30 seconds occurs, PRIMECLUSTER detects a failure of cluster interconnects, forcibly stops the Host OS or guest OS, and a cluster switchover may occur.

2.1.1 Differences between CIP and CF over IP

Although the two terms "CF over IP" and "CIP" (also known as "IP over CF") sound similar, they are completely different functions.

In general, the cluster interconnects are separate from the public network and not used by the TCP/IP stack. To allow applications to use TCP/UPD protocols on top of CF, CF uses the CIP driver.

CIP routes the TCP/IP traffic through the cluster interconnects to the other nodes in the cluster. CIP uses the interfaces configured for CF and does failover and load balancing if multiple interfaces are available.

CIP defines a reliable IP interface for applications on top of the Cluster Foundation (CF) (Figure 1).

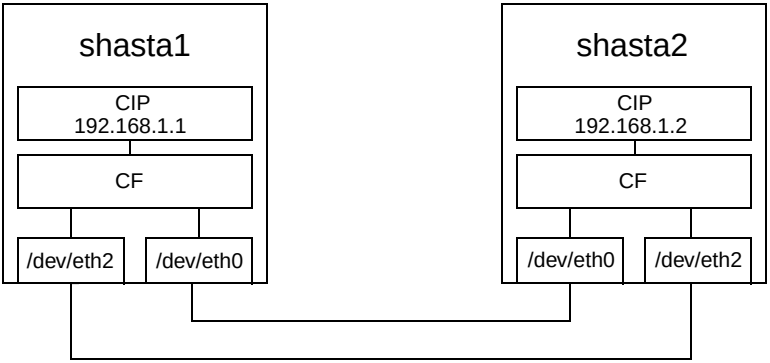

Figure 1: CIP diagram

CIP should not be used in a CF over IP configuration. Instead of creating an additional CIP address for applications like RMS, take the IP address assigned to CF directly.

CF over IP is used for configurations where the nodes are separated by long distances and where standard Ethernet wiring cannot be used. This can be due to the maximum cable length or different segments connected through routers. In this case, CF sends its protocol messages through the IPv4 stack instead of using the low-level network interfaces. This is not operated on IPv6. The IP stack needs to be configured up front and must be available before you start the cluster.

The CF node discovery is done by a JOIN message. This message is sent to the configured destination address. This can be the address of the remote node or the broadcast address of this subnet. Broadcast messages do not travel across routers and subnets. If the remote node is on a different subnet the broadcast address cannot be used or the cluster will not join.

It is recommended to separate the CF networks from the public network. Up to four IP devices can be configured for CF but the best way would be to configure only one IP device and let the IP layer deal with multi path or bonding.

To configure the IP interface over the private interconnect, use an IP address designed for the private network, such as in the following example:

192.168.0.*x*

x is an integer between 1 and 254.

During the cluster joining process, CF sends broadcast messages to other nodes; therefore, all the nodes must be on the same local network. If one of the nodes is on a different network or subnet, the broadcast will not be received by that node. Therefore, the node will fail to join the cluster.

The following are possible scenarios for CF over IP:

- Where the cluster spans over two Ethernet segments of the same sub network. Each sub-level Ethernet protocol is not forwarded across the router but does pass IP traffic.
- When you need to reach beyond the physical cable length. Regular Ethernet is limited to the maximum physical length of the cable. Distances that are longer than the maximum cable length cannot be reached.

Use CF with the Ethernet link-level connection whenever possible because CF over IP implies additional network/protocol information and usually will not perform as well (Figure 2).

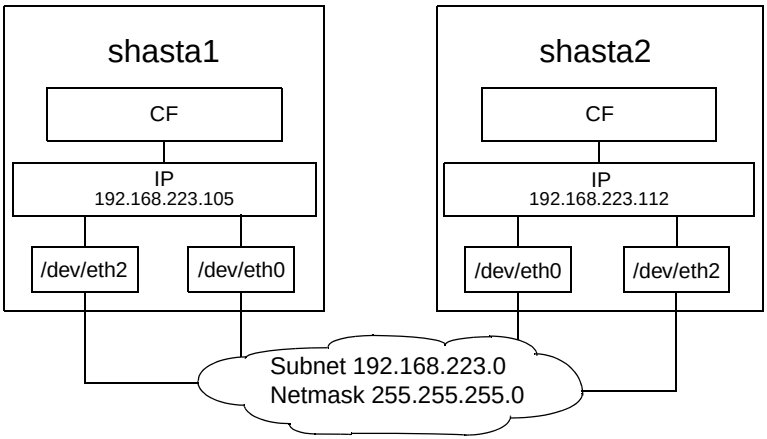

Figure 2: CF over IP diagram

2.1.2 cfset

The cfset(1M) utility can be used to set certain tunable parameters in the CF driver. The values are stored in /etc/default/cluster.config. The cfset(1M) utility can be used to retrieve and display the values from the kernel or the file as follows:

- A new file under /etc/default called cluster.config is created.

- The values defined in /etc/default/cluster.config can be set or changed using the GUI (for cfcp and cfsh during initial cluster configuration) or by using a text editor.
- The file consists of the following tupple entries, *Name* and *Value*:

  Name:

  – This is the name of a CF configuration parameter. It must be the first token in a line.

  – Maximum length for *Name* is 31 bytes. The name must be unique.

  – Duplicate names will be detected and reported as an error when the entries are applied by cfconfig -l and by the cfset(1M) utility (cfset -r and -f option). This will log invalid and duplicate entries to /var/log/messages.

  – cfset(1M) can change the *Value* for the *Name* in the kernel if the driver is already loaded and running.

Value:

- This represents the value to be assigned to the CF parameter. It is a string, enclosed in double quotes or single quotes. Maximum length for *Value* is 4 kilobytes (4K) characters.
- New lines are not allowed inside the quotes.
- A new line or white space marks the close of a token.
- However, if double quotes or single quotes start the beginning of the line, cfset treats the line as a continuation value from the previous value.

- The maximum number of *Name*/*Value* pair entries is 100.
- The hash sign (#) is used for the comment characters. It must be the first character in the line, and it causes the entries on that line to be ignored.
- Single quotes can be enclosed in double quotes or vice versa.

cfset(1M) options are as follows:

cfset [-r | -f | -a | -o *name* | -g *name* | -h]

Refer to the Chapter "Manual pages" and to the cfset(1M) manual page for more details on options.

The tuneables are as follows:

- CLUSTER_TIMEOUT (refer to example that follows)
- CFSH (refer to the following Section "CF security")
- CFCP (refer to the following Section "CF security")
- CLUSTER_IP_TTL (refer to example 2 that follows)

  This parameter becomes enabled when using CF over IP.

  Tunable Description: This is the value of the TTL field in the IP-header for all CF packets.

  Default value: 64

  Valid values: 1-255

- CLUSTER_IP_CTRL_TOS (refer to example 2 that follows)

  This parameter becomes enabled when using CF over IP.

  Tunable Description: This is the value of the TOS 8-bit field in the IP-header for all CF control packets. This includes cluster heartbeat packets. The default value shown below is a best-fit default which sets the 6-bit DSCP field

to binary 100010. This is a DSCP forwarding 4F AF class (compatible with older IP precedence) and specifies the lowest AF drop precedence (least likely to be dropped when congestion is encountered).

Default value: 0x88 (136)

Valid values: 0-255

- CLUSTER_IP_DATA_TOS (refer to example 2 that follows)

  This parameter becomes enabled when using CF over IP.

  Tunable Description: This is the value of the TOS 8-bit field in the IP-header for all CF data packets (non-control packets). The default value shown below is a best-fit default which sets the 6-bit DSCP field to binary 001010. This is a DSCP forwarding 1F AF class (compatible with older IP precedence) and specifies the lowest AF drop precedence (least likely to be dropped when congestion is encountered).

  Default value: 0x28 (40)

  Valid values: 0-255

After any change to cluster.config, run the cfset(1M) command as follows:

 # cfset -r

Example 1

Use cfset(1M) to tune timeout as follows:

CLUSTER_TIMEOUT "30"

This changes the default 10-second timeout to 30 seconds. The minimum value is 1 second. There is no maximum. It is strongly recommended that you use the same value on all cluster nodes.

CLUSTER_TIMEOUT represents the number of seconds that one cluster node waits for a heartbeat response from another cluster node. Once CLUSTER_TIMEOUT seconds has passed, the non-responding node is declared to be in the LEFTCLUSTER state. The default value for CLUSTER_TIMEOUT is 10, which experience indicates is reasonable for most PRIMECLUSTER installations. We allow this value to be tuned for exceptional situations, such as networks which may experience long switching delays.

Example 2

To set tuneables to non-default values:

1. Edit the /etc/default/cluster.config file and add entries for each tunable:

   CLUSTER_IP_TTL 64

   CLUSTER_IP_CTRL_TOS 0x88

   CLUSTER_IP_DATA_TOS 0x28

2. Run cfset -f to verify settings in file.
3. Run cfset -r to load new values to CF.
4. Run cfset -a to verify values in kernel

2.1.3 CF security

PRIMECLUSTER includes the following facilities for cluster communications if you do not want to use .rhosts:

- cfcp/cfsh
- sshconf (not supported by Wizard Tools)

These tools are provided to allow cluster configuration in an environment which does not permit rsh and rcp. They are specialized utilities that do not provide all the functionality of rsh and rcp and are not intended as replacements.

2.1.3.1 cfcp/cfsh

CF includes the ability to allow cluster nodes to execute commands on another node (cfsh) and to allow cluster nodes to copy files from one node to another (cfcp). However, this means that your cluster interconnects must be secure since any node that can join the cluster has access to these facilities. Because of this, these facilities are disabled by default.

PRIMECLUSTER 4.1 and higher offers a chance to configure these facilities. As one of the final steps of the CF Configuration Wizard in the Cluster Adm GUI, there are two checkboxes. Checking one enables remote file copying and checking the other enables remote command execution.

To enable remote access using cfcp/cfsh, set the following parameters in cluster.config :

```
CFCP "cfcp"
CFSH "cfsh"
```

To deactivate, remove the settings from the /etc/default/cluster.config file and run cfset -r.

Refer to the Section "cfset" in this chapter for more information.

2.1.3.2 sshconf

You can use the sshconf tool to set up non-interactive ssh access among a list of nodes. Running sshconf is similar to setting up the .rhosts file for rsh.

sshconf uses the RSA authentication method and protocol version 2. If it exists, sshconf uses the default authentication key $HOME/.ssh/id_rsa, or it creates the key it if does not already exist.

To operate, sshconf needs /bin/bash to exist on all nodes.

Examples of the sshconf tool are as follows:

- Enable one way access between nodes:

  fuji2# **sshconf fuji3 fuji4 fuji5**

  Running this command on fuji2 sets up one way ssh access from fuji2 to fuji3, fuji4, and fuji5 respectively.

- Disable one-way access to a node:

  fuji2# **sshconf -d fuji3 fuji4 fuji5**

  Running this command on fuji2 disables ssh access from fuji2 to fuji3, fuji4, and fuji5. This means that fuji2 does not have ssh access to fuji3, fuji4, and fuji5; however, fuji3, fuji4, and fuji5 still have the same ssh access as before running the command.

- Enable two-way access without password:

  fuji2# **sshconf -c fuji3 fuji4 fuji5**

  Running this command on fuji2 sets up ssh access among fuji3, fuji4, and fuji5 without being asked for a password. Note that fuji2 (where the command is run) is not automatically included. fuji2 only has one-way ssh access to fuji3, fuji4, and fuji5.

The Wizard Tools does not support ssh.

2.1.4 Signed applets

Cluster Admin uses Java applets. The main advantage of trusting signed applets is that Cluster Admin can use the client system's resources. For example, you can copy and paste messages from the Java window into other applications.

When Cluster Admin is first started, a Java security warning dialog allows you to choose the security level for the current and future sessions.

2.1.5 Example of creating a cluster

The following example shows what the Web-Based Admin View and Cluster Admin screens would look like when creating a two-node cluster. The nodes involved are named fuji2 and fuji3, and the cluster name is FUJI.

This example assumes that Web-Based Admin View configuration has already been done. fuji2 is assumed to be configured as the primary management server for Web-Based Admin View, and fuji3 is the secondary management server.

The first step is to start Web-Based Admin View by entering the following URL in a java-enabled browser:

http://*Management_Server***:8081/Plugin.cgi**

fuji2 is a management server. Enter the following:

http://fuji2:8081/Plugin.cgi

After a few moments, a login pop-up appears asking for a user name and password (Figure 3).

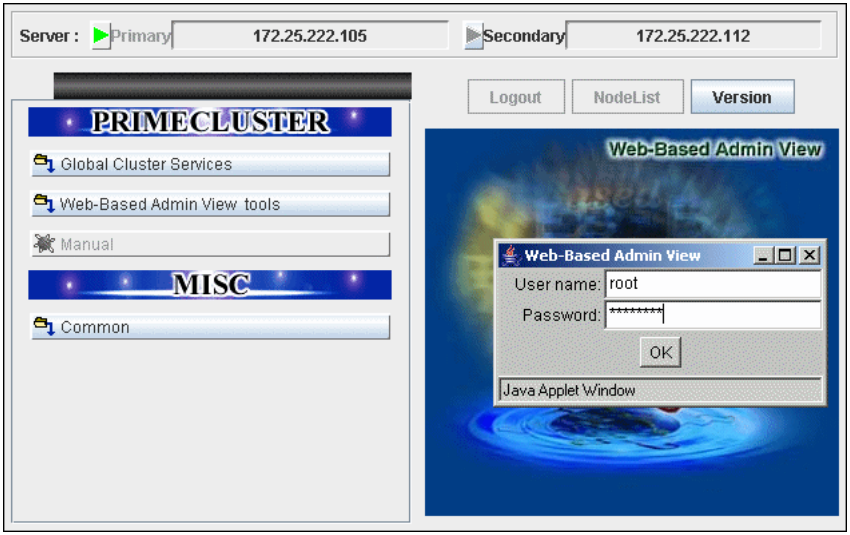

Figure 3: Login pop-up

Since you will be running the Cluster Admin CF Wizard, which does configuration work, you will need a privileged user ID such as root. There are three possible categories of users with sufficient privilege:

- The user root—You can enter root for the user name and root's password on fuji2. The user root is always given the maximum privilege in Web-Based Admin View and Cluster Admin.
- A user in group clroot—You can enter the user name and password for a user on fuji2 who is part of the UNIX group clroot. This user will have maximum privilege in Cluster Admin, but will be restricted in what Web-Based Admin View functions they can perform. This should be fine for CF configuration tasks.
- A user in group wvroot—You can enter the user name and password for a user on fuji2 who is part of the UNIX group wvroot. Users in wvroot have maximum Web-Based Admin View privileges and are also granted maximum Cluster Admin privileges.

For further details on Web-Based Admin View and Cluster Admin privilege levels, refer to "4.3.1 Assigning Users to Manage the Cluster" in the *PRIMECLUSTER Installation and Administration Guide*.

After clicking on the *OK* button, the top menu appears (Figure 4). Click on the button labeled *Global Cluster Services*.

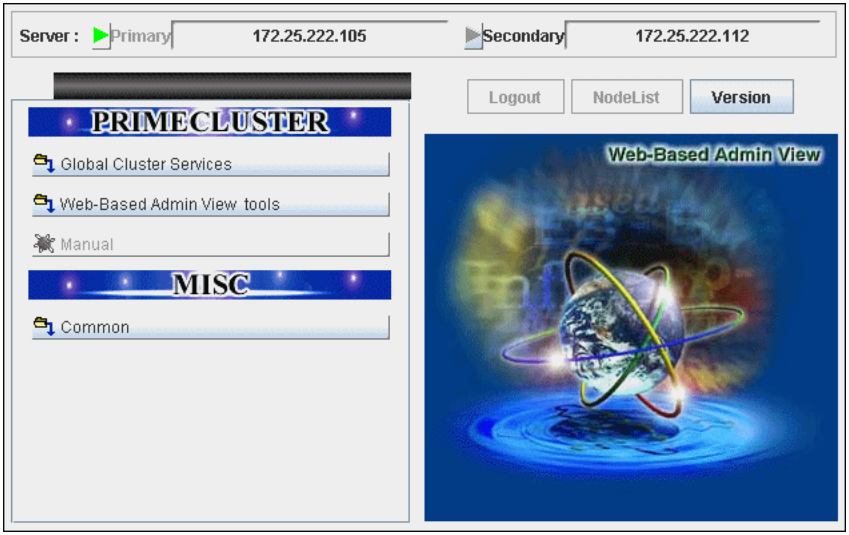

Figure 4: Main Web-Based Admin View window after login

The Cluster Admin selection window appears (Figure 5).

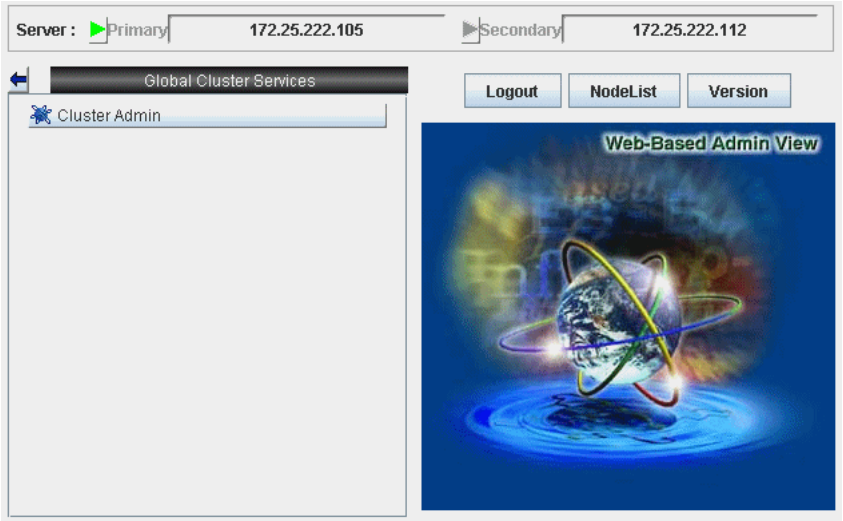

Figure 5: Global Cluster Services window in Web-Based Admin View

Click on the button labeled *Cluster Admin* to launch the Cluster Admin GUI.

The *Choose a node for initial connection* window appears (Figure 6).

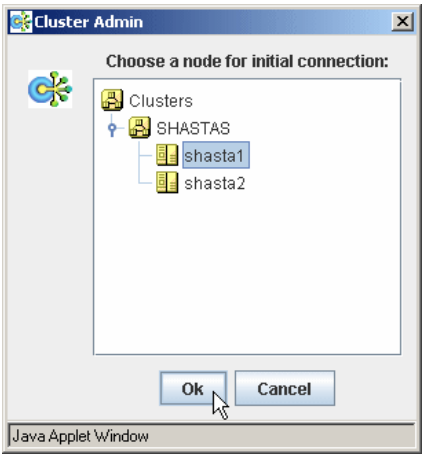

Figure 6: Initial connection pop-up

The *Choose a node for initial connection* window (Figure 6) lists the nodes that are known to the Web-Based Admin View management station. If you select a node where CF has not yet been configured, then Cluster Admin will let you run the CF Wizard on that node.

In this example, neither fuji2 nor fuji3 have had CF configured, so either would be acceptable as a choice. In Figure 6, fuji2 is selected. Clicking on the *OK* button causes the main Cluster Admin GUI to appear. Since CF is not configured on fuji2, a window similar to Figure 7 appears.

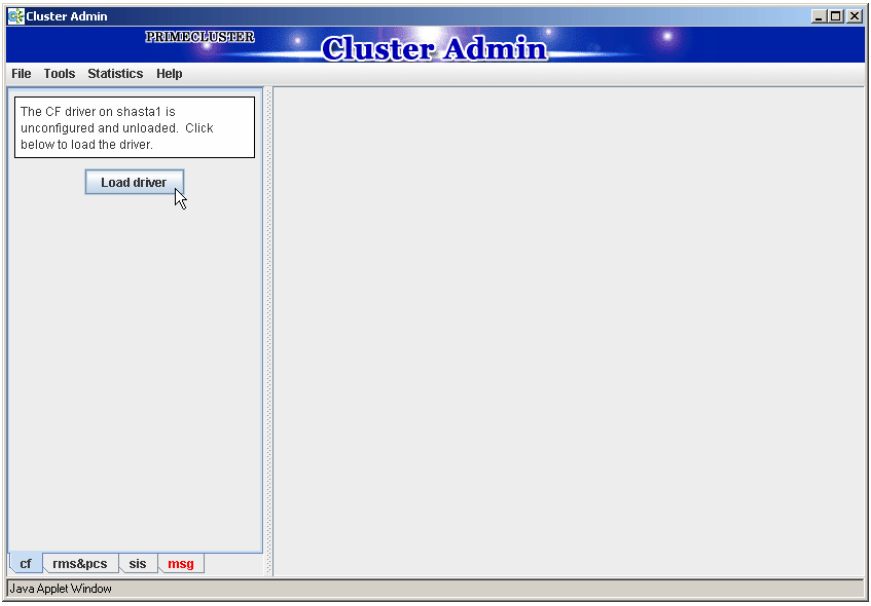

Figure 7: CF is unconfigured and unloaded

Click on the *Load driver* button to load the CF driver.

A window indicating that CF is loaded but not configured appears (Figure 8).

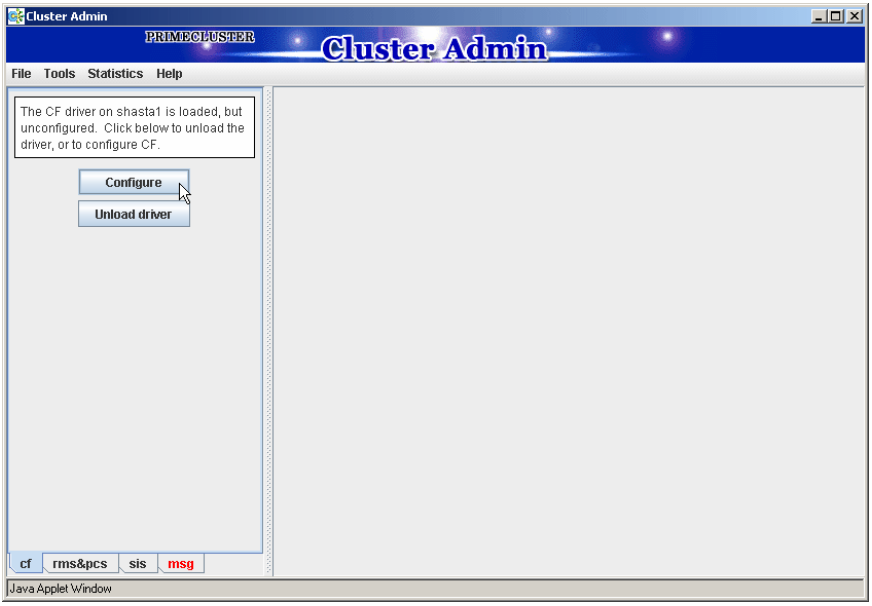

Figure 8: CF loaded but not configured

Click on the *Configure* button to bring up the CF Wizard. The CF Wizard scans for existing clusters.

After the CF Wizard finishes looking for clusters, a window similar to Figure 9 appears.

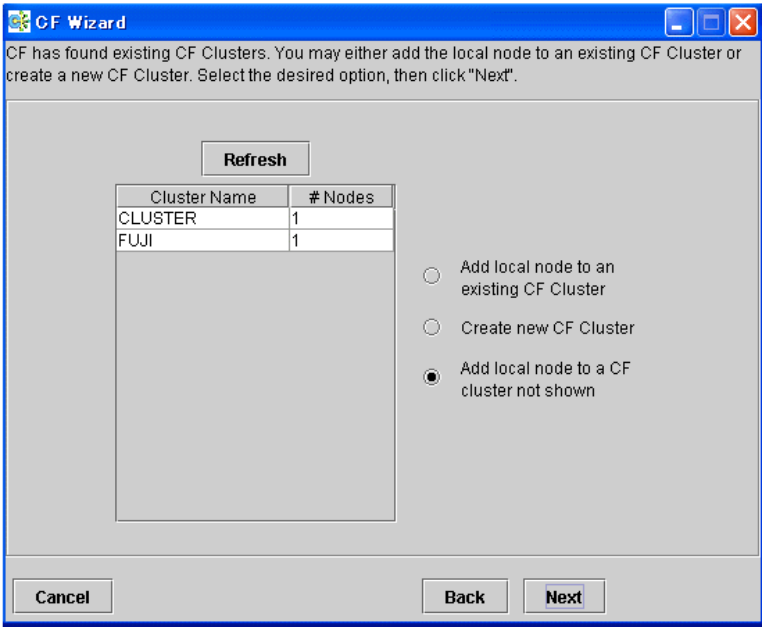

Figure 9: Creating or joining a cluster

This window lets you decide if you want to join an existing cluster or create a new one.

A pure CF over IP cluster will not show up in the *Cluster Name* column. To join a CF over IP cluster, select the *Add local node to a CF cluster not shown* radio button and click *Next*.

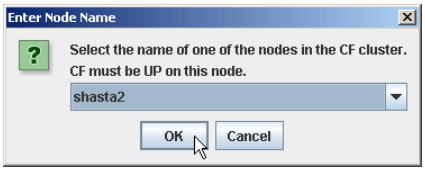

Figure 10: Adding a local node to a CF cluster not shown

Enter the node name of the CF cluster that you want to join. Click *OK* to proceed (Figure 10). After scanning the node and retrieving the existing cluster's details, the CF wizard takes you to the window for joining an existing cluster.

To create a new cluster, select that the *Create new CF Cluster* radio button as shown in Figure 9. Then, click *Next*.

The window for creating a new cluster or for joining an existing cluster appears, depending on your previous selection. Figure 11 shows the window for creating a new cluster. The window for joining an existing cluster is very similar, except you cannot change the cluster name.

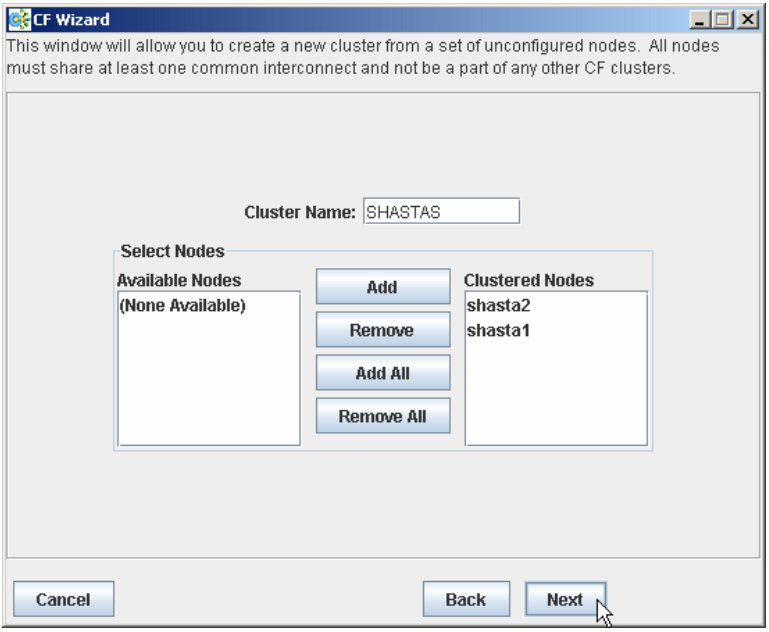

Figure 11: Selecting cluster nodes and the cluster name

This window lets you chose the cluster name and also determine what nodes will be in the cluster. In the example above, we have chosen FUJI for the cluster name.

Below the cluster name are two boxes. The one on the right, under the label *Clustered Nodes*, contains all nodes that you want to become part of this CF cluster. The box on the left, under the label *Available Nodes*, contains all the other nodes known to the Web-Based Admin View management server. You should select nodes in the left box and move them to the right box using the *Add* or *Add All* button. If you want all of the nodes in the left box to be part of the CF cluster, then just click on the *Add All* button.

If you get to this window and you do not see all of the nodes that you want to be part of this cluster, then there is a very good chance that you have not configured Web-Based Admin View properly. When Web-Based Admin View is initially installed on the nodes in a potential cluster, it configures each node as if it were a primary management server independent of every other node. If no additional Web-Based Admin View configuration were done, and you started up Cluster Admin on such a node, then Figure 11 would show only a single node in the right-hand box and no additional nodes on the left-hand side. If you see this, then it is a clear indication that proper Web-Based Admin View configuration has not been done.

Refer to "4.3 Preparations for Starting the Web-Based Admin View Screen" in the *PRIMECLUSTER Installation and Administration Guide* for more details on Web-Based Admin View configuration.

After you have chosen a cluster name and selected the nodes to be in the CF cluster, click on the *Next* button.

The CF Wizard then loads CF on all the selected nodes and does CF pings to determine the network topology. While this activity is going on, a window similar to Figure 12 appears.

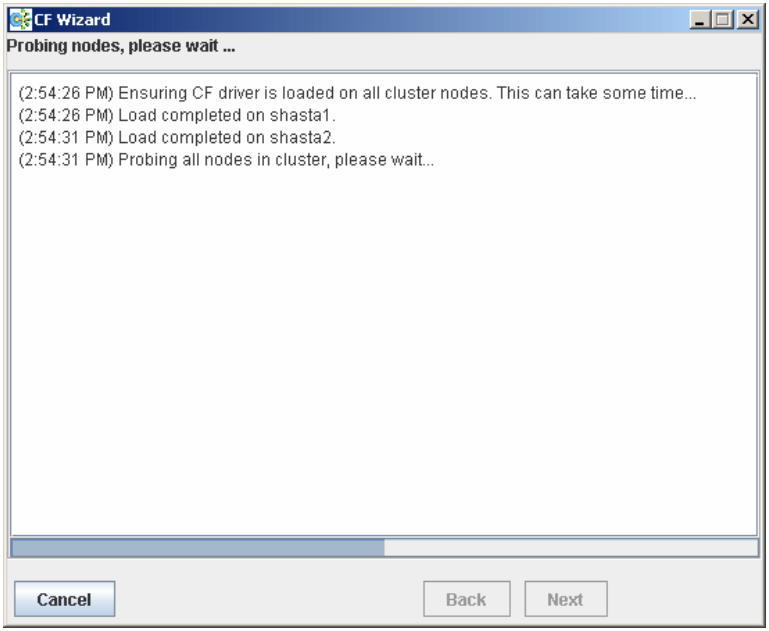

Figure 12: CF loads and pings

Usually, loading the CF driver is a relatively quick process. However, on some systems which use large disk arrays, the first CF load can take several minutes.

The window that allows you to edit the CF node names for each node appears (Figure 13). By default, the CF node names, which are shown in the right-hand column, are the same as the Web-Based Admin View names which are shown in the left-hand column.

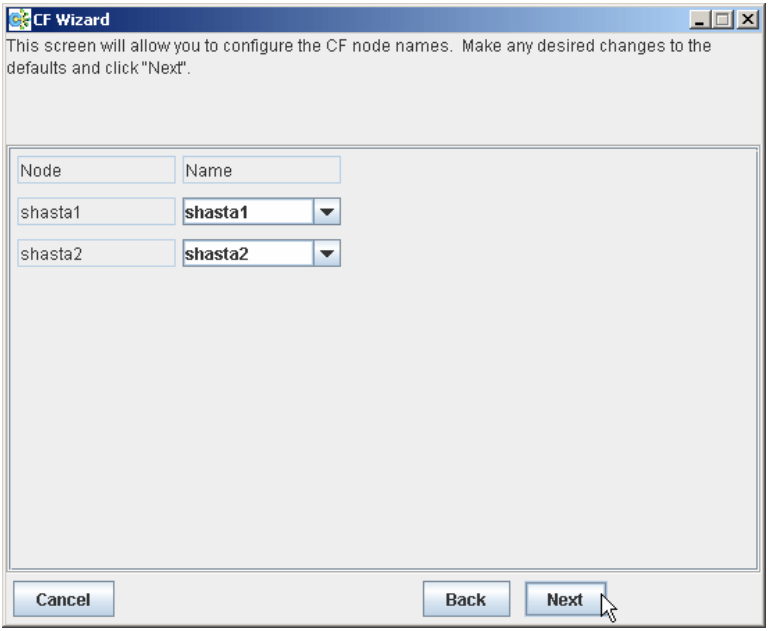

Figure 13: Edit CF node names

Make any changes to the CF node name and click *Next*.

After the CF Wizard has finished the loads and the pings, the CF topology and connection table appears (Figure 14).

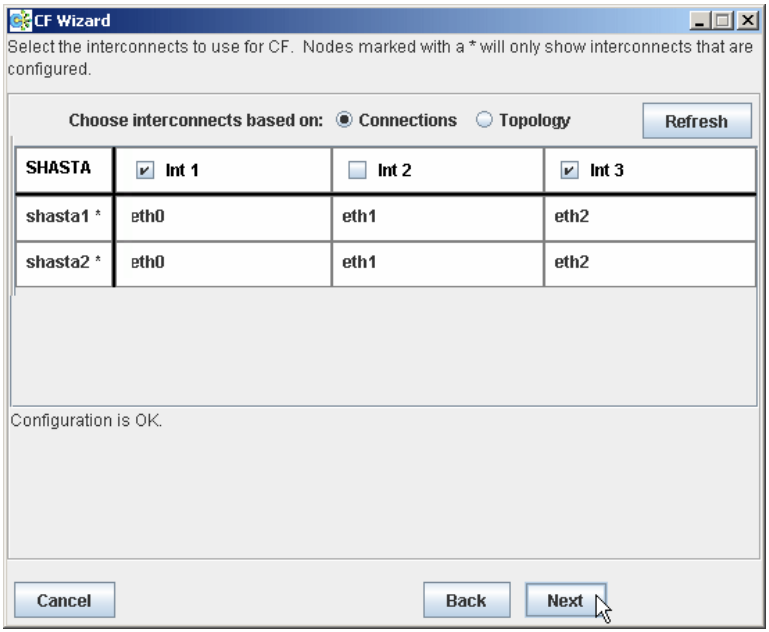

Figure 14: CF topology and connection table

Before using the CF topology and connection table in Figure 14, you should understand the following terms:

- Full interconnect—An interconnect where CF communication is possible to all nodes in the cluster.
- Partial interconnect—An interconnect where CF communication is possible between at least two nodes, but not to all nodes. If the devices on a partial interconnect are intended for CF communications, then there is a networking or cabling problem somewhere.
- Unconnected devices—These devices are potential candidates for CF configuration, but are not able to communicate with any other nodes in the cluster.

The CF Wizard determines all the full interconnects, partial interconnects, and unconnected devices in the cluster using CF pings. If there are one or more full interconnects, then it will display the connection table shown in Figure 14.

Connections table

The connection table lists all full interconnects. Each column with an Int header represents a single interconnect. Each row represents the devices for the node whose name is given in the left-most column. The name of the CF cluster is given in the upper-left corner of the table.

In Figure 14, for example, Interconnect 1 (Int 1) has eth0 on fuji2 and fuji3 attached to it. The cluster name is FUJI.

The connections and topology tables typically show devices that are on the public network. Using devices on a public network is a security risk; therefore, in general, do not use any devices on the public network as a CF interconnect. Instead, use devices on a private network.

Although the CF Wizard may list Int 1, Int 2, and so on, it should be pointed out that this is simply a convention in the GUI. CF itself does not number interconnects. Instead, it keeps track of point-to-point routes to other nodes.

Occasionally, there may be problems setting up the networking for the cluster. Cabling errors may mean that there are no full interconnects. If you click on the button next to *Topology*, the CF Wizard will display all the full interconnects, partial interconnects, and unconnected devices it has found. If a particular category is not found, it is omitted. For example, in Figure 14, only full interconnects are shown because no partial interconnects or unconnected devices were found on fuji2 or fuji3.

To configure CF using the connection table, click on the interconnects that have the devices that you want to use.

When you are satisfied with your choices, click on *Next* to go to the CF over IP configuration window (Figure 15).

Topology table

The topology table gives more flexibility in configuration than the connection table. In the connection table, you could only select an interconnect, and all devices on that interconnect would be configured. In the topology table, you can individually select devices.

While you can configure CF using the topology table, you may wish to take a simpler approach. If no full interconnects are found, then display the topology table to see what your networking configuration looks like to CF. Using this information, correct any cabling or networking problems that prevented the full interconnects from being found. Then go back to the CF Wizard window where the cluster name was entered and click on *Next* to cause the Wizard to reprove the interfaces. If you are successful, then the connection table will show the full interconnects, and you can select them. Otherwise, you can repeat the process.

The text area at the bottom of the window lists problems or warnings concerning the configuration.

When you are done, click on *Next* to go to the CF over IP configuration window (Figure 15).

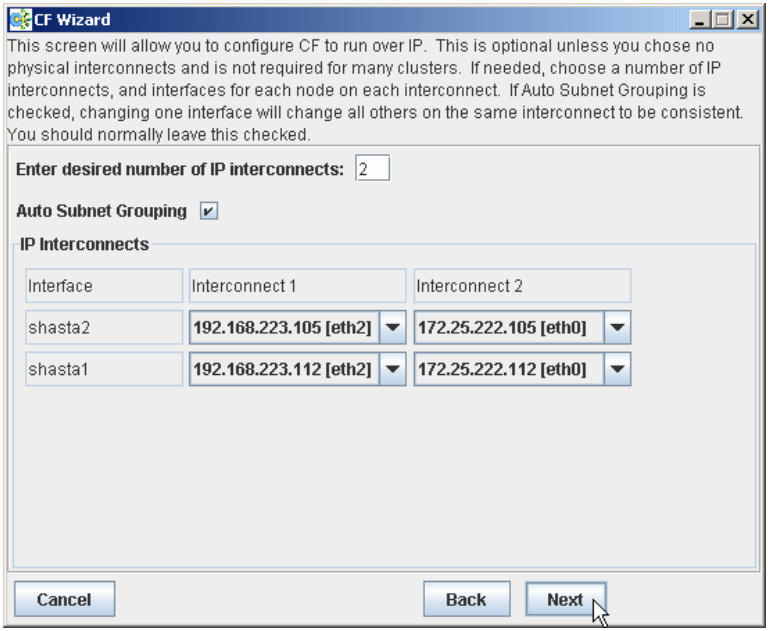

Figure 15: CF over IP window

The screen shown in Figure 15 lets you configure CF to run over IP. If you have already configured CF over Ethernet in the topology table or the connection table, you do not have to change any settings on this screen. Leave the number of IP interconnects set to its default of 0, and click *Next*.

CF can use either Ethernet packets or IP for its communication. The topology table and connection table discussed previously allow you to configure CF to use Ethernet packets. This is the preferred CF configuration since CF over Ethernet is significantly faster than CF over IP.

However, CF over Ethernet requires Ethernet link-level connectivity between the nodes in a cluster. In certain disaster recovery scenarios, there may only be IP connectivity between hosts. This is typically the case when the hosts are separated by large geographical distances.

CF over IP uses IP subnetworks in the same way that CF over Ethernet uses physical interconnects. Each IP interconnect must correspond to exactly one subnetwork. For example, suppose that your nodes had the following IP interfaces:

```
node      subnet 1         subnet 2          subnet 3
-----------------------------------------------------------
shasta1   172.25.222.105  192.168.223.105   185.33.48.105
shasta2   172.25.222.112  192.168.223.112   185.33.48.112
```

Using CF over IP, you might configure one IP interconnect to use the IP addresses 192.168.223.105 and 192.168.223.112. You could configure a second IP interface using the addresses 185.33.48.105 and 185.33.48.112.

But if you need CF over IP, then set the number of IP interconnects to 2 (or more if desired). The Wizard will propose IP interconnects. The IP interconnects are conveniently sorted by subnetwork. If you think that a particular subnetwork is missing a node, then double check that the netmask and broadcast addresses are properly configured for all nodes on subnetwork.

Select the subnetworks that you want to use as your IP interconnects. You should avoid using addresses on your public network. CF allows promiscuous joins without any limits, so it is best to use private subnetworks for your IP interconnects.

With this setting, CF can be configured to run over the IP interface. After entering the number of required IP interconnects and pressing [Enter], the CF Wizard will display interconnects sorted by available subnetworks, netmasks, and broadcast addresses.

All the IP addresses for all the nodes on a given IP interconnect must be on the same IP subnetwork and should have the same netmask and broadcast address.

Auto Subnet Grouping should always be checked in this window. If it is checked and you select one IP address for one node, then all of the other nodes in that column have their IP addresses changed to interfaces on the same subnetwork.

Choose the IP interconnects from the combo boxes on this window, and click on *Next*. The CIP Wizard windows like Figure 16 and Figure 17 appear.

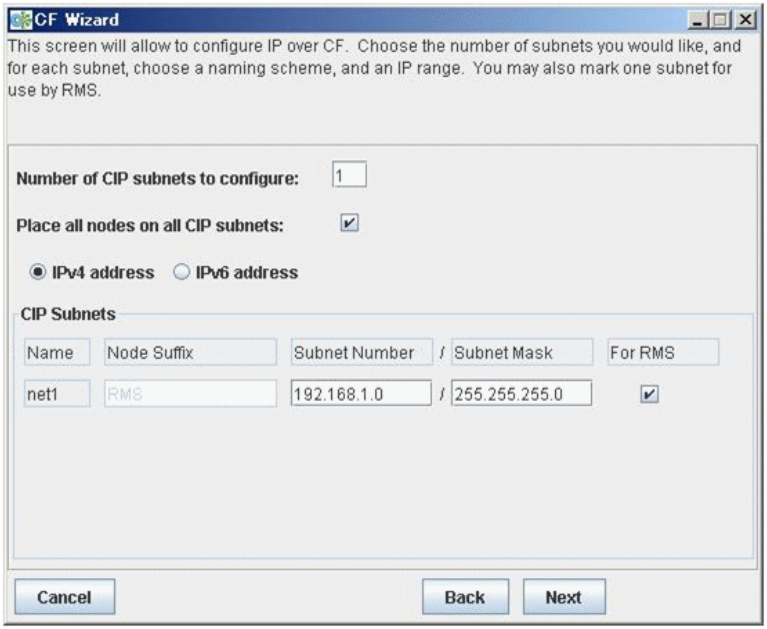

Figure 16: CIP wizard (IPv4) window

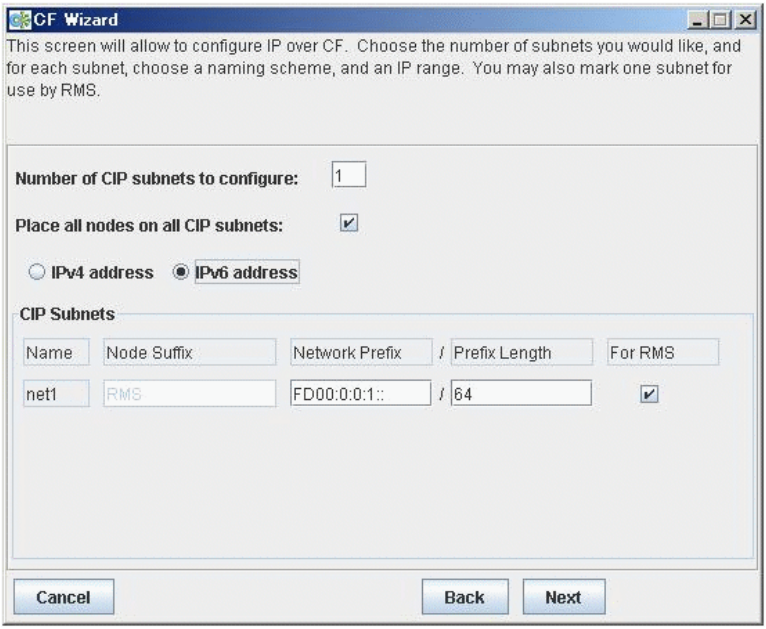

Figure 17: CIP wizard (IPv6) window

This window allows you to configure CIP. You can enter a number in the box after *Number of CIP subnets to configure* to set the number of CIP subnets to configure. The maximum number of CIP subnets is 8.

For each defined subnet, the CIP Wizard configures a CIP interface on each node defined in the CF cluster.

Set either IPv4 or IPv6 as the IP address to set to the CIP interface.

By selecting either of the [IPv4 address] or [IPv6 address] radio button, you can switch the window shown in Figure 16: CIP wizard (IPv4) window and Figure 17: CIP wizard (IPv6) window.

When using IPv4 for CIP interface

The following values are assigned for CIP interface:

- The IP address will be a unique IP number on the subnet specified in the *Subnet Number* field. The node portions of the address start at 1 and are incremented by 1 for each additional node.

  The CIP Wizard will automatically fill in a default value for the *Subnet Number* filed for each CIP subnetwork requested. The default values are taken from the private IP address range specified by RFC 1918. Note that the values

entered in the *Subnet Number* field have 0 for their node portion even though the CIP Wizard starts the numbering at 1 when it assigns the actual node IP addresses.

- The IP name of the interface will be of the form *cfnameSuffix* where *cfname* is the name of a node from the CF Wizard, and the *Suffix* is specified in the field *Node Suffix*. If the checkbox *For RMS* is selected, then the *Node Suffix* will be set to RMS and will not be editable. If you are using RMS, one CIP network must be configured for RMS.
- The *Subnet Mask* will be the value specified.

In Figure 16, the system administrator has selected 1 CIP network. The *For RMS* checkbox is selected, so the RMS suffix will be used. Default values for the *Subnet Number* and *Subnet Mask* are also selected. The nodes defined in the CF cluster are fuji2 and fuji3. This will result in the following configuration:

- On fuji2, a CIP interface will be configured with the following:
  IP nodename: fuji2RMS
  IP address: 192.168.1.1
  Subnet Mask: 255.255.255.0
- On fuji3, a CIP interface will be configured with the following:
  IP nodename: fuji3RMS
  IP address: 192.168.1.2
  Subnet Mask: 255.255.255.0

When using IPv6 for CIP interface

The following values are assigned for CIP interface:

- The IP address will be a unique IP number on the network prefix specified in the *Prefix* field. Interface ID of the address start at 1 and are incremented by 1 for each additional node.

  The CIP Wizard will automatically fill in a default value for the *Prefix* field for each CIP subnetwork requested. The default values are taken from the Unique Local Unicast Address range specified by RFC 4193. Note that the values entered in the *Prefix* field have 0 for their interface ID portion even though the CIP Wizard starts the numbering at 1 when it assigns the actual node IP addresses.
- The IP name of the interface will be of the form *cfnameSuffix* where *cfname* is the name of a node from the CF Wizard, and the *Suffix* is specified in the field *Node Suffix*. If the checkbox *For RMS* is selected, then the *Node Suffix* will be set to RMS and will not be editable. If you are using RMS, one CIP network must be configured for RMS.
- The *Prefix Length* will be the value specified.

In Figure 17, the system administrator has selected 1 CIP network. The *For RMS* checkbox is selected, so the RMS suffix will be used. Default values for the *Prefix* and *Prefix Length* are also selected. The nodes defined in the CF cluster are fuji2 and fuji3. This will result in the following configuration:

- On fuji2, a CIP interface will be configured with the following:
  IP nodename: fuji2RMS
  IPv6 address: FD00:0:0:1::1
  Prefix Length: 64
- On fuji3, a CIP interface will be configured with the following:
  IP nodename: fuji3RMS
  IPv6 address: FD00:0:0:1::2
  Prefix Length: 64

The CIP Wizard stores the configuration information in the file /etc/cip.cf on each node in the cluster. This is the default CIP configuration file. The Wizard will also update /etc/hosts on each node in the cluster to add the new IP node names.

When you click on the *Next* button, CIM configuration window appears (Figure 18).

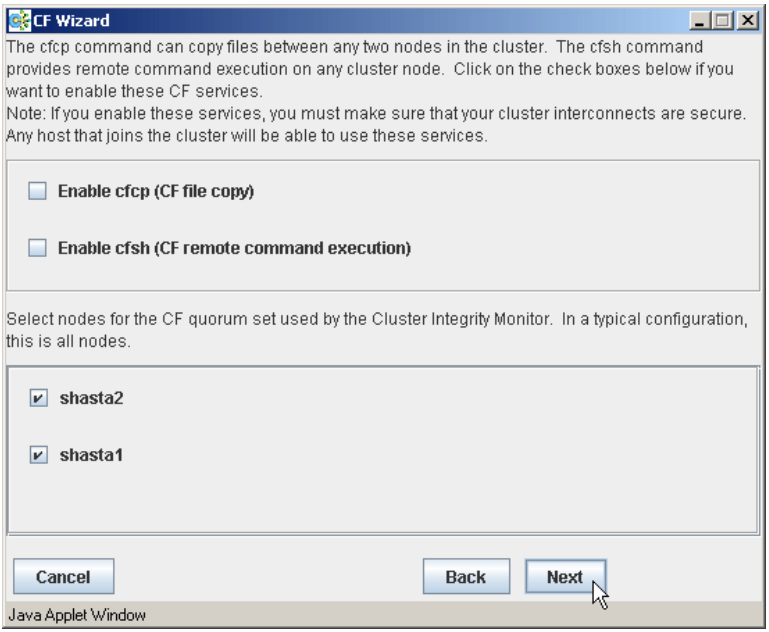

Figure 18: CIM configuration window

The CIM configuration window in Figure 18 has the following parts:

- The upper portion allows you to enable cfcp and cfsh.

  cfcp is a CF-based file copy program. It allows files to be copied among the cluster hosts. cfsh is a remote command execution program that, similar to cfcp, works between nodes in the cluster. The use of these programs is optional. In this example these items are not selected. If you enable these services, however, any node that has access to the cluster interconnects can copy files or execute commands on any node with root privileges.

- The lower portion allows you to determine which nodes should be monitored by CIM.

This window also lets you select which nodes should be part of the CF quorum set. The CF quorum set is used by the CIM to tell higher level services when it is safe to access shared resources.

Caution
Do not change the default selection of the nodes that are members of the CF quorum set unless you fully understand the ramifications of this change.

A checkbox next to a node means that node will be monitored by CIM. By default, all nodes are checked. For almost all configurations, you will want to have all nodes monitored by CIM.

This window will also allow you to configure CF Remote Services. You can enable either remote command execution, remote file copying, or both.

Caution
Enabling either of these means that you must trust all nodes on the CF interconnects and the CF interconnects must be secure. Otherwise any system able to connect to the CF interconnects will have access to these services.

Click on the *Next* button to go to the summary window (Figure 19).

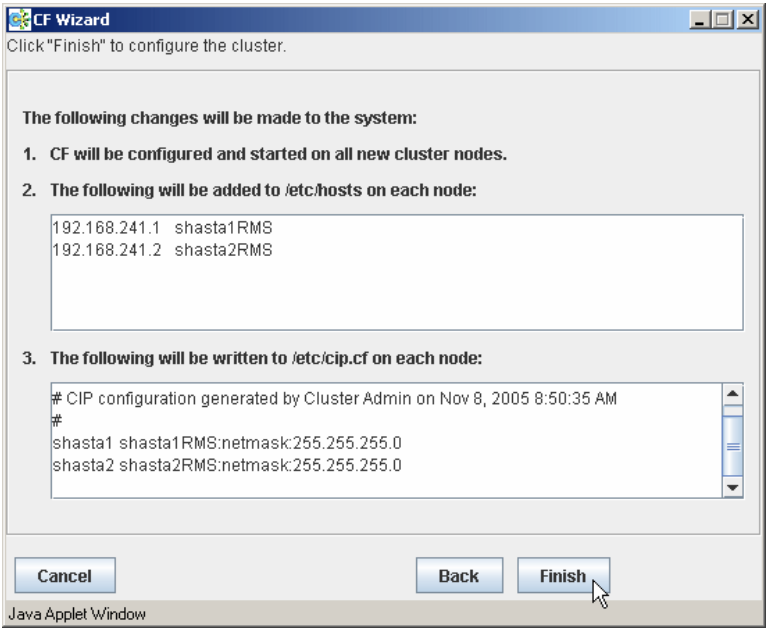

Figure 19: Summary window

This window summarizes the major changes that the CF, CIP, and CIM Wizards will perform. When you click on the *Finish* button, the CF Wizard performs the actual configuration on all nodes.

A window similar to Figure 20 is displayed while the configuration is being done.

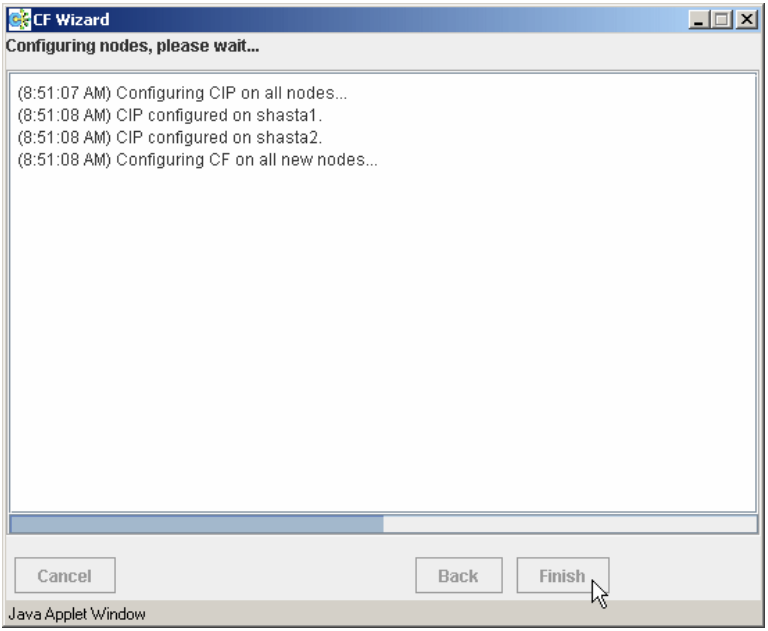

Figure 20: Configuration processing window

This window is updated after each configuration step. When the configuration successfully completes, a small completion pop-up window appears (Figure 21).

Figure 21: Configuration completion pop-up

Click on the *OK* button, and the pop-up is dismissed. The configuration processing window now has a *Finish* button (Figure 22).

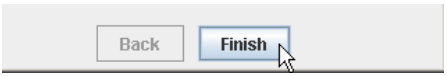

Figure 22: Configuration window after completion

Click on the *Finish* button to dismiss the window in Figure 22. A small pop-up appears asking if you would like to run the SF Wizard. Click on *yes*, and run the SF Wizard (described in the Section "Invoking the Configuration Wizard").

After the CF (and optionally the SF) Wizards are done, you see the main CF window. After several moments, the window will be updated with new configuration and status information (Figure 23).

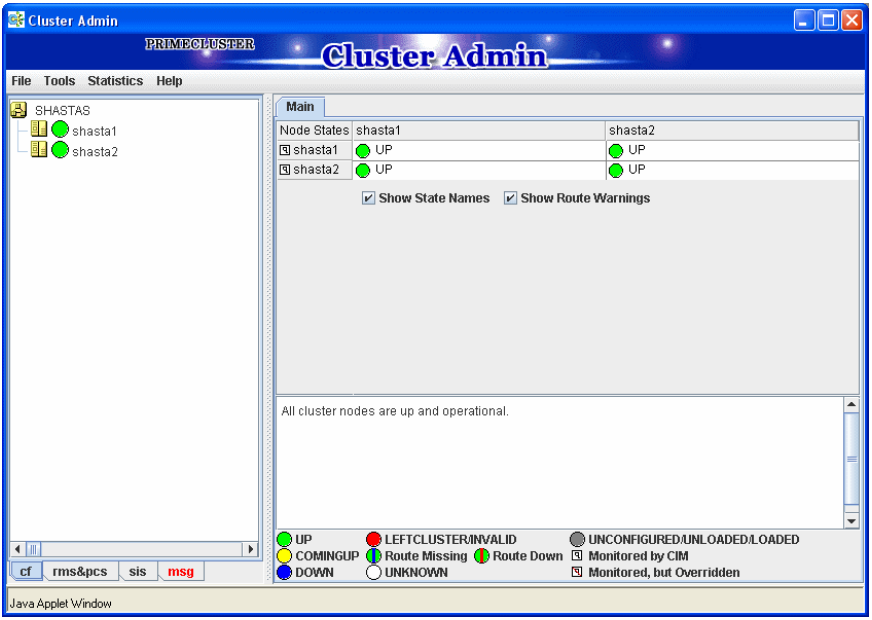

Figure 23: Main CF window

2.1.6 Adding a new node to CF

This section describes how to add a node to an existing CF cluster.

The first step is to make sure that Web-Based Admin View is properly configured on the new node. Refer to the *Web-Based Admin View Operation Guide* for additional details on Web-Based Admin View configuration options.

After you have properly configured Web-Based Admin on the new node, you should start Cluster Admin. If you are already running the Cluster Admin GUI, exit it and then restart it.

The first window that Cluster Admin displays is the small initial connection pop-up window (Figure 6). This window lists all of the nodes which are known to Web-Based Admin View. If the new node is not present in this list, then you should recheck your Web-Based Admin configuration and also verify that the new node is up.

To add the new node, select it in the initial connection pop-up. After making your selection, run the CF Wizard by clicking on the *Configure* button (Figure 8) The CF Wizard will appear, and you can use it to join the existing CF cluster.

The CF Wizard will allow you to configure CF, CIM, and CIP on the new node. After it is run, you should also run the SF Wizard to configure the Shutdown Facility on the new node.

You will also need to do additional configuration work for other PRIME-CLUSTER products you might be using such as the Cluster Resource Manager (CRM), RMS, GFS, and so forth.

2.1.7 Example of CF configuration by CLI

When configuring CF by CLI, perform the following steps.
In this section, the cluster system configured with two nodes where the CF node names are "fuji2" and "fuji3", is explained as an example.

1. Create CIP configuration files.
   Specify /etc/cip.cf as below on all the nodes which configure the cluster system.

   Example:

   ```
   fuji2    fuji2RMS:netmask:255.255.255.0
   fuji3    fuji3RMS:netmask:255.255.255.0
   ```

   

   Note

   If you manually create /etc/cip.cf, you cannot reconfigure CF by Cluster Admin. To reconfigure CF by Cluster Admin, delete the /etc/cip.cf file beforehand.

2. Set IP addresses.
   Specify /etc/hosts as below on all the nodes which configure the cluster system.
   Example:

<*cip address1*> fuji2RMS
<*cip address2*> fuji3RMS

3. Enable remote access by using cfcp/cfsh.
   Specify /etc/default/cluster.config as below on all the nodes which configure the cluster system.

   CFCP "cfcp"
   CFSH "cfsh"

4. Edit /etc/default/cluster on all the nodes.

   a) Edit /etc/default/cluster to create the files of the following contents:
   nodename <*CF node name*>
   clustername <*Cluster name*>
   device <*Cluster interconnect 1*>
   device <*Cluster interconnect 2*>

   Example:

   nodename fuji2
   clustername TESTCLUSTER
   device /dev/fjgi0
   device /dev/fjgi1

   

   Note

   Make sure that the node name to be defined in nodename is the CF node name, not the node name of the OS.

   b) Set the owner, group, and access permission.
   # **chown root:root /etc/default/cluste**r
   # **chmod 600 /etc/default/cluster**

   c) Reboot the nodes.

5. Execute the following command with any node in the cluster system and set the Cluster Integrity Monitor (CIM).

   # **rcqconfig -a <*nodename*> ...**

   nodename : CF node name

   Example:

   # **rcqconfig -a fuji2 fuji3**

6. Check that it can be communicated with the RMS node name.

   Example: When checking from fuji2

   # **ping fuji3RMS**

If it cannot be communicated, check that CF node names, RMS node names, and CIP addresses set in /etc/cip.cf, /etc/default/cluster, and /etc/hosts are correct again.

2.2 CIP configuration file

The CIP configuration file is stored in /etc/cip.cf on each node in the cluster. Normally, you can use the GUI to create this file during cluster configuration time. However, there may be times when you wish to manually edit this file.

The format of a CIP configuration file entry is as follows:

cfname CIP_Interface_Info [*CIP_Interface_Info ...*] [IPv6]

- *cfname* tells what node the configuration information is for.
- *CIP_Interface_Info* gives information needed to configure a single CIP interface.

  Normally, the configuration information of all the CIP interfaces on all the nodes are contained in the cip.cf configuration file.
  - For IPv4, specify *CIP_Interface_Info* with the following format:

    IPv4-Address[:*Option*[:*Option...*]]

    Specify it without any spaces even around colons.

    For *IPv4-Address*, specify as a number in Internet standard dotted-decimal notation or as the Host name. When specifying with the Host name, it needs to be defined in /etc/hosts.

    The IP address can also have additional options following it. These options are passed to the configuration command ifconfig. Each option is separated from the IP address and other option by colons (:).
  - For IPv6, specify *CIP_Interface_Info* with the following format:

    Hostname:"["*IPv6-Address*/*prefix_length*"]"

    Specify it without any spaces around colons, slashes, and inside of each blankets "[", "]".

    For *Hostname*, describe the Host name to specify the cip address.

    For *IPv6-Address* and *prefix_length*, specify the IPv6 address and the prefix length denoted as a hexadecimal code which is separated by Internet standard colons.
- When using the IPv6 address, specify "IPv6" in the end of the line.

For example, the CIP configuration done in Section "Example of creating a cluster" would produce the following CIP configuration file:

```
fuji2   fuji2RMS:netmask:255.255.255.0
fuji3   fuji3RMS:netmask:255.255.255.0
```

Although not shown in this example, the CIP syntax does allow multiple CIP interfaces for a node to be defined on a single line. The cip.cf manual page has more details about the cip.cf file.

If you make changes to the cip.cf file by hand, you should be sure that the file exists on all nodes, and all nodes are specified in the file. Be sure to update all nodes in the cluster with the new file. Changes to the CIP configuration file will not take effect until CIP is stopped and restarted.

After stopping all applications that use CIP, restart CIP by stopping and starting CF.

For instructions on starting and stopping CF, see Section "Starting and stopping CF".

2.3 Cluster Configuration Backup and Restore (CCBR)

Caution
CCBR only saves PRIMECLUSTER configuration information. It does not replace an external, full backup facility.

CCBR provides a simple method to save the current PRIMECLUSTER configuration information of a cluster node. It also provides a method to restore the configuration information whenever a node update has caused severe trouble or failure, and the update (and any side-effects) must be removed. CCBR provides a node-focused backup and restore capability. Multiple cluster nodes must each be handled separately.

CCBR provides the following commands:

- cfbackup(1M)—Saves all information into a directory that is converted to a compressed tar archive file.
- cfrestore(1M)—Extracts and installs the saved configuration information from one of the cfbackup(1M) compressed tar archives.

After cfrestore(1M) is executed, you must reactivate the RMS configuration in order to start RMS. Once the reactivation of the RMS configuration is done, RMS will have performed the following tasks:

- Checked the consistency of the RMS configuration
- Established the detector links for RMS to be able to monitor resources
- Ensured proper communication between cluster nodes
- Created the necessary aliases for the shell commands used in the Wizard Tools. This is done automatically during RMS activation.

Please refer to the PRIMECLUSTER *Reliant Monitor Services (RMS) Configuration and Administration Guide* for details on how to activate RMS Configuration.

To guarantee that the cfrestore(1M) command will restore a functional PRIMECLUSTER configuration, it is recommended that there be no hardware or operating system changes since the backup was taken, and that the same versions of the PRIMECLUSTER products are installed.

Because the installation or reinstallation of some PRIMECLUSTER products add kernel drivers, device reconfiguration may occur. This is usually not a problem. However, if Network Interface Cards (NICs) have been installed, removed, replaced, or moved, the device name (for example, the number 2 in /dev/eth2) can change. Any changes of this nature can, in turn, cause a restored PRIMECLUSTER configuration to be invalid.

cfbackup(1M) and cfrestore(1M) consist of a framework and plug-ins. The framework and plug-ins function as follows:

1. The framework calls the plug-in for the SMAWcf package.
2. This plug-in creates and updates the saved-files list, the log files, and error log files.
3. All the other plug-ins for installed PRIMECLUSTER products are called in name sequence.
4. Once all plug-ins have been successfully processed, the backup directory is archived by means of tar(1M) and compressed.
5. The backup is logged as complete and the file lock on the log file is released.

The cfbackup(1M) command runs on a PRIMECLUSTER node to save all the cluster configuration information. To avoid any problem, this command should be concurrently executed on every cluster node to save all relevant PRIME-CLUSTER configuration information. This command must be executed as root.

If a backup operation is aborted, no tar archive is created. If the backup operation is not successful for one plug-in, the command processing will abort rather than continue with the next plug-in. cfbackup(1M) exits with a status of zero on success and non-zero on failure.

The cfrestore(1M) command runs on a PRIMECLUSTER node to restore all previously saved PRIMECLUSTER configuration information from a compressed tar archive. The node must be in single-user mode with CF not loaded. The node must not be an active member of a cluster. The command must be executed as root. cfrestore(1M) exits with a status of zero on success and non-zero on failure.

It is recommended to reboot once cfrestore(1M) returns successfully. If cfrestore(1M) aborts, the reason for this failure should be examined carefully since the configuration update may be incomplete.

You cannot run cfbackup(1M) and cfrestore(1M) at the same time on the same node.

Some PRIMECLUSTER information is given to a node when it joins the cluster. The information restored is not used. To restore and to use this PRIMECLUSTER information, the entire cluster needs to be DOWN, and the first node to create the cluster must be the node with the restored data. When a node joins an existing, running cluster, the restored configuration is gone because it is the first node in the cluster that determines which restored configuration to use.

The following files and directories that are fundamental to the operation of the cfbackup(1M) and cfrestore(1M) commands:

- The /opt/SMAW/ccbr/plugins directory contains executable CCBR plug-ins. The installed PRIMECLUSTER products supply them.
- The /opt/SMAW/ccbr/ccbr.conf file must exist and specifies the value for CCBRHOME, the pathname of the directory to be used for saving CCBR archive files. A default ccbr.conf file, with CCBRHOME set to /var/spool/SMAW/SMAWccbr is supplied as part of the SMAWccbr package.

  The system administrator can change the CCBRHOME pathname at anytime. It is recommended that the system administrator verify that there is enough disk space available for the archive file before setting CCBRHOME. The system administrator might need to change the CCBRHOME pathname to a file system with sufficient disk space.

  

  It is important to remember that re-installing the SMAWccbr package will reset the contents of the /opt/SMAW/ccbr/ccbr.conf file to the default package settings.

  The following is an example of ccbr.conf:

```
#!/bin/ksh -
#ident "@(#)ccbr.conf Revision: 12.1 02/05/08 14:45:57"
#
# CCBR CONFIGURATION FILE
#
# set CCBR home directory
#
CCBRHOME=/var/spool/SMAW/SMAWccbr
export CCBRHOME
```

- The /opt/SMAW/ccbr/ccbr.gen (generation number) file is used to form the name of the CCBR archive to be saved into (or restored from) the CCBRHOME directory. This file contains the next backup sequence number (or generation number), which is appended to the archive name as a unique identifier.

  If this file is ever deleted, cfbackup(1M) and/or cfrestore(1M) will create a new file containing the value string of 1. Both commands will use either the generation number specified as a command argument, or the file value if no command argument is supplied. The cfbackup(1M) command additionally checks that the command argument is not less than the value of the /opt/SMAW/ccbr/ccbr.gen file. If the command argument is less than the value of the /opt/SMAW/ccbr/ccbr.gen file, the cfbackup(1M) command will use the file value instead.

  Upon successful execution, the cfbackup(1M) command updates the value in this file to the next sequential generation number. The system administrator can update this file at any time.

- If cfbackup(1M) backs up successfully, a compressed tar archive file with the following name will be generated in the CCBRHOME directory as follows:

  *hostname*_ccbr*N*.tar.Z

  hostname is the nodename and *N* is the number suffix for the generation number.

  For example, in the cluster node fuji2, with the generation number 5, the archive file name is as follows:

  fuji2_ccbr5.tar.Z

- Each backup request creates a backup tree directory. The directory is as follows:

  CCBRHOME/*nodename*_ccbr*N*

  nodename is the node name and *N* is the number suffix for the generation number.

  CCBROOT is set to this directory.

  For example, enter the following on the node fuji2:

  fuji2# **cfbackup 5**

  Using the default setting for CCBRHOME, the following directory will be created:

  /var/spool/SMAW/SMAWccbr/fuji2_ccbr5

  This backup directory tree name is passed as an environment variable to each plug-in.

- The CCBRHOME/ccbr.log log file contains startup, completion messages, and error messages. All the messages are time stamped.
- The CCBROOT/errlog log file contains specific error information when a plug-in fails. All the messages are time stamped.
- The CCBROOT/plugin.blog or CCBROOT/plugin.rlog log files contain startup and completion messages from each backup/restore attempt for each plug-in. These messages are time stamped.

Refer to the Chapter "Manual pages" for more information on cfbackup(1M) and cfrestore(1M).

cfbackup example

The following command backs up and validates the configuration files for all CCBR plug-ins that exist on the system fuji2.

```
fuji2# cfbackup
```

CCBR performs the backup automatically and does not require user interaction. Processing has proceeded normally when a message similar to the following appears at the end of the output:

```
04/30/04 09:16:20 cfbackup 11 ended
```

This completes the backup of PRIMECLUSTER.

In the case of an error, the subdirectory /var/spool/SMAW/SMAWccbr/fuji2_ccbr11 is created.

Refer to the Chapter "Diagnostics and troubleshooting" for more details on troubleshooting CCBR.

cfrestore example

Before doing cfrestore(1M), CF needs to be unloaded, the system needs to be in single-user mode, and the file systems /opt, /var, and /usr need to be online.

The following files are handled differently during cfrestore(1M):

- root files—These are the files under the CCBROOT/root directory. They are copied from the CCBROOT/root file tree to their corresponding places in the system file tree.
- OS files—These files are the operating system files that are saved in the archive but not restored. The system administrator might need to merge the new OS files and the restored OS files to get the necessary changes.

For example, on fuji2 we entered the following command to restore the configuration to backup 11.

```
fuji2# cfrestore 11
```

The restore process asks you to confirm the restoration and then carries out the process automatically. Processing has proceeded normally when a message similar to the following appears at the end of the output:

```
05/05/04 13:49:19 cfrestore 11 ended
```

This completes the PRIMECLUSTER restore.

3 CF Registry and Integrity Monitor

This chapter discusses the purpose and physical characteristics of the CF registry (CFREG), and it discusses the purpose and implementation of the Cluster Integrity Monitor (CIM).

This chapter discusses the following:

- The Section "CF Registry" discusses the purpose and physical characteristics of the CF synchronized registry.
- The Section "Cluster Integrity Monitor" discusses the purpose and implementation of CIM.

3.1 CF Registry

The CFREG provides a set of CF base product services that allows cluster applications to maintain cluster global data that must be consistent on all of the nodes in the cluster and must live through a clusterwide reboot.

Typical applications include cluster-aware configuration utilities that require the same configuration data to be present and consistent on all of the nodes in a cluster (for example, cluster volume management configuration data).

The data is maintained as named registry entries residing in a data file where each node in the cluster has a copy of the data file. The services will maintain the consistency of the data file throughout the cluster.

A user-level daemon (cfregd), runs on each node in the cluster, and is responsible for keeping the data file on the node where it is running synchronized with the rest of the cluster. The cfregd process will be the only process that ever modifies the data file. Only one synchronization daemon process will be allowed to run at a time on a node. If a daemon is started with an existing daemon running on the node, the started daemon will log messages that state that a daemon is already running and terminate itself. In such a case, all execution arguments for the second daemon will be ignored.

3.2 Cluster Integrity Monitor

The purpose of the CIM is to allow applications to determine when it is safe to perform operations on shared resources. It is safe to perform operations on shared resources when a node is a member of a cluster that is in a consistent state.

A consistent state is means that all the nodes of a cluster that are members of the CIM set are in a known and safe state. The nodes that are members of the CIM set are specified in the CIM configuration. Only these nodes are considered when the CIM determines the state of the cluster. When a node first joins or forms a cluster, the CIM indicates that the cluster is consistent only if it can determine the status of the other nodes that make up the CIM set and that those nodes are in a safe state.

As methods for guaranteeing a quorum, PRIMECLUSTER supports the following three CIM methods:

- Node State Management (NSM) method
- RCI method
- MMB method

The NSM method uses the state of the other node as reported by CF and periodically determines whether that node is in a known state. If the state is known, The CF states that are determined as known are UP and DOWN. If the node is being activated or is in LEFTCLUSTER state, that state is determined to be unknown (a quorum is not guaranteed).

The RCI method uses the PRIMEPOWER RCI interface and asynchronously determines the state of a cluster node.

The MMB method uses the PRIMEQUEST MMB interface and asynchronously determines the state of a cluster node.

In this way, the CIM provides applications with a set of functions that determine nodes are in a consistent state (whether or not there is a quorum).

PRIMECLUSTER uses CIM to determine whether a user application process that uses resources shared by multiple nodes in a cluster can be processed safely without triggering process contention. In other words, it is safe to perform operations on shared resources if the node executing the process is a member of a cluster system that is in a consistent state (quorum). In the PRIME-CLUSTER system, the consistent state is set when all nodes in the cluster system monitored by the CIM are either in operating (UP) or stopped (DOWN) state and are also in a safe state. The nodes monitored by the CIM to all nodes that were set when the CIM was configured. Only these nodes are considered when the CIM checks the cluster status. When a node first joins or forms a cluster, the CIM indicates that the cluster is consistent only it can determine the status of the other nodes that make up the CIM set and that those nodes are in a safe state.

CIM currently supports the Node State Management (NSM) method. The CIM reports on a cluster state that a node state is known and safe (True), or a node state is unknown (False) for the node. True and False are defined as follows:

True—All CIM nodes in the cluster are in a known and safe state.

False—One or more CIM nodes in the cluster are in an unknown or unsafe state.

3.2.1 Configuring CIM

You can perform CIM procedures through the following methods:

- Cluster Admin GUI—This is the preferred method of operation. Refer to the Section "Adding and removing a node from CIM" for the GUI procedures.
- CLI—Refer to the Chapter "Manual pages" for complete details on the CLI options and arguments, some of which are described in this section. The commands can be found in the following directory:

  /opt/SMAW/SMAWcf/bin

rcqconfig

The CIM is configured using the command rcqconfig(1M) after CF starts. The rcqconfig(1M) command is used to set up or to change the CIM configuration. You only need to run this command if you are not using Cluster Admin to configure CIM.

When rcqconfig(1M) is invoked, it checks that the node is part of the cluster. When the rcqconfig(1M) command is invoked without any option, it checks if any configuration is present in the CFReg.database after the node joins the cluster. If there is none, it returns as error. When you are using the GUI, these actions are done as part of the configuration process.

rcqconfig(1M) configures a quorum set of nodes, among which CF decides the quorum state. rcqconfig(1M) is also used to show the current configuration. If rcqconfig(1M) is invoked without any configuration changes or with only the -v option, rcqconfig(1M) will apply any existing configuration to all the nodes in the cluster. It will then start or restart the quorum operation. rcqconfig(1M) can be invoked from the command line to configure or to start the quorum.

3.2.2 Query of the quorum state

CIM recalculates the quorum state when it is triggered by some node state change. However you can force the CIM to recalculate it by running rcqquery(1M) at any time. Refer to the Chapter "Manual pages" for complete details on the CLI options and arguments.

rcqquery(1M) functions as follows:

- Queries the state of quorum and gives the result using the return code. It also gives you readable results if the verbose option is given.
- Returns True if the states of all the nodes in the quorum set are known. If the state of any node is unknown, then it returns False.
- Exits with a status of zero when a quorum exists, and it exits with a status of 1 when a quorum does not exist. If an error occurs during the operation, then it exits with any other non-zero value other than 1.

3.2.3 Reconfiguring quorum

Refer to the Section "Adding and removing a node from CIM" for the GUI procedures.

CLI

The configuration can be changed at any time and is effective immediately. When a new node is added to the quorum set of nodes, the node being added must be part of the cluster so as to guarantee that the new node also has the same quorum configuration. Removing a node from the quorum set can be done without restriction.

When the configuration information is given to the command rcqconfig(1M) as arguments, it performs the transaction to CFREG to update the configuration information. Until CIM is successfully configured and gets the initial state of the quorum, CIM has to respond with the quorum state of False to all queries.

Examples

In this example the cluster has formed but no quorum was established. Display the states of all the nodes in the cluster as follows:

fuji2# **cftool -n**

Node	Number	State	Os	Cpu
fuji2	1	UP	Linux	Pentium
fuji3	2	UP	Linux	Pentium

Display the current quorum configuration as follows:

fuji2# **rcqconfig -g**

Nothing is displayed, since no nodes have been added so far.

Add new nodes in a quorum set of nodes as follows:

fuji2# **rcqconfig -a fuji2 fuji3**

Display the current quorum configuration parameters as follows:

fuji2# **rcqconfig -g**

QUORUM_NODE_LIST= fuji2 fuji3

Delete nodes from a quorum set of nodes as follows:

fuji2# **rcqconfig -d fuji2**

Display the current quorum configuration parameters after one node is deleted as follows:

fuji2# **rcqconfig -g**

QUORUM_NODE_LIST= fuji3

Add a new node, fuji11 (which is not in the cluster), in a quorum set of nodes as follows:

fuji2# **rcqconfig -a fuji2 fuji3 fuji11**

Cannot add node fuji11 that is not up.

Since CF only configured the cluster to consist of fuji2 and fuji3, fuji11 does not exist. The quorum set remains unchanged.

fuji2# **rcqconfig -g**

QUORUM_NODE_LIST= fuji3

4 Cluster resource management

This chapter discusses the Resource Database, which is a synchronized clusterwide database, holding information specific to several PRIMECLUSTER products.

This chapter discusses the following:

- The Section "Overview" introduces cluster resource management.
- The Section "Kernel parameters for Resource Database" discusses the default values of the Linux kernel which have to be modified when the Resource Database is used.
- The Section "Resource Database configuration" details how to set up the Resource Database for the first time on a new cluster.
- The Section "Start up synchronization" discusses how to implement a start up synchronization procedure for the Resource Database.
- The Section "Adding a new node" describes how to add a new node to the Resource Database.

4.1 Overview

The cluster Resource Database is a dedicated database used by PRIME-CLUSTER products. It is not a general purpose database which a customer could use for their own applications.

4.2 Kernel parameters for Resource Database

The default values of Linux kernel parameters have to be modified when the Resource Database is used. This section lists the kernel parameters that have to be changed. You could configure the kernel parameters at runtime by using the sysctl(8) utility. The following shows how to change and apply the new kernel parameters.

Change the values for SHMMAX as follows:

- Edit the file /etc/sysctl.conf
- Add the entry: kernel.shmmax = value

Change the values for the semaphore parameters as follows:

kernel.sem = *value1 value2 value3 value4*

Where the preceding values are defined as the following:

- *value1* = value for SEMMSL
- *value2* = value for SEMMNS
- *value3* = value for SEMOPM
- *value4* = value for SEMMNI

The Resource Manager does not use SEMMSL and SEMOPM; therefore, leave these values unchanged. The following commands show the values for these parameters prior to any modification:

- #sysctl -a | grep sem
- #sysctl -a | grep shm

The following command changes the values for the parameters:

- #sysctl -p *filename* (default for *filename* is /etc/sysctl.conf)

This command applies the values from /etc/sysctl.conf to the kernel. It is not necessary to rebuild the kernel.

Use sysctl(8) to modify kernel parameters at runtime. Refer to sysctl(8) manual pages for the details on how sysctl performs.

1. semmni

   Value required for Resource Database is 20 or more.

2. semmns

   Value required for Resource Database is 30 or more.

3. shmmax

   shmmax = 1048576 + *the value necessary for Resource Database or more* as shown by the following:

   Number of resources=(*a*)+(*b*)

   (*a*) is the (*number of shared disks*) x (*number of sharing nodes*+1) x 2, where the *number of shared disks* is defined as follows:

   For disk array unit, use the number of logical units (LUN).

   For devices other than disk array unit, use the number of physical disks. *Number of sharing nodes* is defined as follows:

   The number of nodes connected to the shared disks.

   (*b*) is the total number of local disks (number of local disks in all cluster configuration nodes).

4. shmmni

   Value required for Resource Database is 30 or more.

As kernel parameters are used for these tuning parameters, they are validated after the configured node has been restarted. Therefore, if the number of nodes and logical disks increase as a result of expanding the facilities, you must perform reinstallation and then restart each node constituting the cluster.

If an additional node is added to the cluster, or if more disks are added after your cluster has been up and running, recalculate using the number of nodes and logical disks after expansion.

4.3 Resource Database configuration

This section discusses how to set up the Resource Database for the first time on a new cluster. The following procedure assumes that the Resource Database has not previously been configured on any of the nodes in the cluster.

If you need to add a new node to the cluster, and the existing nodes are already running the Resource Database, then a slightly different procedure needs to be followed. Refer to the Section "Adding a new node" for details.

Before you begin configuring the Resource Database, you must first make sure that CIP is properly configured on all nodes. The Resource Database uses CIP for communicating between nodes, so it is essential that CIP is working.

The Resource Database also uses the CIP configuration file /etc/cip.cf to establish the mapping between the CF node name and the CIP name for a node. If a particular node has multiple CIP interfaces, then only the first one is used. This will correspond to the first CIP entry for a node in /etc/cip.cf. It will also correspond to cip0 on the node itself.

Because the Resource Database uses /etc/cip.cf to map between CF and CIP names, it is critical that this file be the same on all nodes. If you used the Cluster Admin CF Wizard to configure CIP, then this will already be the case. If you created some /etc/cip.cf files by hand, then you need to make sure that all nodes are specified and they are the same across the cluster.

In general, the CIP configuration is fairly simple. You can use the Cluster Admin CF Wizard to configure a CIP subnet after you have configured CF. If you use the Wizard, then you will not need to do any additional CIP configuration. See the Section "CF, CIP, and CIM configuration" for more details.

After CIP has been configured, you can configure the Resource Database on a new cluster by using the following procedure. This procedure must be done on all the nodes in the cluster.

1. Log in to the node with system administrator authority.
2. Verify that the node can communicate with other nodes in the cluster over CIP.
   To test CIP network connectivity, execute the ping(1M) command or the ping6(8) command (when using the IPv6 address). The file /etc/cip.cf contains the CIP names that you should use in the ping(1M) command or the ping6(8) command.

   If you are using RMS and you have only defined a single CIP subnetwork, then the CIP names will be of the following form:

*cfname*RMS

For example, if you have two nodes in your cluster named fuji2 and fuji3, then the CIP names for RMS would be fuji2RMS and fuji3RMS, respectively. You could then run the following commands:

fuji2# **ping fuji3RMS**

fuji3# **ping fuji2RMS**

This tests the CIP connectivity.

3. Execute the clsetup command. When used for the first time to set up the Resource Database on a node, it is called without any arguments as follows:

   # /etc/opt/FJSVcluster/bin/clsetup

4. Execute the clgettree command to verify that the Resource Database was successfully configured on the node, as shown in the following:

   # /etc/opt/FJSVcluster/bin/clgettree

   The command should complete without producing any error messages, and you should see the Resource Database configuration displayed in a tree format.

   For example, on a two-node cluster consisting of fuji2 and fuji3, the clgettree command might produce output similar to the following:

```
Cluster 1 cluster
   Domain 2 Domain0
      Shared 7 SHD_Domain0
         Node 3 fuji2 UNKNOWN
         Node 5 fuji3 UNKNOWN
```

If you need to change the CIP configuration to fix the problem, you will also need to run the clinitreset command and start the information process over.

The format of clgettree is more fully described in its manual page. For the purpose of setting up the cluster, you need to check the following:

- Each node in the cluster should be referenced in a line that begins with the word Node.
- The clgettree output must be identical on all nodes.

If either of the above conditions are not met, then it is possible that you may have an error in the CIP configuration. Double-check the CIP configuration using the methods described earlier in this section. The actual steps are as follows:

1. Make sure that CIP is properly configured and running.
2. Run clinitreset on all nodes in the cluster.

   # /etc/opt/FJSVcluster/bin/clinitreset
3. Reboot each node.
4. Rerun the clsetup command on each node.

   # /etc/opt/FJSVcluster/bin/clsetup
5. Use the clgettree command to verify the configuration.

   # /etc/opt/FJSVcluster/bin/clgettree

4.4 Start up synchronization

A copy of the Resource Database is stored locally on each node in the cluster. When the cluster is up and running, all of the local copies are kept in sync. However, if a node is taken down for maintenance, then its copy of the Resource Database may be out of date by the time it rejoins the cluster. Normally, this is not a problem. When a node joins a running cluster, then its copy of the Resource Database is automatically downloaded from the running cluster. Any stale data that it may have had is thus overwritten.

There is one potential problem. Suppose that the entire cluster is taken down before the node with the stale data had a chance to rejoin the cluster. Then suppose that all nodes are brought back up again. If the node with the stale data comes up long before any of the other nodes, then its copy of the Resource Database will become the master copy used by all nodes when they eventually join the cluster.

To avoid this situation, the Resource Database implements a start up synchronization procedure. If the Resource Database is not fully up and running anywhere in the cluster, then starting the Resource Database on a node will cause that node to enter into a synchronization phase. The node will wait up to StartingWaitTime seconds for other nodes to try to bring up their own copies of the Resource Database. During this period, the nodes will negotiate among themselves to see which one has the latest copy of the Resource Database. The synchronization phase ends when either all nodes have been accounted

for or StartingWaitTime seconds have passed. After the synchronization period ends, the latest copy of the Resource Database that was found during the negotiations will be used as the master copy for the entire cluster.

The default value for StartingWaitTime is 60 seconds.

This synchronization method is intended to cover the case where all the nodes in a cluster are down, and then they are all rebooted together. For example, some businesses require high availability during normal business hours, but power their nodes down at night to reduce their electric bill. The nodes are then powered up shortly before the start of the working day. Since the boot time for each node may vary slightly, the synchronization period of up to StartingWaitTime ensures that the latest copy of the Resource Database among all of the booting nodes is used.

Another important scenario in which all nodes may be booted simultaneously involves the temporary loss and then restoration of power to the lab where the nodes are located.

However, for this scheme to work properly, you must verify that all nodes in the cluster have boot times that differ by less than StartingWaitTime seconds. Furthermore, you might need to modify the value of StartingWaitTime to a value that is appropriate for your cluster.

Modify the value of StartingWaitTime as follows:

1. Start up all of the nodes in your cluster simultaneously. It is recommended that you start the nodes from a cold power on. Existing nodes are not required to reboot when a new node is added to the cluster.

2. After the each node has come up, look in /var/log/messages for message number 2200. This message is output by the Resource Database when it first starts. For example, enter the following command:

   # grep 2200 /var/log/messages
   Feb 23 19:00:41 fuji2 dcmmond[407]: [ID 888197 daemon.notice] FJSVcluster: INFO: DCM: 2200: Cluster configuration management facility initialization started.

   Compare the timestamps for the messages on each node and calculate the difference between the fastest and the slowest nodes. This will tell you how long the fastest node has to wait for the slowest node.

3. Check the current value of StartingWaitTime by executing the clsetparam command on any of the nodes. For example, enter the following command:

 # /etc/opt/FJSVcluster/bin/clsetparam -p StartingWaitTime
60

The output for our example shows that StartingWaitTime is set to 60 seconds.

4. If there is a difference in start up times found in Step 2, the StartingWaitTime, or if the two values are relatively close together, then you should increase the StartingWaitTime parameter. You can do this by running the clsetparam command on any one node in the cluster. For example, enter the following command:

   # /etc/opt/FJSVcluster/bin/clsetparam -p StartingWaitTime 300

   This sets the StartingWaitTime to 300 seconds.

When you change the StartingWaitTime parameter, it is not necessary to stop the existing nodes. The new parameter will be effective for all nodes at the next reboot. Refer to the Chapter "Manual pages" for more details on the possible values for StartingWaitTime.

4.4.1 Start up synchronization and the new node

After the Resource Database has successfully been brought up on the new node, then you need to check if the StartingWaitTime used for start up synchronization is still adequate. If the new node boots much faster or slower than the other nodes, then you may need to adjust the StartingWaitTime time.

4.5 Adding a new node

If you have a cluster where the Resource Database is already configured, and you would like to add a new node to the configuration, then you should follow the procedures in this section. You will need to make a configuration change to the currently running Resource Database and then configure the new node itself. The major steps involved are listed below:

1. Back up the currently running Resource Database. A copy of the backup is used in a later step to initialize the configuration on the new node. It also allows you to restore your configuration to its previous state if a serious error is encountered in the process.
2. Reconfigure CF and CIP to include the new nodes and initialize.
3. Reconfigure the currently running Resource Database so it will recognize the new node.
4. Initialize the Resource Database on the new node.

5. Verify that the StartingWaitTime is sufficient for the new node, and modify this parameter if necessary.

Figure 24 shows these steps as a flow chart.

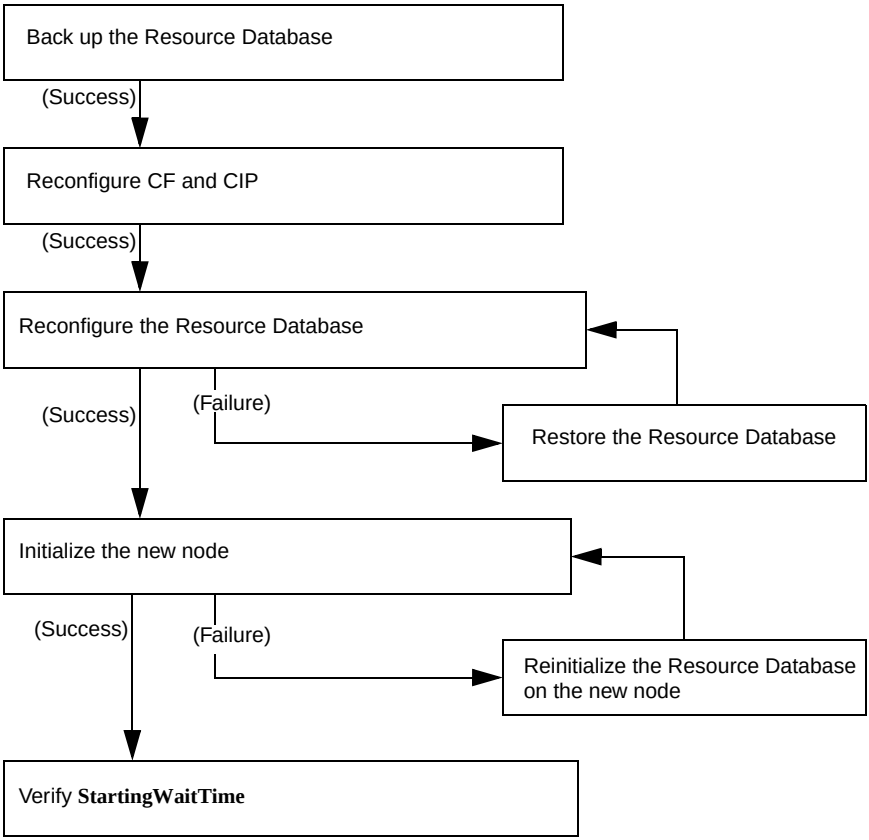

Figure 24: Adding a new node

The sections that follow describe each step in more detail.

4.5.1 Backing up the Resource Database

Before you add a new node to the Resource Database, you should first back up the current configuration. The backup will be used later to help initialize the new node. It is also a safeguard. If the configuration process is unexpectedly interrupted by a panic or some other serious error, then you may need to restore the Resource Database from the backup.

The configuration process itself should not cause any panics. However, if some non-PRIMECLUSTER software panics or if the SF causes a power cycle because of a CF cluster partition, then the Resource Database configuration process could be so severely impacted that a restoration from the backup would be needed.

The restoration process requires all nodes in the cluster to be in single user mode.

Since the Resource Database is synchronized across all of its nodes, the backup can be done on any node in the cluster where the Resource Database is running. The steps for performing the backup are as follows:

1. Log onto any node where the Resource Database is running as the system administrator authority.
2. Run the command clbackuprdb to back the Resource Database up to a file. The syntax is as follows:

   # /etc/opt/FJSVcluster/bin/clbackuprdb -f *file*

   For example:

   # /etc/opt/FJSVcluster/bin/clbackuprdb -f */mydir/***backup_rdb**

   clbackuprdb stores the Resource Database as a compressed tar file. Thus, in the above example, the Resource Database would be stored in */mydir/*backup_rdb.tar.Z.

   Make sure that you do not place the backup in a directory whose contents are automatically deleted upon reboot (for example, /tmp).

The hardware configuration must not change between the time a backup is done and the time that the restore is done. If the hardware configuration changes, you will need to take another backup. Otherwise, the restored database would not match the actual hardware configuration, and new hardware resources would be ignored by the Resource Database.

4.5.2 Reconfiguring the Resource Database

After you have backed up the currently running Resource Database, you will need to reconfigure the database to recognize the new node. Before you do the reconfiguration, however, you need to perform some initial steps.

After these initial steps, you should reconfigure the Resource Database. This is done by running the clsetup command on any of the nodes which is currently running the Resource Database. Since the Resource Database is synchronized across all of its nodes, the reconfiguration takes effect on all nodes. The steps are as follows:

1. Log in to any node where the Resource Database is running as the root user.
2. If this node is not the same one where you made the backup, then copy the backup to this node. Then run the clsetup command with the -a and -g options to reconfigure the database. The syntax in this case is as follows:

   /etc/opt/FJSVcluster/bin/clsetup -a *cfname -g file*

   cfname is the CF name of the new node to be added, and *file* is the name of the backup file without the .tar.* suffix. * represents the extension of the type of tar compression (Z or gz).

   For example, suppose that you want to add a new node whose CF name is fuji4 to a cluster. If the backup file on an existing node is named */mydir/*rdb.tar.Z, then the following command would cause the Resource Database to be configured for the new node:

   # cd /etc/opt/FJSVcluster/bin/

   # ./clsetup -a fuji4 -g */mydir/***rdb.tar.Z**

   If clsetup is successful, then you should immediately make a new backup of the Resource Database. This backup will include the new node in it. Be sure to save the backup to a place where it will not be lost upon a system reboot.

   If an unexpected failure such as a panic occurs, then you may need to restore the Resource Database from an earlier backup. See the Section "Restoring the Resource Database" for details.

3. To verify if the reconfiguration was successful, run the clgettree command and ensure that the new node is displayed in the output. If it is not present, then recheck the CIP configuration to see if it omitted the new node. If the problem is the CIP configuration, then you need to do the following to recover:

   a) Correct the CIP configuration on all nodes. Make sure that CIP is running with the new configuration on all nodes.

   b) Restore the Resource Database from backup.

   c) Rerun the clsetup command to reconfigure the Resource Database.

4.5.3 Configuring the Resource Database on the new node

After the Resource Database has been reconfigured on the existing nodes in the cluster, you are ready to set up the Resource Database on the new node itself.

The first step is to verify the CIP configuration on the new node. The file /etc/cip.cf should reference the new node. The file should be the same on the new node as it is on existing nodes in the cluster. If you used the Cluster Admin CF Wizard to configure CF and CIP for the new node, then CIP should already be properly configured.

You should also verify that the existing nodes in the cluster can ping the new node using the ping(1M) command or the ping6(8) command (when using the IPv6 address). If the new node has multiple CIP subnetworks, then recall that the Resource Database only uses the first one that is defined in the CIP configuration file.

After verifying that CIP is correctly configured and working, then you should do the following:

1. Log in to the new node as root.
2. Copy the latest Resource Database backup to the new node. This backup was made in Step 2 of the second list in the Section "Reconfiguring the Resource Database".
3. Run the command clsetup with the -s option. The syntax for this case is as follows:

   # /etc/opt/FJSVcluster/bin/clsetup -s *file*

   file is the name of the backup file.

If we continue our example of adding fuji4 to the cluster and we assume that the backup file rdb.tar.Z was copied to */mydir*, then the command would be as follows:

 # /etc/opt/FJSVcluster/bin/clsetup -s */mydir/***rdb.**tar.Z

If for some reason the clsetup command does not completes, then you should execute the clinitreset complete. After clinitreset completes, reboot the node and retry the clsetup command.

If the clsetup command completes successfully, then you should run the clgettree command to verify that the configuration has been set-up properly. The output should include the new node. It should also be identical to output from clgettree run on an existing node.

If the clgettree output indicates an error, then recheck the CIP configuration. If you need to change the CIP configuration on the new node, then you will need to do the following on the new node after the CIP change:

a) Run clinitreset.
 # /etc/opt/FJSVcluster/etc/bin/clinitreset

b) Reboot.

c) Rerun the clsetup command described above.

4.5.4 Adjusting StartingWaitTime

After the Resource Database has successfully been brought up in the new node, then you need to check if the StartingWaitTime used in startup synchronization is still adequate. If the new node boots much faster or slower than the other nodes, then you may need to adjust the StartingWaitTime time. Refer to the Section "Start up synchronization and the new node" for further information.

4.5.5 Restoring the Resource Database

The procedure for restoring the Resource Database is as follows:

1. Copy the file containing the Resource Database to all nodes in the cluster.
2. Shutdown all cluster nodes.
3. Reboot all nodes to single user mode.

The restore procedure absolutely requires that all nodes in the cluster must be in single user mode.

4. Restore the Resource Database on each node with the clrestorerdb command. The syntax is:

   # clrestorerdb -f *file**

   file is the backup file with the .tar. suffix omitted. * represents the extension of the type of tar compression that your Linux distribution uses (Z or gz).

   For example, suppose that a restoration was being done on a two-node cluster consisting of nodes fuji2 and fuji3, and that the backup file was copied to */mydir/*backup_rdb.tar.Z on both nodes. The command to restore the Resource Database on fuji2 and fuji3 would be as follows:

   fuji2# **cd /etc/opt/FJSVcluster/bin/**

   fuji2# **./clrestorerdb -f** */mydir/***backup_rdb.tar.Z**

   fuji3# **cd /etc/opt/FJSVcluster/bin/**

   fuji3# **./clrestorerdb -f** */mydir/***backup_rdb.tar.Z**

5. After Steps 1 through 5 have been completed on all nodes, then reboot all of the nodes with the following command:

   # **init 6**

5 GUI administration

This chapter covers the administration of features in the Cluster Foundation (CF) portion of Cluster Admin.

This chapter discusses the following:

- The Section "Overview" introduces the Cluster Admin GUI.
- The Section "Starting Cluster Admin GUI and logging in" describes logging in and shows the first windows you will see.
- The Section "Main CF table" describes the features of the main table.
- The Section "CF route tracking" details the CF route tracking GUI interface.
- The Section "Node details" explains how to get detailed information.
- The Section "Displaying the topology table" discusses the topology table, which allows you to display the physical connections in the cluster.
- The Section "Starting and stopping CF" describes how to start and stop CF.
- The Section "Marking nodes DOWN" details how to mark a node DOWN.
- The Section "Using PRIMECLUSTER log viewer" explains how to use the PRIMECLUSTER log viewer, including how to view and search syslog messages.
- The Section "Displaying statistics" discusses how to display statistics about CF operations.
- The Section "Heartbeat monitor" describes how to monitor the percentage of heartbeats that are being received by CF.
- The Section "Adding and removing a node from CIM" describes how to add and remove a node from CIM.
- The Section "Unconfigure CF" explains how to use the GUI to unconfigure CF.
- The Section "CIM Override" discusses how to use the GUI to override CIM, which causes a node to be ignored when determining a quorum.

5.1 Overview

CF administration is done by means of the Cluster Admin GUI. The following sections describe the CF Cluster Admin GUI options.

5.2 Starting Cluster Admin GUI and logging in

The first step is to start Web-based Admin View by entering the following URL in a java-enabled browser:

http://*Management_Server***:8081/Plugin.cgi**

In this example, if fuji2 is a management server, enter the following:

http://fuji2:8081/Plugin.cgi

This brings up the Web-Based Admin View main window (Figure 25).

Figure 25: Cluster Admin start-up window

Enter a user name in the *User name* field and the password and click on *OK*.

Use the appropriate privilege level while logging in. There are three privilege levels: *root* privileges, *administrative* privileges, and *operator* privileges.

With the root privileges, you can perform all actions including configuration, administration and viewing tasks. With administrative privileges, you can view as well as execute commands but cannot make configuration changes. With the operator privileges, you can only perform viewing tasks.

In this example we are using root and not creating user groups.

Click on the *Global Cluster Services* button and the *Cluster Admin* button appears (Figure 26).

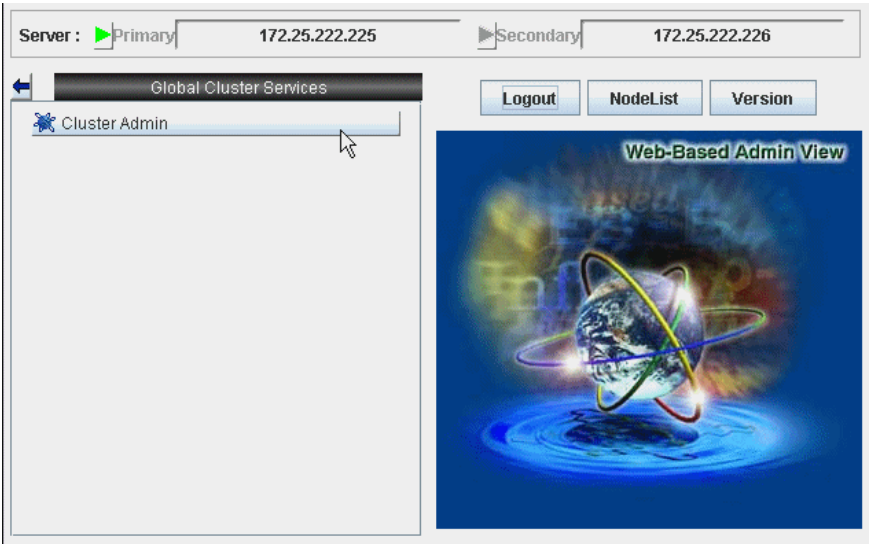

Figure 26: Cluster Admin top window

Click on the *Cluster Admin* button.

The *Choose a node for initial connection* window appears (Figure 27).

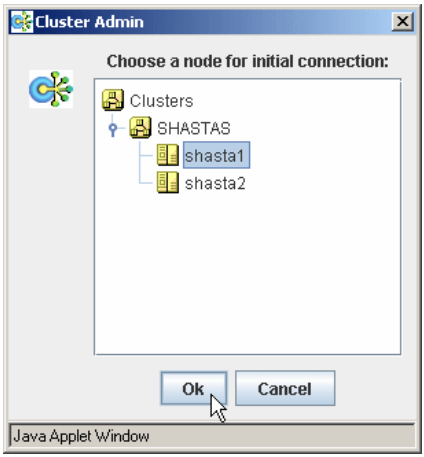

Figure 27: Initial connection choice window

Select a node and click on *Ok*.

The Cluster Admin main window appears (Figure 28).

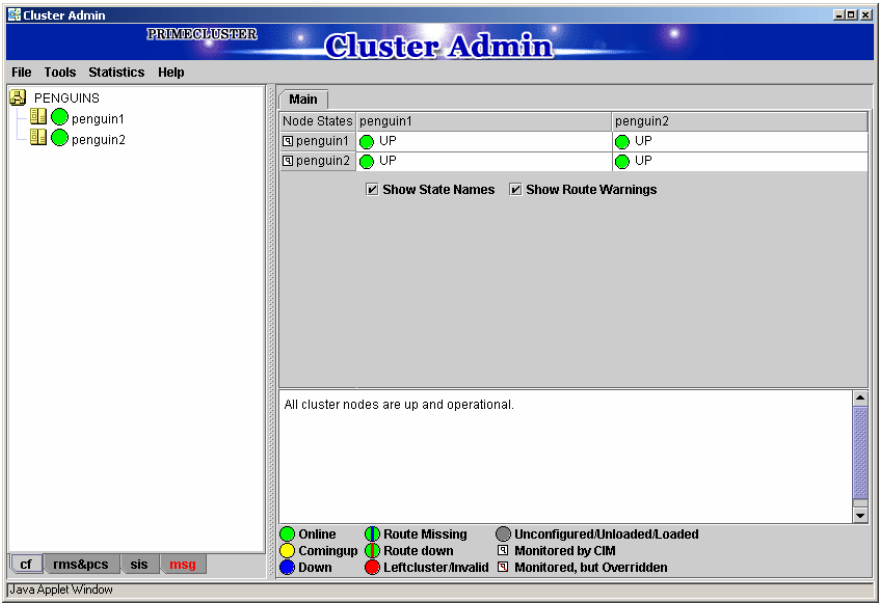

Figure 28: Cluster Admin main window

By default, the *cf* tab is selected and the CF main window is presented. Use the appropriate privilege level while logging in. The tab for RMS will appear as *rms&pcs* when PCS is installed and as *rms* in configurations where PCS is not installed.

5.3 Main CF table

When the GUI is first started, or after the successful completion of the configuration wizard, the main CF table will be displayed in the right panel. A tree showing the cluster nodes will be displayed in the left panel. An example of this display is shown in Figure 28.

The tree displays the local state of each node, but does not give information about how that node considers other nodes. If two or more nodes disagree about the state of a node, one or more colored exclamation marks appear next to the node. Each exclamation mark represents the node state of which another node considers that node to be.

The table in the right panel is called the main CF table. The column on the left of the table lists the CF states of each node of the cluster as seen by the other nodes in the cluster. For instance, the cell in the second row and first column is the state of fuji3 as seen by the node fuji2.

There is an option at the bottom of the table to toggle the display of the state names. This is on by default. If this option is turned off, and there is a large number of nodes in the cluster, the table will display the node names vertically to allow a larger number of nodes to be seen.

There are two types of CF states. Local states are the states a node can consider itself in. Remote states are the states a node can consider another node to be in. Table 1 lists the local states.

CF state	Description
UNLOADED	The node does not have a CF driver loaded.
LOADED	The node has a CF driver loaded, but is not running.
COMINGUP	The node is in the process of starting and should be UP soon.
UP	The node is up and running normally.
INVALID	The node has an invalid configuration and must be reconfigured.
UNKNOWN	The GUI has no information from this node. This can be temporary, but if it persists, it probably means the GUI cannot contact that node.

Table 1: Local states

CF state	Description
UNCON-FIGURED 	The CF driver is loaded but the node is not yet configured to run CF.

Table 1: Local states

Table 2 lists the remote states.

CF state	Description
UP 	The node is up and part of this cluster.
DOWN 	The node is down and not in the cluster.
UNKNOWN 	The reporting node has no opinion on the reported node.
LEFTCLUSTER 	The node has left the cluster unexpectedly, probably from a crash. To ensure cluster integrity, it will not be allowed to rejoin until marked DOWN.

Table 2: Remote states

5.4 CF route tracking

If a node is UP, but it has one or more DOWN routes, the green circle in the main CF table will have a red line through it (Figure 29).

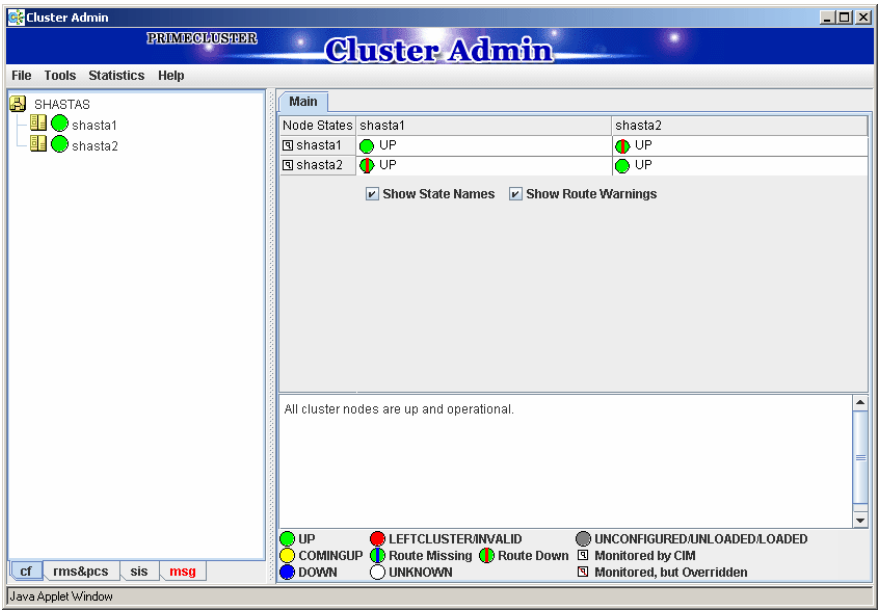

Figure 29: CF route DOWN

In this example, one of the network interfaces on fuji2 has been unplugged. Cluster Admin, therefore, shows that a route is DOWN. Since fuji3 cannot contact fuji2 over that interface, it also shows that there is a route down on fuji2. To see which routes are DOWN, click on the node in the left-panel tree and look at the route table.

If CF starts with one or more interfaces missing, then the green circle in the main CF table will have a blue line through it (Figure 30).

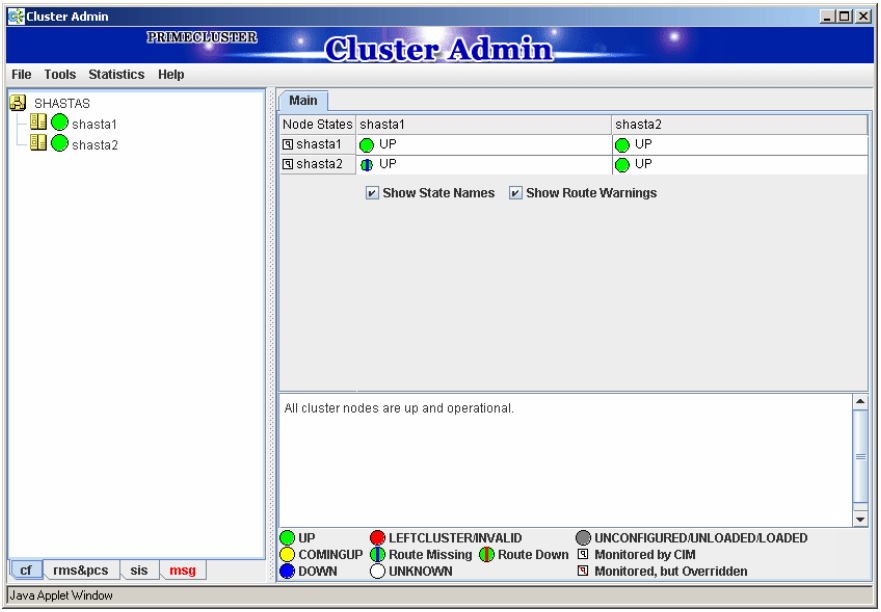

Figure 30: CF interface missing

In Figure 30, fuji3 has a broken connection to fuji2, and Cluster Admin indicates that a route is missing.

In our example, clicking on fuji2 in the left-panel tree shows that there is no route from fuji2 to the eth2 interface on fuji3 (Figure 31).

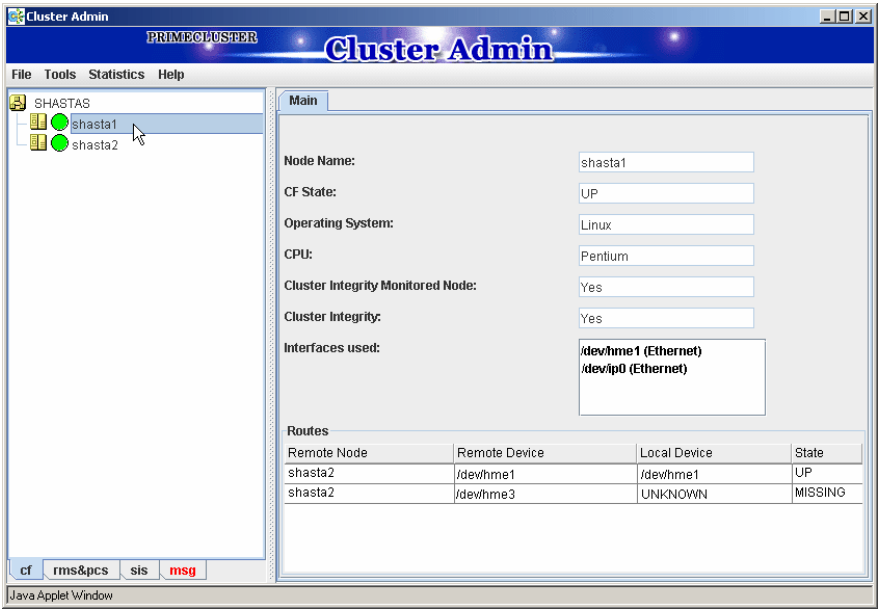

Figure 31: CF route table

5.5 Node details

To get detailed information on a cluster node, left-click on the node in the left tree. This replaces the main table with a display of detailed information. (To bring the main table back, left-click on the cluster name in the tree.)

The panel displayed is similar to the display in Figure 32.

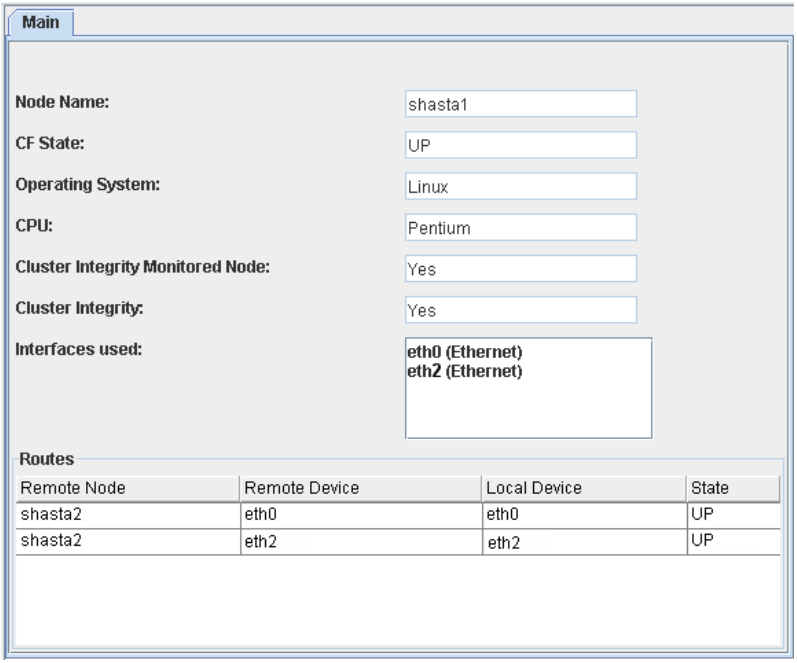

Figure 32: CF node information

Shown are the node's name, its CF state(s), operating system, platform, and the interfaces configured for use by CF. The states listed will be all of the states the node is considered to be in. For instance, if the node considers itself UNLOADED and other nodes consider it DOWN, DOWN/UNLOADED will be displayed.

The bottom part of the display is a table of all of the routes being used by CF on this node. It is possible for a node to have routes go down if a network interface or interconnect fails, while the node itself is still accessible.

5.6 Displaying the topology table

To examine and diagnose physical connectivity in the cluster, select *Tools -> Topology*. This menu option will produce a display of the physical connections in the cluster. This produces a table with the nodes shown along the left side and the interconnects of the cluster shown along the top. Each cell of the table lists the interfaces on that node connected to the interconnect. There is also a checkbox next to each interface showing if it is being used by CF. This table makes it easy to locate cabling errors or configuration problems at a glance.

An example of the topology table is shown in Figure 33.

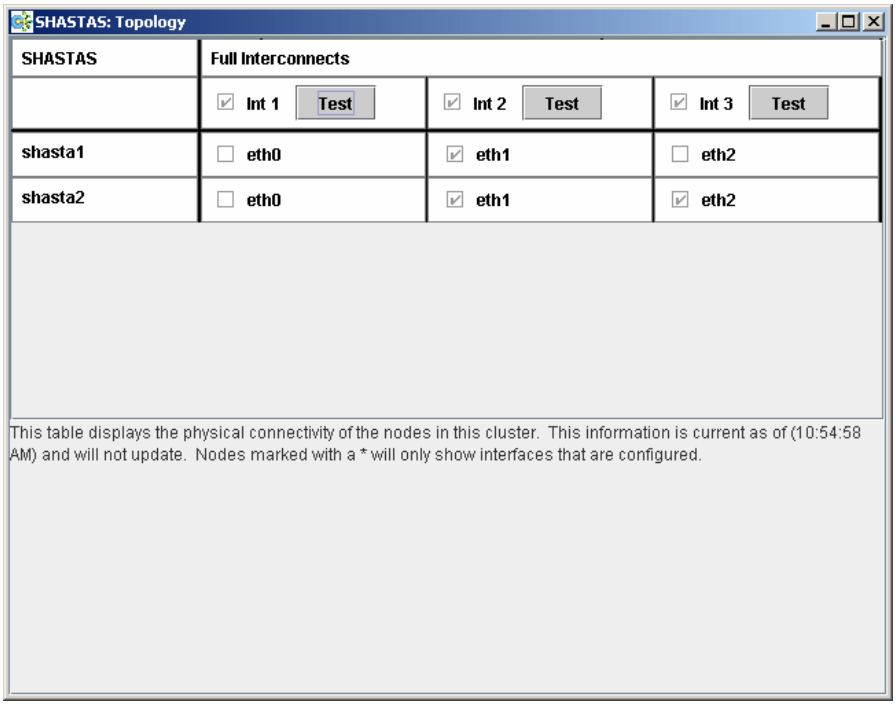

Figure 33: CF topology table

Pressing the *Test* button launches the Response Time monitor.

This tool allows you to see the response time for any combination of two nodes on that interconnect (Figure 34).

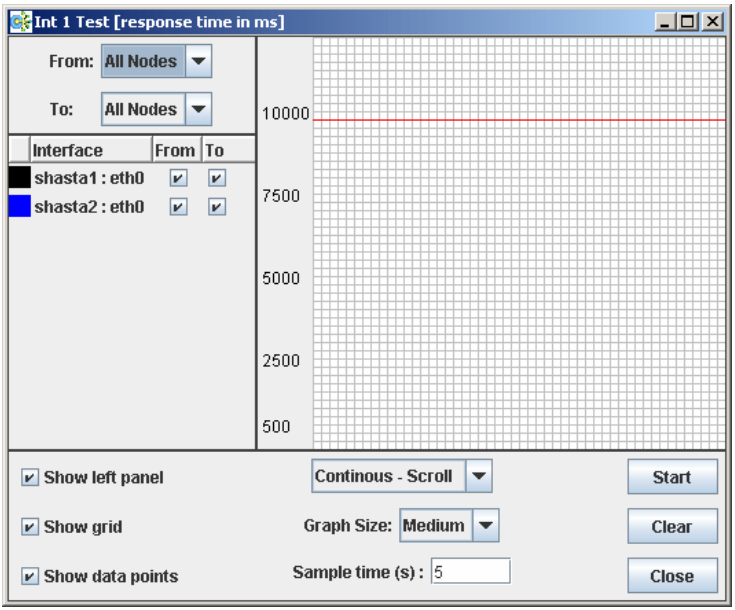

Figure 34: Response Time monitor

The Y axis is the response time for CF pings in milliseconds and the X axis is a configurable period. The red line is the upper limit of the response time before CF will declare nodes to be in the LEFTCLUSTER state.

The controls to the left of the graph determine the nodes for which the graph displays data as follows:

- Set the selection boxes at the top to a specific node name, or to *All Nodes*.
- Select the check boxes next to the node names to specify specific nodes.

The controls on the left of the bottom panel control how the graphing and information collection is done as follows:

- Check the *Show left panel* check box to hide the left panel to provide more room for the graph.
- Check the *Show grid* check box to turn the grid on and off.
- Check the *Show data points* check box to display a simple line graph.

The controls in the middle of the bottom panel are as follows:

- The top drop-down menu controls how the graph is drawn. The following options are available:
  - *Continuous-Scroll*—Creates a continuous graph, so that when there are more data points than space, the graph scrolls.
  - *Continuous-Clear*—Graphs continuously until the graph is full, and then it starts a new graph.
  - *Single Graph*— Draws a single graph only.
- *Graph size*—Allows you to control how many data points are drawn.
- *Sample time*—Controls how often data points are taken.
- The buttons on the lower right control starting and stopping of the graph, clearing it, and closing the graph window.

The buttons on the right of the bottom panel are as follows:

- *Start/Stop*—Starts or stops the Response Time Monitor.
- *Clear*—Clears the data and starts a new graph.
- *Close*—Closes the Response Time Monitor and returns you to the CF Main screen.

The Response Time Monitor is a tool for expert users such as consultants or skilled customers. Its output must be interpreted carefully. The Response Time Monitor uses user-space CF pings to collect its data. If the CF traffic between nodes in a cluster is heavy, then the Response Time Monitor may show slow response times, even if the cluster and the interconnects are working properly. Likewise, if a user does CF pings from the command line while the Response Time Monitor is running, then the data may be skewed.

For best results, the Response Time Monitor should be run at times when CF traffic is relatively light, and the CF nodes are only lightly loaded.

5.7 Starting and stopping CF

There are two ways that you can start or stop CF from the GUI. The first is to simply right-click on a particular node in the tree in the left-hand panel. A state sensitive pop-up menu for that node will appear. If CF on the selected node is in a state where it can be started (or stopped), then the menu choice *Start CF* (or *Stop CF*) will be offered. Figure 35 shows the content-sensitive menu pop-up when you select *Start CF*.

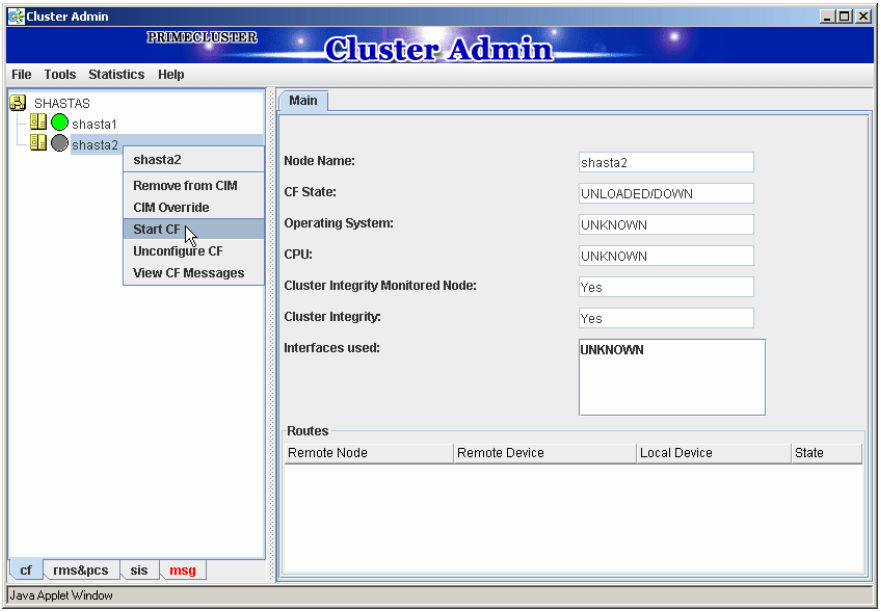

Figure 35: Starting CF

You can also go to the *Tools* pull-down menu and select either *Start CF* or *Stop CF* (Figure 36). A pop-up listing all the nodes where CF may be started or stopped will appear (Figure 37). You can then select the desired node to carry out the appropriate action.

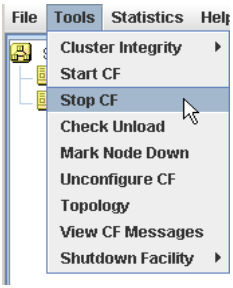

Figure 36: Starting and stopping CF from the Tools menu

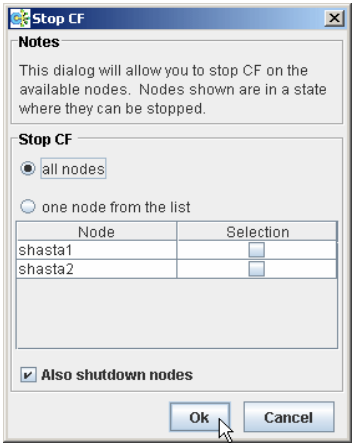

Figure 37: Stopping CF and shutting down all nodes

The CF GUI gets its list of CF nodes from the node you selected as the initial connection node (Figure 27). If CF is not up and running on the initial connection node, then the CF GUI will not display the list of nodes in the tree in the left panel.

Because of this, if you do not choose the *all nodes* option, and you want to stop CF on multiple nodes (including the initial node) by means of the GUI, ensure that the initial connection node is the last one on which you stop CF.

5.7.1 Starting CF

If CF is stopped on the initial connection node, the Cluster Admin main window appears with the CF options of *Load driver* or *Unconfigure* (Figure 38). The CF state must be UNLOADED or LOADED to start CF on a node.

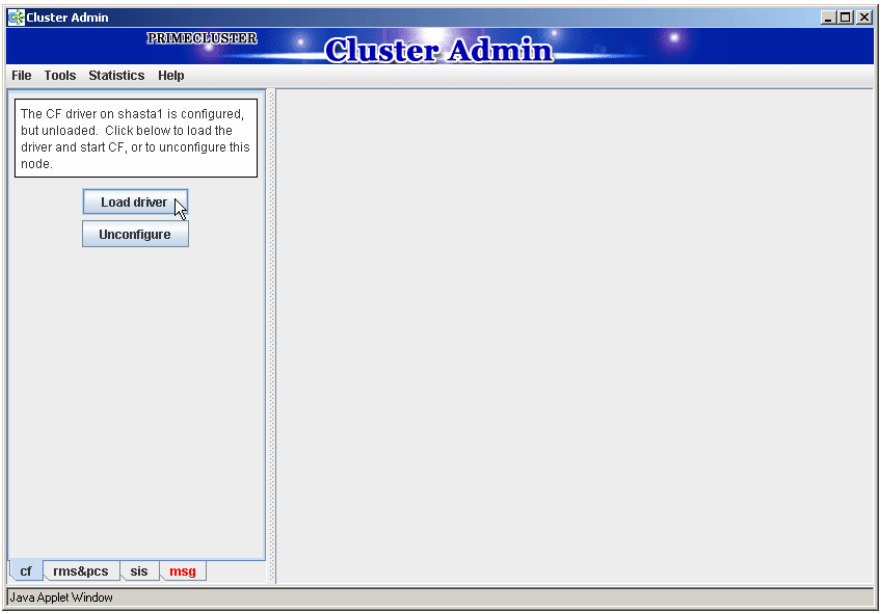

Figure 38: CF configured but not loaded

Click on the *Load driver* button to start the CF driver with the existing configuration.

The Start CF services popup appears (Figure 39). By default all CF services that have been installed on that node are selected to be started. The contents of this list may vary according to the installed products.

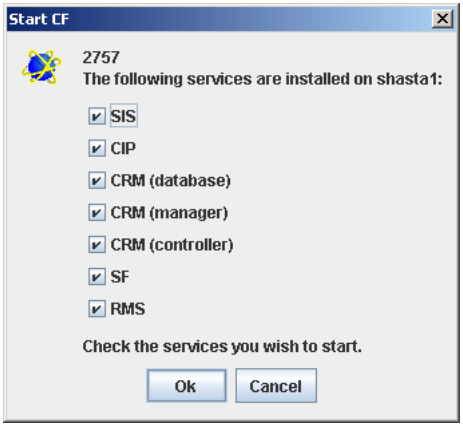

Figure 39: Start CF services pop-up

You may exclude CF services from startup by clicking on the selection check box for each service that you do not want to start. This should be done by experts only.

Click on the *Ok* button and a status popup appears with the results of each service start operation (Figure 40).

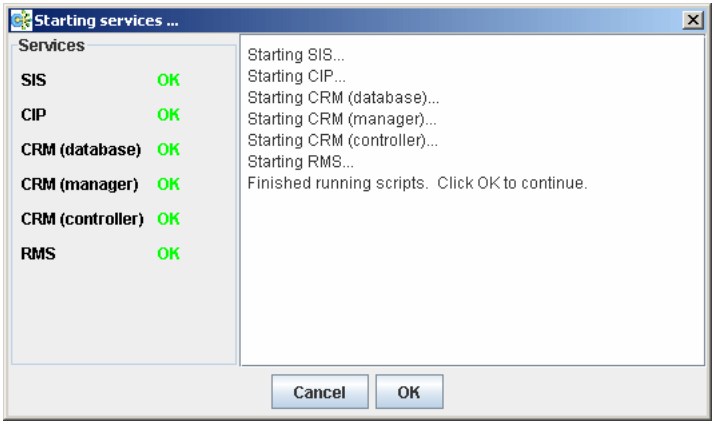

Figure 40: Start CF services status window

Click on the *Ok* button to return to the Cluster Admin main window.

5.7.2 Stopping CF

Right-click on a CF node name and select *Stop CF* (Figure 41).

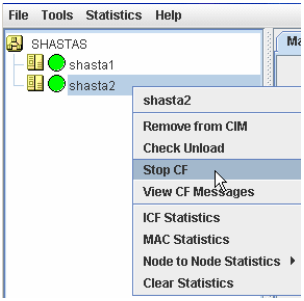

Figure 41: Stop CF

A confirmation pop-up appears (Figure 42). Choose *Yes* to continue.

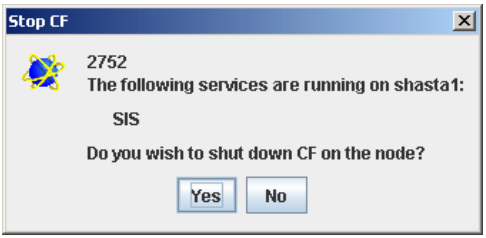

Figure 42: Stopping CF

Before stopping CF, all services that run over CF on that node should first be shut down. When you invoke *Stop CF* from the GUI, it will use the CF dependency scripts to see what services are still running. It will print out a list of these in a pop-up and ask you if you wish to continue. If you do continue, it will then run the dependency scripts to shut down these services. If any service does not shutdown, then the *Stop CF* operation will fail.

The dependency scripts currently include only PRIMECLUSTER products. If third-party products, for example Oracle RAC, are using PAS or CF services, then the GUI will not know about them. In such cases, the third-party product should be shut down before you attempt to stop CF.

To stop CF on a node, the node's CF state must be UP, COMINGUP, or INVALID.

5.8 Marking nodes DOWN

If a node is shut down normally, it is considered DOWN by the remaining nodes. If it leaves the cluster unexpectedly, it will be considered LEFTCLUSTER. It is important to mark a node DOWN as SOON as possible to allow normal cluster operation for the remaining nodes. The menu option *Tools->Mark Node Down* allows nodes to be marked as DOWN.

Marking a node DOWN should be only done if the node is actually down (inoperable or inoperative); otherwise, this could cause data corruption.

To do this, select *Tools->Mark Node Down*. This displays a dialog of all of the nodes that consider another node to be LEFTCLUSTER. Clicking on one of them displays a list of all the nodes that node considered LEFTCLUSTER. Select one and then click *OK*. This clears the LEFTCLUSTER status on that node.

Refer to the Chapter "LEFTCLUSTER state" for more information on the LEFTCLUSTER state.

5.9 Using PRIMECLUSTER log viewer

The CF log messages for a given node may be displayed by right-clicking on the node in the tree and selecting *View CF Messages*.

Alternately, you may go to the *Tools* menu and select *View CF Messages*. This brings up a pop-up where you can select the node whose syslog messages you would like to view.

When invoked from within CF, the PRIMECLUSTER log viewer only displays CF syslog messages. To view messages from other products, select the *Products* button in the *Product Filter* window pane (Figure 43).

Figure 43 shows an example of the PRIMECLUSTER log viewer.

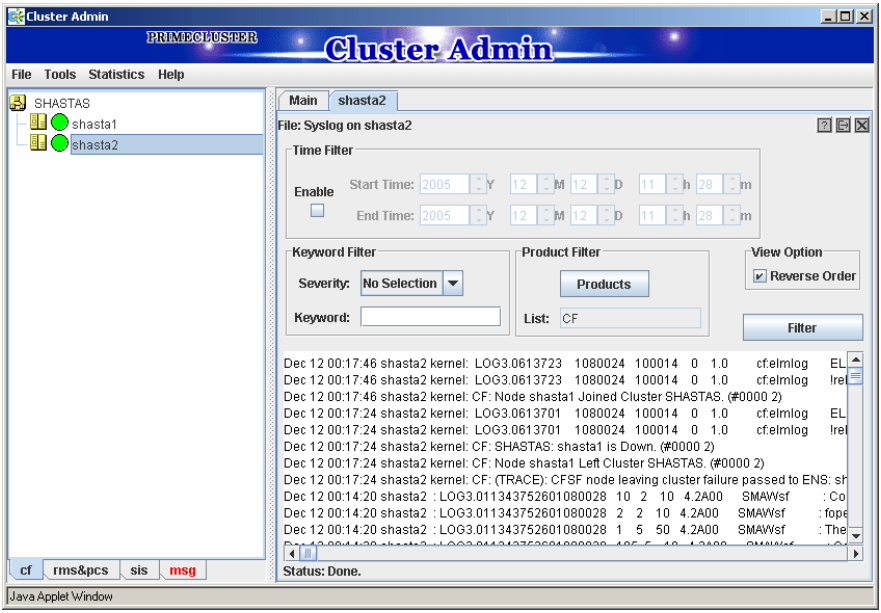

Figure 43: PRIMECLUSTER log viewer

The messages appear in the right-hand panel. If you click on the *Products* button, then only the messages appear for the product that you select. To list the error messages for all of the installed PRIMECLUSTER products, choose *All messages*. Your choice is then listed in the *List* field.

The PRIMECLUSTER log viewer has search filters based on date/time/keyword and severity levels.

The *Reverse Order* checkbox is selected by default. This option reverses the order of the messages. To disable this feature, deselect the checkbox.

5.9.1 Search based on time filter

To perform a search based on a start and end time, click the check box for *Enable*, specify the start and end times for the search range, and click on the *Filter* button (Figure 44).

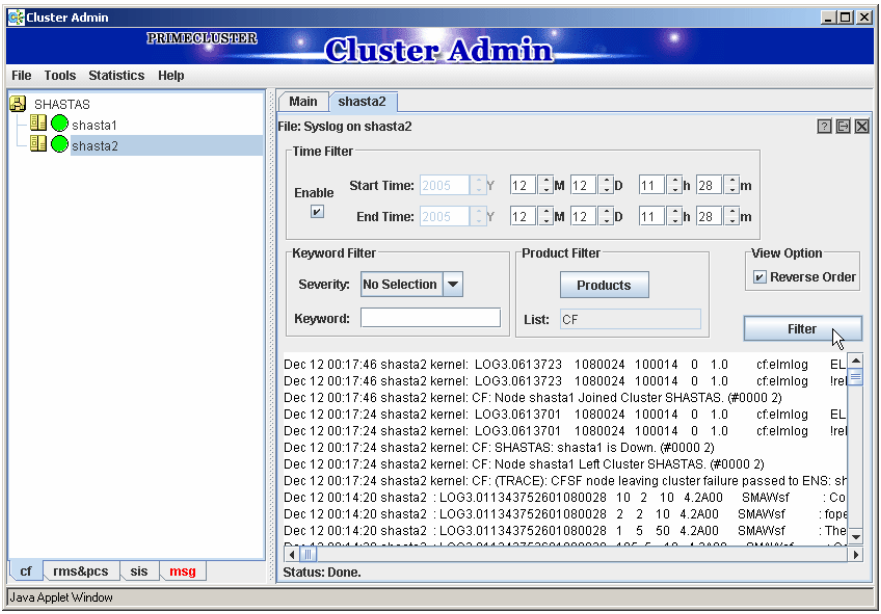

Figure 44: Search based on date/time

5.9.2 Search based on keyword

To perform a search based on a keyword, enter a keyword and click on the *Filter* button (Figure 45).

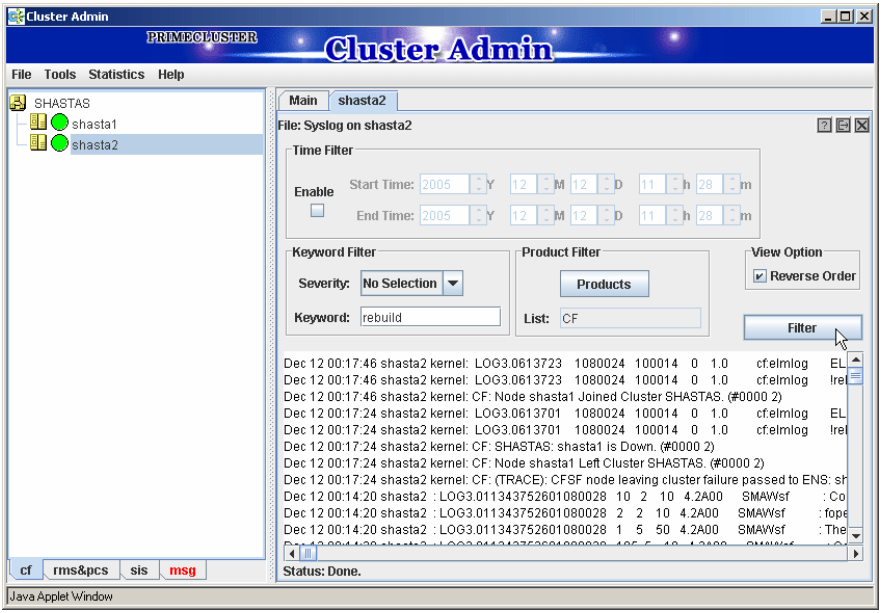

Figure 45: Search based on keyword

5.9.3 Search based on severity levels

To perform a search based severity levels, click on the *Severity* pull-down menu. You can choose from the severity levels shown in Table 3 and click on the *Filter* button. Figure 46 shows the log for a search based on severity level.

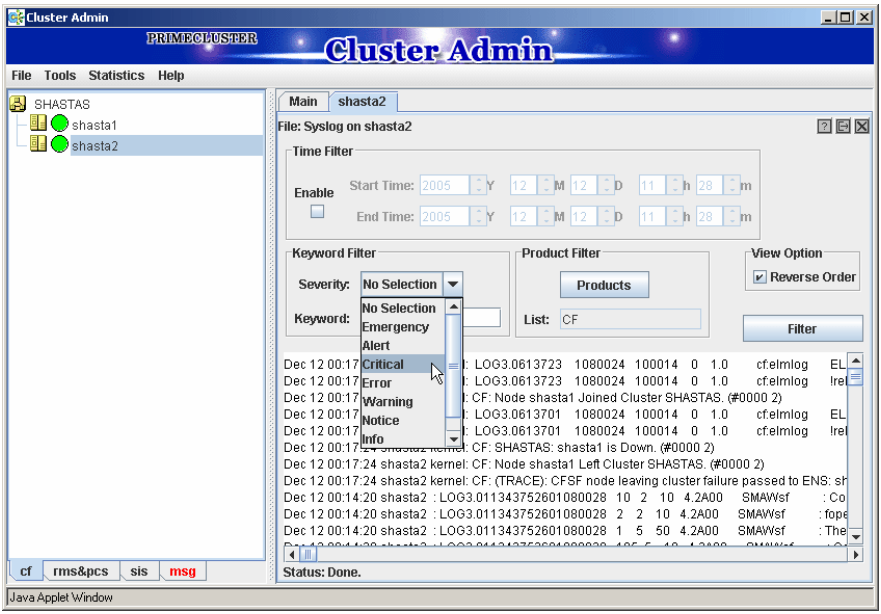

Figure 46: Search based on severity

Severity level	Severity description
Emergency	Systems cannot be used.
Alert	Immediate action is necessary.
Critical	Error that makes it impossible for the associated PRIMECLUSTER product to continue running.
Error	Error condition that arises unexpectedly, causing the associated PRIMECLUSTER function to terminate abnormally.
Warning	Minor error that does not terminate the offending function.

Table 3: PRIMECLUSTER log viewer severity levels

Severity level	Severity description
Notice	Normal but important condition
Info	Provides information on the status of a PRIME-CLUSTER operation.
Debug	Verbose message that provides more information on why an error condition occurred.

Table 3: PRIMECLUSTER log viewer severity levels

5.10 Displaying statistics

CF can display various statistics about its operation. There are three types of statistics available:

- ICF
- MAC
- Node to Node

To view the statistics for a particular node, right-click on that node in the tree and select the desired type of statistic.

Alternately, you can go to the *Statistics* menu and select the desired statistic. This will bring up a pop-up where you can select the node whose statistics you would like to view. The list of nodes presented in this pop-up will be all nodes whose states are UP as viewed from the login node.

Figure 47 shows the display window for ICF Statistics.

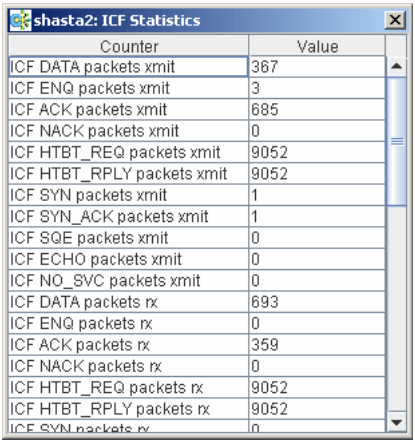

Figure 47: ICF statistics

Figure 48 shows the display window for MAC Statistics.

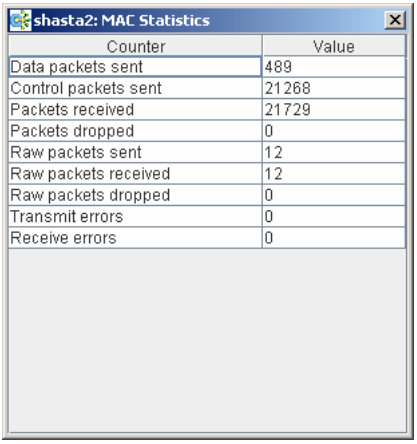

Figure 48: MAC statistics

Figure 49 shows the display window for node to node statistics.

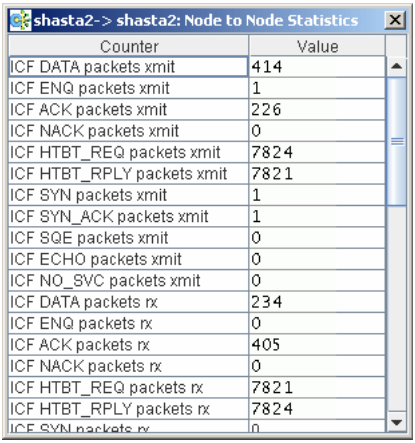

Figure 49: Node to Node statistics

The statistics counters for a node can be cleared by right-clicking on a node and selecting *Clear Statistics* from the command pop-up. The *Statistics* menu also offers the same option.

5.11 Heartbeat monitor

To display the Heartbeat monitor, go to the *Statistics* menu and select *Heartbeat Monitor* (Figure 50).

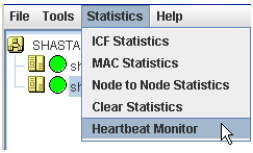

Figure 50: Selecting the Heartbeat monitor

The Heartbeat monitor allows you to monitor the percentage of heartbeats that are being received by CF over time. On a healthy cluster, this is normally close to 100 percent.

The Y axis is the percentage of heartbeats that have been successfully received and the X axis is a configurable time interval (Figure 51).

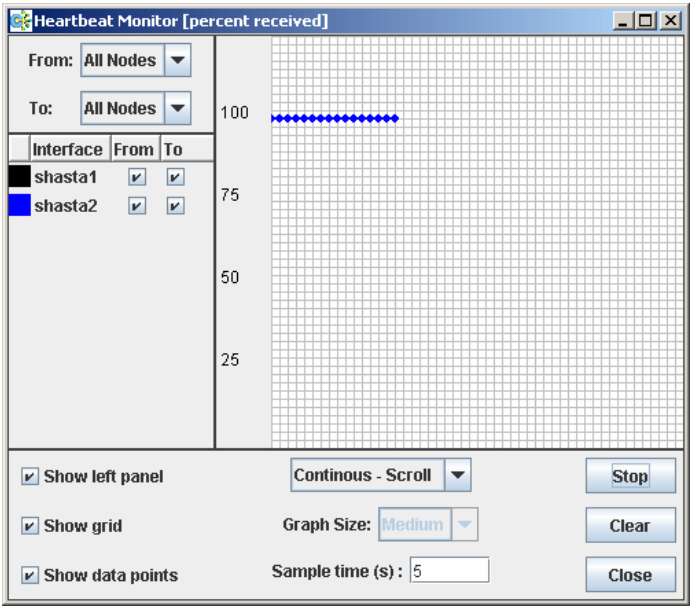

Figure 51: Heartbeat monitor

The controls on the left panel determine which data the graph shows as follows:

- The selection boxes at the top can be set to an individual node, or to *All Nodes*.
- The check boxes below the selection boxes allow the enabling and disabling of specific nodes.

The controls on the left of the bottom panel control how the graphing and information collection is done as follows:

- The *Show left panel* check box hides the left panel to provide more room for the graph.
- The *Show grid* check box turns the grid on and off.
- The *Show data points* check box can be turned off to display a simple line graph.

The controls in the bottom panel are as follows:

- The drop-down menu below the graph controls how the graph is drawn. The following options are available:
  - *Continuous-Scroll*—creates a continuous graph, so that when there are more data points than space, the graph scrolls.
  - *Continuous-Clear*—graphs continuously, but when the graph is full, clears it and starts a new graph.
  - *Single Graph*— creates a single graph only.
- *Graph size*—allows you to control how many data points are drawn.
- *Sample time*—controls how often data points are taken.
- The buttons on the lower right control starting and stopping of the graph, clearing it, and closing the graph window.

5.12 Adding and removing a node from CIM

To add a node to CIM, click on the *Tools* pull-down menu. Select *Cluster Integrity* and *Add to CIM* from the expandable pull-down menu (Figure 52).

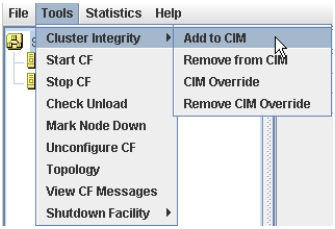

Figure 52: CIM options

The *Add to CIM* pop-up display appears. Choose the desired CF node and click on *Ok* (Figure 53).

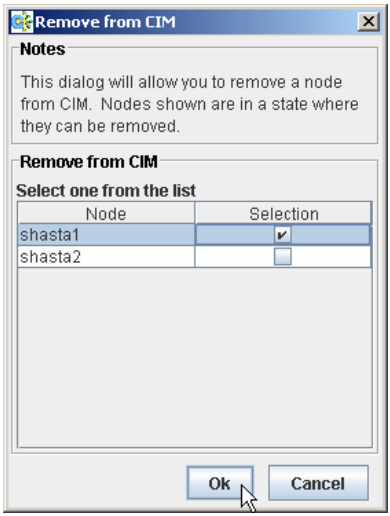

Figure 53: Add to CIM

To remove a node from CIM by means of the *Tools* pull-down menu, select *Cluster Integrity* and *Remove from CIM* from the expandable pull-down menu. Choose the CF node to be removed from the pop-up and click on *Ok*. A node can be removed at any time.

Refer to the Section "Cluster Integrity Monitor" for more details on CIM.

5.13 Unconfigure CF

To unconfigure a CF node, first stop CF on that node. Then, from the *Tools* pull-down menu, click on *Unconfigure CF*.

The *Unconfigure CF* pop-up display appears. Select the check box for the CF node to unconfigure, and click on *Ok* (Figure 54).

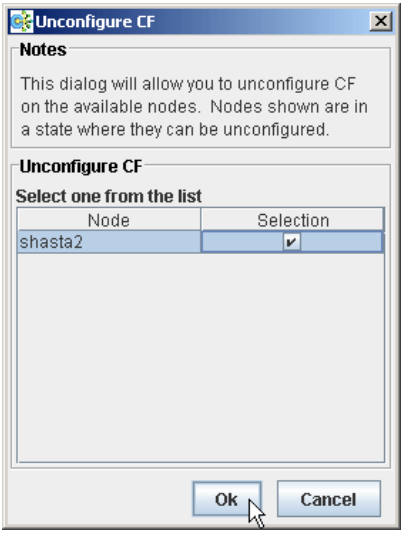

Figure 54: Unconfigure CF

The unconfigured node will no longer be part of the cluster. However, other cluster nodes will still show that node as DOWN until they are rebooted.

5.14 CIM Override

The CIM Override option causes a node to be ignored when determining a quorum. A node cannot be overridden if its CF state is UP. To select a node for CIM Override, right-click on a node and choose *CIM Override* (Figure 55).

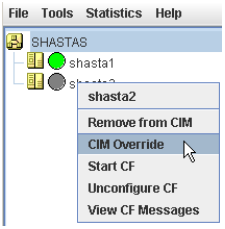

Figure 55: CIM Override

A confirmation pop-up appears (Figure 56).

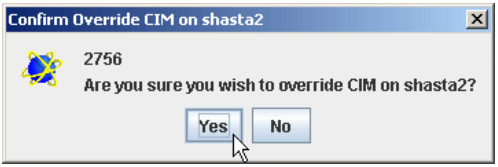

Figure 56: CIM Override confirmation

Click *Yes* to confirm.

Setting CIM override is a temporary action. It may be necessary to remove it manually again. This can be done by right-clicking on a node and selecting *Remove CIM Override* from the menu (Figure 57).

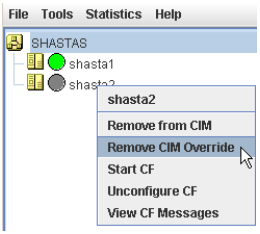

Figure 57: Remove CIM Override

CIM override is automatically removed when a node rejoins the cluster.

6 LEFTCLUSTER state

This chapter defines and describes the LEFTCLUSTER state.

This chapter discusses the following:

- The Section "Description of the LEFTCLUSTER state" describes the LEFTCLUSTER state in relation to the other states.
- The Section "Recovering from LEFTCLUSTER" discusses the different ways a LEFTCLUSTER state is caused and how to clear it.

Occasionally, while CF is running, you may encounter the LEFTCLUSTER state, as shown by running the cftool -n command. A message will be printed to the console of the remaining nodes in the cluster. This can occur under the following circumstances:

- Broken interconnects—All cluster interconnects going to another node (or nodes) in the cluster are broken.
- Panicked nodes—A node panics.
- Reboot—Shutting down a node with the reboot command.

Nodes running CF should normally be shut down with the shutdown command or with the init command. These commands will run the rc scripts that will allow CF to be cleanly shut down on that node. If you run the reboot -f, halt -f, or poweroff -f command, the rc scripts are not run, and the node will go down while CF is running. This will cause the node to be declared to be in the LEFTCLUSTER state by the other nodes.
When a node comes back up after being rebooted and attempts to rejoin the cluster, the join process automatically changes the node's state from LEFTCLUSTER to DOWN so that it can rejoin the cluster.

If the Shutdown Facility (SF) is fully configured and running on all cluster nodes, it will try to resolve the LEFTCLUSTER state automatically. If SF is not configured and running, or the SF fails to clear the state, the state has to be cleared manually. This section explains the LEFTCLUSTER state and how to clear this state manually.

6.1 Description of the LEFTCLUSTER state

Each node in a CF cluster keeps track of the state of the other nodes in the cluster. For example, the other node's state may be UP, DOWN, or LEFTCLUSTER.

LEFTCLUSTER is an intermediate state between UP and DOWN, which means that the node cannot determine the state of another node in the cluster because of a break in communication.

For example, consider the three-node cluster shown in Figure 58.

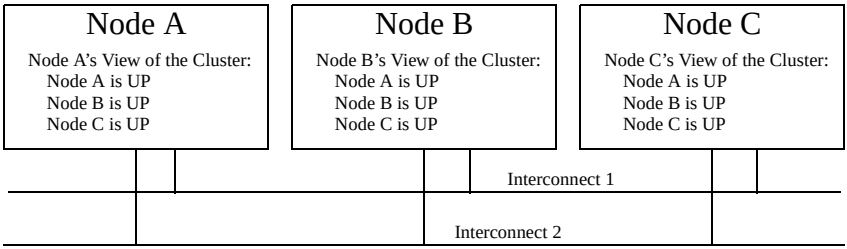

Figure 58: Three-node cluster with working connections

Each node maintains a table of what states it believes all the nodes in the cluster are in.

Now suppose that there is a cluster partition in which the connections to Node C are lost. The result is shown in Figure 59.

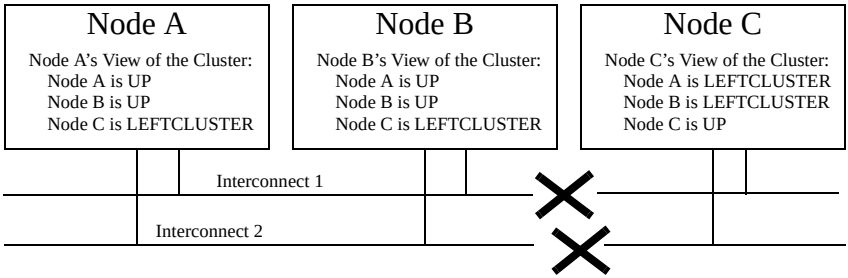

Figure 59: Three-node cluster where connection is lost

Because of the break in network communications, Nodes A and B cannot be sure of Node C's true state. They therefore update their state tables to say that Node C is in the LEFTCLUSTER state. Likewise, Node C cannot be sure of the true states of Nodes A and B, so it marks those nodes as being in the LEFTCLUSTER in its state table.

LEFTCLUSTER is a state that a particular node believes other nodes are in. It is never a state that a node believes that it is in. For example, in Figure 59, each node believes that it is UP.

The purpose of the LEFTCLUSTER state is to warn applications which use CF that contact with another node has been lost and that the state of such a node is uncertain. This is very important for RMS.

For example, suppose that an application on Node C was configured under RMS to fail over to Node B if Node C failed. Suppose further that Nodes C and B had a shared disk to which this application wrote.

RMS needs to make sure that the application is, at any given time, running on either Node C or B but not both, since running it on both would corrupt the data on the shared disk.

Now suppose for the sake of argument that there was no LEFTCLUSTER state, but as soon as network communication was lost, each node marked the node it could not communicate with as DOWN. RMS on Node B would notice that Node C was DOWN. It would then start an instance of the application on Node C as part of its cluster partition processing. Unfortunately, Node C isn't really DOWN. Only communication with it has been lost. The application is still running on Node C. The applications, which assume that they have exclusive access to the shared disk, would then corrupt data as their updates interfered with each other.

The LEFTCLUSTER state avoids the above scenario. It allows RMS and other application using CF to distinguish between lost communications (implying an unknown state of nodes beyond the communications break) and a node that is genuinely down.

When SF notices that a node is in the LEFTCLUSTER state, it contacts the previously configured Shutdown Agent and requests that the node which is in the LEFTCLUSTER state be shut down. With PRIMECLUSTER, a weight calculation determines which node or nodes should survive and which ones should be shut down. SF has the capability to arbitrate among the shutdown requests and shut down a selected set of nodes in the cluster, such that the subcluster with the largest weight is left running and the remaining subclusters are shutdown.

In the example given, Node C would be shut down, leaving Nodes A and B running. After the SF software shuts down Node C, SF on Nodes A and B clear the LEFTCLUSTER state such that Nodes A and B see Node C as DOWN. Refer to the Chapter "Shutdown Facility" for details on configuring SF and shutdown agents.

Note that a node cannot join an existing cluster when the nodes in that cluster believe that the node is in the LEFTCLUSTER state. Therefore, any nodes in LEFTCLUSTER state have to be recovered before they can join an existing cluster.

6.2 Recovering from LEFTCLUSTER

When a node comes back up after being rebooted and attempts to rejoin the cluster, the join process automatically changes the node's state from LEFTCLUSTER to DOWN so that it can rejoin the cluster. When this occurs, the join server initially sees the node that is attempting to join the cluster as being in the LEFTCLUSTER state. The join server signals the joining node that it is busy because the joining node is not in the DOWN state. It then notifies all of the remaining nodes in the cluster that the joining node is DOWN and to start the node-down processing, which must be completed before the node is allowed to rejoin the cluster. The joining node continues retrying to join the cluster until the node-down processing is completed on all of the cluster nodes at which time the joining node is allowed to rejoin the cluster.

If SF is not running on all nodes, or if SF is unable to shut down the node which left the cluster, and the LEFTCLUSTER condition occurs, then the system administrator must manually clear the LEFTCLUSTER state. The procedure for doing this depends on how the LEFTCLUSTER condition occurred.

6.2.1 Caused by a panic/hung node

The LEFTCLUSTER state may occur because a particular node panicked or hung. In this case, the procedure to clear LEFTCLUSTER is as follows:

1. Make sure the node is really down. If the node panicked and came back up, proceed to Step 2. If the node is in the debugger, exit the debugger. The node will reboot if it panicked, otherwise shut down the node, called the *offending node* in the following discussion.

2. Use the Cluster Admin GUI to log into one of the running nodes in the cluster while the offending node is suspended. Go to the CF main window and select *Mark Node Down* from the *Tools* pull-down menu, then mark the offending node as DOWN. This may also be done from the command line by using the following command:

   # **cftool -k**

3. Reboot the offending node, it should automatically join the cluster.

The state of the node is automatically changed from the state of LEFTCLUSTER in the state of DOWN so that the node may enter again when the offending node is reactivated.

6.2.2 Caused by staying in the kernel debugger too long

In Figure 60, Node C was placed in the kernel debugger too long so it appears as a hung node. Nodes A and B decided that Node C's state was LEFTCLUSTER.

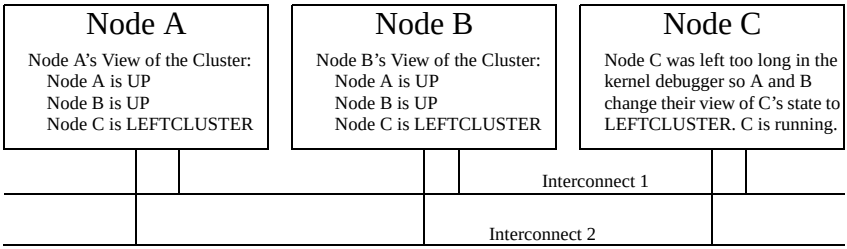

Figure 60: Node C placed in the kernel debugger too long

To recover from this situation, you would need to do the following:

1. Shut down Node C, and bring it back up.
2. If Node C fails to join the cluster and remains in the LEFTCLUSTER state after being shutdown and coming back up, start up the Cluster Admin on Node A or B. Use *Mark Node Down* from the *Tools* pull-down menu in the CF portion of the GUI to mark Node C DOWN.
3. The node should successfully join the cluster.

6.2.3 Caused by a cluster partition

A cluster partition s refer to a communications failure in which all CF communications between sets of nodes in the cluster are lost. In this case, the cluster itself is effectively partitioned into sub-clusters.

To manually recover from a cluster partition, you must do the following:

1. Decide which of the sub-clusters you want to survive. Typically, you will chose the sub-cluster that has the largest number of nodes in it or the one where the most important hardware is connected or the most important application is running.
2. Shut down all of the nodes in the sub-cluster which you don't want to survive.
3. Fix the network break so that connectivity is restored between all nodes in the cluster.
4. Bring the nodes back up.

5. If the nodes fail to join the cluster and remain in the LEFTCLUSTER state after being shutdown and coming back up, use the Cluster Admin GUI to log on to one of the surviving nodes and run the CF portion of the GUI. Select *Mark Node Down* from the *Tools* menu to mark all of the shut down nodes as DOWN.
6. The nodes should successfully join the cluster.

For example, consider Figure 61.

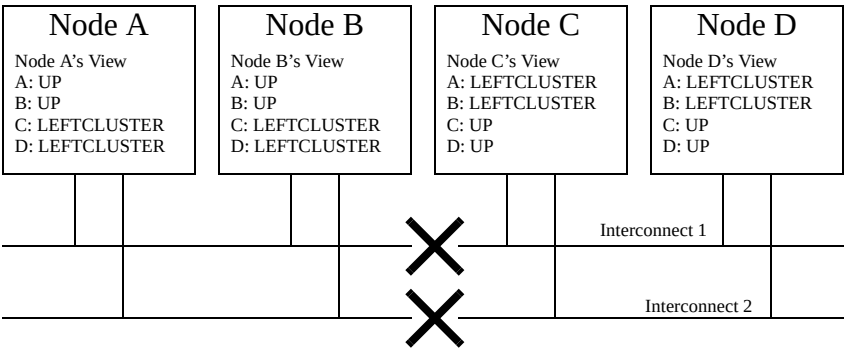

Figure 61: Four-node cluster with cluster partition

In Figure 61, a four-node cluster has suffered a cluster partition. Both of its CF interconnects (Interconnect 1 and Interconnect 2) have been severed. The cluster is now split into two sub-clusters. Nodes A and B are in one sub-cluster while Nodes C and D are in the other.

To recover from this situation, in instances where SF fails to resolve the problem, you would need to do the following:

1. Decide which sub-cluster you want to survive. In this example, let us arbitrarily decide that Nodes A and B will survive.
2. Shut down all of the nodes in the other sub-cluster, here Nodes C and D.
3. Fix the interconnect break on Interconnect 1 and Interconnect 2 so that both sub-clusters will be able to communicate with each other again.
4. Bring Nodes C and D back up.
5. If the LEFTCLUSTER state persists on Nodes C or D, run the Cluster Admin GUI on either Node A or Node B. Start the CF portion of the GUI and go to *Mark Node Down* from the *Tools* pull-down menu. Mark any nodes still in the LEFTCLUSTER state as DOWN.

7 CF topology table

This chapter discusses the CF topology table as it relates to the CF portion of the Cluster Admin GUI.

This chapter discusses the following:

- The Section "Basic layout" discusses the physical layout of the topology table.
- The Section "Selecting devices" discusses how the GUI actually draws the topology table.
- The Section "Examples" shows various network configurations and what their topology tables would look like.

The CF topology table is part of the CF portion of the Cluster Admin GUI. The topology table may be invoked from the *Tools->Topology* menu item in the GUI (refer to the Section "Displaying the topology table" in the Chapter "GUI administration"). It is also available during CF configuration in the CF Wizard in the GUI.

The topology table is designed to show the network configuration from perspective of CF. It shows what devices are on the same interconnects and can communicate with each other.

The topology table only considers Ethernet devices. It does not include any IP interconnects that might be used for CF, even if CF over IP is configured.

Displayed devices

The topology table is generated by doing CF pings on all nodes in the cluster and then analyzing the results.

The rest of this chapter discusses the format of the topology table. The examples implicitly assume that all devices can be seen on each node. Again, this would be the case when first configuring a CF cluster.

7.1 Basic layout

The basic layout of the topology table is shown in Table 4.

FUJI	Full interconnects		Partial interconnects		Unconnected devices
	Int 1	Int 2	Int 3	Int 4	
fuji2	eth0 eth2	eth1	eth3	eth5	eth4 eth6
fuji3	eth0	eth2	missing	eth1	
fuji4	eth1	eth2	eth3	missing	eth4

Table 4: Basic layout for the CF topology table

The upper-left-hand corner of the topology table gives the CF cluster name. Below it, the names of all of the nodes in the cluster are listed.

The CF devices are organized into three major categories:

- Full interconnects—Have working CF communications to each of the nodes in the cluster.
- Partial interconnects—Have working CF communications to at least two nodes in the cluster, but not to all of the nodes.
- Unconnected devices—Have no working CF communications to any node in the cluster.

If a particular category is not present, it will be omitted from the topology table. For example, if the cluster in Table 4 had no partial interconnects, then the table headings would list only full interconnects and unconnected devices (as well as the left-most column giving the clustername and node names).

Within the full interconnects and partial interconnects category, the devices are further sorted into separate interconnects. Each column under an Int number heading represents all the devices on an interconnect. (The column header *Int* is an abbreviation for *Interconnect*.) For example, in Table 4, there are two full interconnects listed under the column headings of Int 1 and Int 2.

Each row for a node represents possible CF devices for that node.

Thus, in Table 4, Interconnect 1 is a full interconnect. It is attached to eth0 and eth2 on fuji2. On fuji3, it is attached to eth0, and on fuji4, it is attached to eth1.

Since CF runs over Ethernet devices, the eth*n* devices in Table 4 represent the Ethernet devices found on the various systems. The actual names of these devices will vary depending on the type of Ethernet controllers on the system. For nodes whose CF driver was loaded with -L, only configured devices will be shown.

It should be noted that the numbering used for the interconnects is purely a convention used only in the topology table to make the display easier to read. The underlying CF product does not number its interconnects. CF itself only knows about CF devices and point-to-point routes.

If a node does not have a device on a particular partial interconnect, then the word missing will be printed in that node's cell in the partial interconnects column. For example, in Table 4, fuji3 does not have a device for the partial interconnect labeled Int 3.

7.2 Selecting devices

The basic layout of the topology table is shown in Table 5. However, when the GUI actually draws the topology table, it puts check boxes next to all of the interconnects and CF devices as shown in Table 5.

FUJI	**Full interconnects**		**Partial interconnects**		**Unconnected devices**
	☒ Int 1	**☒ Int 2**	**☐ Int 3**	**☐ Int 4**	
fuji2	☒ eth0 ☐ eth2	☒ eth1	☐ eth3	☐ eth5	☐ eth4 ☐ eth6
fuji3	☒ eth0	☒ eth2	missing	☐ eth1	
fuji4	☒ eth1	☒ eth2	☐ eth3	missing	☐ eth4

Table 5: Topology table with check boxes shown

The check boxes show which of the devices were selected for use in the CF configuration. (In the actual topology table, check marks appear instead of x's.)

When the topology table is used outside of the CF Wizard, these check boxes are read-only. They show what devices were previously selected for the configuration. In addition, the unchecked boxes (representing devices which were not configured for CF) will not be seen for nodes where -L was used to load CF.

When the topology table is used within the CF Wizard, then the check boxes may be used to select which devices will be included in the CF configuration. Clicking on the check box in an Int *number* heading will automatically select all devices attached to that interconnect. However, if a node has multiple devices connected to a single interconnect, then only one of the devices will be selected.

For example, in Table 5, fuji2 has both eth0 and eth2 attached to Interconnect 1. A valid CF configuration allows a given node to have only one CF device configured per interconnect. Thus, in the CF Wizard, the topology table will only allow eth0 or eth2 to be selected for fuji2. In the above example, if eth2 were selected for fuji2, then eth0 would automatically be unchecked.

If the CF Wizard is used to add a new node to an existing cluster, then the devices already configured in the running cluster will be displayed as read-only in the topology table. These existing devices may not be changed without unconfiguring CF on their respective nodes.

7.3 Examples

The following examples show various network configurations and what their topology tables would look like when the topology table is displayed in the CF Wizard on a totally unconfigured cluster. For simplicity, the check boxes are omitted.

Example 1

In this example, there is a three-node cluster with three full interconnects (Figure 62).

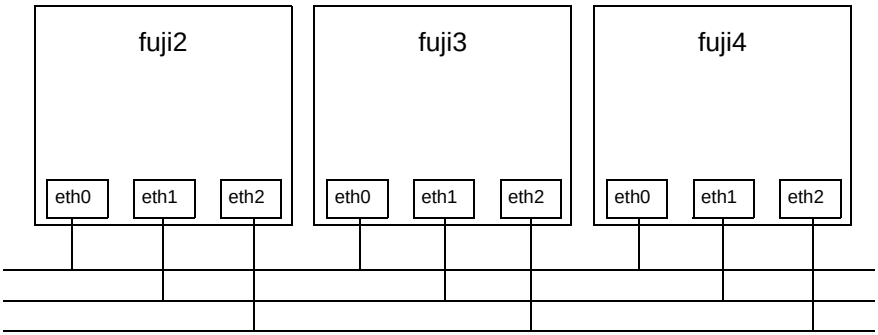

Figure 62: A three-node cluster with three full interconnects

The resulting topology table for Figure 62 is shown in Table 6.

FUJI	Full interconnects		
	Int 1	**Int 2**	**Int 3**
fuji2	eth0	eth1	eth2
fuji3	eth0	eth1	eth2
fuji4	eth0	eth1	eth2

Table 6: Topology table for 3 full interconnects

Since there are no partial interconnects or unconnected devices, those columns are omitted from the topology table.

Example 2

In this example, fuji2's Ethernet connection for eth1 has been broken (Figure 63).

The resulting topology table for Figure 63 is shown in Table 7.

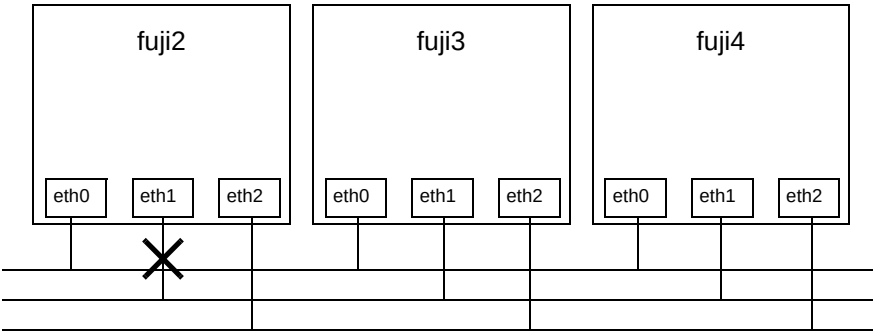

Figure 63: Broken ethernet connection for eth1 on fuji2

FUJI	Full interconnects		Partial interconnects	Unconnected devices
	Int 1	**Int 2**	**Int 3**	
fuji2	eth0	eth2	missing	eth1
fuji3	eth0	eth2	eth1	
fuji4	eth0	eth2	eth1	

Table 7: Topology table with broken Ethernet connection

In Table 7, eth1 for fuji2 now shows up as an unconnected device. Since one of the interconnects is missing a device for fuji2, the Partial Interconnect column now shows up. Note that the relationship between interconnect numbering and the devices has changed between Table 6 and Table 7. In Table , for example, all eth1 devices were on Int 2. In Table 6, the eth1 devices for Nodes B and C are now on the partial interconnect Int 3. This change in numbering illustrates the fact that the numbers have no real significance beyond the topology Table 7.

Example 3

This example shows a cluster with severe networking or cabling problems in which no full interconnects are found.

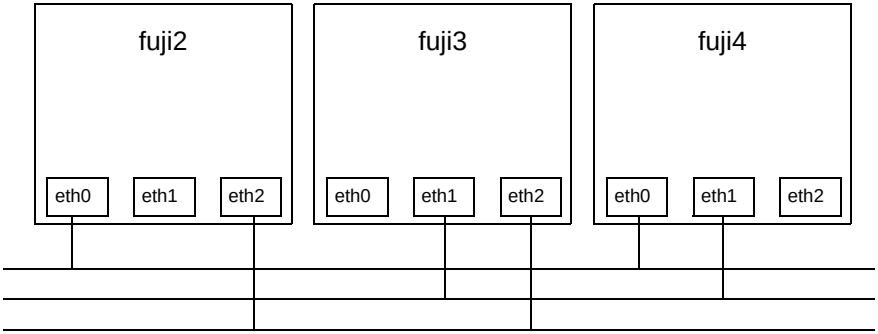

Figure 64: Cluster with no full interconnects

The resulting topology table for Figure 64 is shown in Table 8.

FUJI	Partial interconnects			Unconnected devices
	Int 1	Int 2	Int 3	
fuji2	eth0	missing	eth2	eth1
fuji3	missing	eth1	eth2	eth0
fuji4	eth0	eth1	missing	eth2

Table 8: Topology table with no full interconnects

In Table 8, the full interconnects column is omitted since there are none. Note that if this configuration were present in the CF Wizard, the wizard would not allow you to do configuration. The wizard requires that at least one full interconnect must be present.

8 Shutdown Facility

This chapter describes the components and advantages of PRIMECLUSTER Shutdown Facility (SF) and provides administration information.

Certain product options are region-specific. For information on the availability a specific Shutdown Agent (SA), contact field engineers.

This chapter discusses the following:

- The Section "Overview" describes the components of SF.
- The Section "Available SAs" describes the available agents for use by the SF.
- The Section "SF split-brain handling" describes the methods for resolving split cluster situations.
- The Section "Configuring the Shutdown Facility" describes the configuration of SF and its agents.
- The Section "SF administration" provides information on administering SF.
- The Section "Logging" describes the log files used by SF and its agents.

8.1 Overview

The SF provides the interface for managing the shutdown of cluster nodes when error conditions occur. The SF also advises other PRIMECLUSTER products of the successful completion of node shutdown so that recovery operations can begin.

The SF is made up of the following major components:

- The Shutdown Daemon (SD)
- One or more Shutdown Agents (SA)
- sdtool(1M) command

Shutdown Daemon

The SD is started at system boot time and is responsible for the following:

- Monitoring the state of all cluster nodes
- Monitoring the state of all registered SAs
- Reacting to indications of cluster node failure and verifying or managing node elimination
- Resolving split-brain conditions
- Notifying other PRIMECLUSTER products of node elimination completion

The SD uses SAs to perform most of its work with regard to cluster node monitoring and elimination. In addition to SA's, the SD interfaces with the Cluster Foundation layer's ENS system to receive node failure indications and to advertise node elimination completion.

Shutdown Agents

The SA’s role is to attempt to shut down a remote cluster node in a manner in which the shutdown can be guaranteed. Some of the SAs are shipped with the SF product, but may differ based on the architecture of the cluster node on which SF is installed. SF allows any PRIMECLUSTER service layer product to shut down a node whether RMS is running or not.

An SA is responsible for shutting down, and verifying the shutdown of a cluster node. Each SA uses a specific method for performing the node shutdown such as:

- SA_wtinps uses an IPS unit.
- SA_blade provides an SA for the Fujitsu Technology Solutions Blade servers.
- SA_IPMI supplies an SA for IPMI-based systems.
- SA_lkcd provides an SA that uses the kernel panic status of other nodes.
- SA_mmb provides an SA that uses the management board (MMB) on PRIMEQUEST nodes.
- SA_vmgp provides an SA of the system which uses the virtual machine function to PRIMEQUEST 400/500 series.
- SA_vmSPgp and SA_vmSPgr provide an SA of the system which uses the Xen virtual machine function to PRIMEQUEST 1000 series.

- SA_icmp provides an SA that checks whether a node to be stopped is in the active or inactive state by using a network route.
- SA_vmchkhost provides an SA of the system which uses the Xen/KVM virtual machine function.
- SA_libvirtgp and SA_libvirtgr provide an SA of the system which uses the KVM virtual machine function.
- SA_sunF is the SA for Sun Fire system controllers.

The Section "Available SAs" discuss SAs in more detail.

If more than one SA is used, the first SA in the configuration file is used as the primary SA. SD always uses the primary SA. The other secondary SAs are used as fall back SAs only if the primary SA fails for some reason.

sdtool command

The sdtool(1M) utility is the command line interface for interacting with the SD. With it the administrator can:

- Start and stop the SD (although this is typically done with an RC script run at boot time)
- View the current state of the SA's
- Force the SD to reconfigure itself based on new contents of its configuration file
- Dump the contents of the current SF configuration
- Enable/disable SD debugging output
- Eliminate a cluster node

Although the sdtool(1M) utility provides a cluster node elimination capability, the preferred method for controlled shutdown of a cluster node is the /sbin/shutdown command.

8.2 Available SAs

This section describes the following set of supported SAs:

- IPS—Internet Power Switch (Unsupport)
- SNMP—Simple Network Management Protocol (Unsupport)

- Blade
- IPMI—Intelligent Platform Management Interface
- kdump
- MMB—PRIMEQUEST management board
- vmgp
- vmSP
- ICMP
- VMCHKHOST
- libvirt
- Sun Fire

8.2.1 IPS

The Internet Power Switch (IPS) SA is SA_wtinps. This SA provides a node shutdown function using the Western Telematic Inc. Internet Power Switch (WTI IPS) unit to power-cycle selected nodes in the cluster.

Setup and configuration

The WTI IPS unit must be configured according to the directions in the manual shipped with the unit. At the very least, an IP address must be assigned to the unit and a password must be enabled. Make sure that the cluster node's power plugs are plugged into the IPS box and that the command confirmation setting on the IPS box is set to on.

It is advisable to have the IPS box on a robust LAN connected directly to the cluster nodes.

The boot delay of every configured plug in the IPS box should be set to 10 seconds.

If you want to set the boot delay to any other value, make sure that the "timeout value" for the corresponding SA_wtinps agent should be set such that it is greater than this boot delay value by at least 10 seconds. To set this value, use the detailed configuration mode for SF.

If more than a single plug is assigned to a single node (which means that more than one plug will be operated per /on, /off, /boot command), the "boot delay" of these plugs must be assigned to a value larger than 10 seconds, otherwise timeouts may occur. The timeout value of the corresponding SA_wtinps should be set as follows:

timeout = *boot_delay* + (*2* 2 * *no_of_plugs*) + 10

The log file is stored in the following:

/var/opt/SMAWsf/log/SA_wtinps.log

8.2.2 SNMP

SNMP is a function of the unsupport in this version.

The SNMP Shutdown Agent, SA_snmp, provides a Shutdown Agent for specific machines, using SNMP switchable devices and the SNMP agents present on the system. The SF product uses SA_snmp to provide a node shutdown facility for these nodes.

The machines should be configured according to the directions in the manual or manuals shipped with the unit. For SA_snmp to work properly, ensure the following:

- Check whether the software containing an SNMP package is properly installed. This is required for the proper functioning of the SNMP switchable device.
- Ensure that the machines can communicate with each other by means of SNMP commands. This includes the setting of proper security groups and communities in the SNMP configuration. Make a note of the SNMP community string that has both read and write permissions. This string is to be mentioned in the SA_snmp configuration files.

The log file is stored in the following:

/var/opt/SMAWsf/log/SA_snmp.log

8.2.3 Blade

The Blade Shutdown Agent, SA_blade, provides a Shutdown Agent for the Fujitsu Technology Solutions Blade servers. This is used in the SF product to provide a node shutdown facility for these nodes.

Setup and configuration

The Blade server should be configured according to the directions in the manual or manuals shipped with the unit.

For SA_blade to work properly, ensure the following:

- Check whether ServerView software, containing an SNMP package, is properly installed.
- Ensure that the server Blades can talk to the management Blade by means of SNMP commands. This includes the setting of proper security groups and communities in the SNMP configuration. Make a note of the SNMP community string that has both read and write permissions. This string is to be mentioned in the SA_blade configuration files. This means that the Blade Servers can both read and write (or change) the data in the SNMP MIB's on the management station.

The log file is stored in the following:

/var/opt/SMAWsf/log/SA_blade.log

8.2.4 IPMI

For complete configuration details, refer to the appropriate hardware manual that came with your system.

Setup and configuration

Before using the Intelligent Platform Management Interface (IPMI) SA you must make some configuration changes in the BIOS and use the Remote View CD. The following is an example of the changes that need to be made:

Your configuration settings might differ:

1. In the BIOS, select *Console Redirection* and disable it.
2. Next, boot to the Remote View CD. Press the [Esc] key to enter DOS mode.
3. In the RTDS directory, run the ipmiview.exe program.
4. Select *User Management* and create a user called *administrator*.
5. Press the [Enter] or [Return] key to select *<ENTER> Configure User*. Assign a password and select *1 - EnableUser* for *Operation*.
6. Press the [Esc] key to go back to the previous screen.
7. Press the [F2] key to select *<F2> Configure User Access*.
8. Select *802.3_LAN*
9. You must enter *4* for *Privilege Limit*.

The log file is stored in the following:

/var/opt/SMAWsf/log/SA_ipmi.log

8.2.5 kdump

The kdump shutdown agent SA_lkcd is the SA that can be used when kdump is used. After another node panics, this SA executes high-speed switchover while a crash dump is being collected by kdump.

The log file is stored in the following:

/var/opt/SMAWsf/log/SA_lkcd.log

8.2.6 MMB

The MMB shutdown agent uses the MMB of PRIMEQUEST to provide shutdown mechanisms for nodes.

The MMB shutdown agent provides the following two shutdown mechanisms:

- SA_mmbp—Shutdown mechanism that triggers panic in nodes through the MMB.
- SA_mmbr—Shutdown mechanism that resets nodes through the MMB.

Setup and configuration

To operate the MMB shutdown agent properly, you must install and set up the HBA blockage function and the PSA/SVmco. The setup procedure is described below.

1. Installing the HBA blockage function and PSA/SVmco

   The HBA blockage function and the PSA/SVmco report node status changes that take place through the MMB to the shutdown facilities. Install the HBA blockage function and the PSA/SVmco before you set up the shutdown facilities. For installation instructions, refer to the following manuals:

   – *Attachment Driver Guide* that comes with the computer

   – *PRIMEQUEST 500/400 SERIES INSTALLATION MANUAL*

   – *PRIMEQUEST 1000 Series Installation Manual*

   – *PRIMEQUEST 1000 Series ServerView Mission Critical Option User Manual*

2. Setting up the PSA/SVmco and the MMB units

   The PSA/SVmco and MMB must be set up so that node status changes are reported properly to the shutdown facility through the MMB units. Set up the PSA/SVmco units before setting up the shutdown facility. For setup instructions, refer to the following manuals:

   – *PRIMEQUEST 500/400 SERIES INSTALLATION MANUAL*

   – *PRIMEQUEST 1000 Series Installation Manual*

   – *PRIMEQUEST 1000 Series ServerView Mission Critical Option User Manual*

   You must create an RMCP user so that PRIMECLUSTER can link with the MMB units.

   In all PRIMEQUEST instances that make up the PRIMECLUSTER system, be sure to create a user who uses RMCP to control the MMB. To create a user who uses RMCP to control the MMB, log in to MMB Web-UI, and create the user from the Remote Server Management window of the *Network Configuration* menu. Create the user as shown below.

   – Set *Privilege* to "Admin."

   – Set *Status* to "Enabled."

   For details about creating a user who uses RMCP to control the MMB, see the "*Reference Manual: GUI/Commands,*" which comes with the computer.

   The user name created here and the specified password are used when the shutdown facility is set up. Record the user name and the password.

The MMB units have two types of users:

– User who controls all MMB units

– User who uses RMCP to control the MMB

The user created here is the user who uses RMCP to control the MMB. Be sure to create the correct type of user.

3. Setting up the HBA blockage function

   Be sure to carry out this setup if you are using a shared disk.

   If a panic occurs, the HBA blockage function closes the HBA connected to the shared disk and aborts I/O processing to the shared disk. By doing so, this function maintains the data integrity in the shared disk and makes high-speed switchover possible.

   In all nodes, specify the device bus path of the shared disk (the GDS device path if GDS is being used) in the HBA blockage function command to add the shared disk as a target in which the HBA function is to be stopped. If GDS is being used, carry out this set after you complete the GDS setup. For setup instructions, see the *Attachment Driver Guide* that comes with the computer.

4. Setting the I/O completion wait time

   Some shared disk unit require a fixed I/O completion wait time, which is the time from when a node fails until the new operation mode starts operating. This I/O completion wait time maintains the integrity of I/O operations to shared disks if a node fails due to a panic or similar condition and failover occurs.

   Although 0 second is set as the initial value of the I/O completion wait time, change this value to an appropriate value if you are using a shared disk unit that requires an I/O completion wait time.

   [Reference] All Fujitsu RAID devices do not require an I/O completion wait time. Therefore this setup is unnecessary.

   Carry out this setup after you complete the CF setup. For setup instructions, see "Setting the I/O Completion Wait Time" in the PRIMECLUSTER *Installation and Administration Guide*.

   When an I/O completion wait time is set, the switchover time if a node failure, such as a panic condition, occurs, increases by that amount of time.

The log file is stored in the following:

/var/opt/SMAWsf/log/SA_mmbp.log

/var/opt/SMAWsf/log/SA_mmbr.log

8.2.7 vmgp

The vmgp SA, SA_vmgp, provides the shutdown function for nodes (guest OS) in systems that are using the virtual machine function to PRIMEQUEST 400/500 series.

The log file is stored in the following:

/var/opt/SMAWsf/log/SA_vmgp.log

8.2.8 vmSP

The Shutdown Agent of Virtual Machine Service Provider (vmSP) provides the shutdown function for nodes (guest OS) in systems that are using the Xen virtual machine function to PRIMEQUEST 1000 series.

The Shutdown Agent of Virtual Machine Service Provider (vmSP) provides the following two shutdown facilities:

- SA_vmSPgp—SF that panics nodes (guest OS)
- SA_vmSPgr—SF that resets nodes (guest OS)

The log file is stored in the following:

/var/opt/SMAWsf/log/SA_vmSPgp.log

/var/opt/SMAWsf/log/SA_vmSPgr.log

8.2.9 ICMP

The ICMP shutdown agent, SA_icmp, checks whether a node to be stopped is in the active or inactive state by using a network route.

If there is no response from the node to be stopped in all specified network routes, it determines that the node to be stopped is in the inactive state and terminates normally. If any responses are returned from the node in one or more network routes, it determines that the node to be stopped is in the active state and terminates abnormally.

The log file is stored in the following:

/var/opt/SMAWsf/log/SA_icmp.log

8.2.10 VMCHKHOST

The VMCHKHOST shutdown agent, SA_vmchkhost, is the SA that can be used when the function, which switches the Host OS in the event of an error, is used on the Xen/KVM virtual machine function.

The log file is stored in the following:

/var/opt/SMAWsf/log/SA_vmchkhost.log

8.2.11 libvirt

The libvirt shutdown agent provides the shutdown function for nodes (guest OS) in systems that are using the KVM virtual machine function.

There are two types of the libvirt shutdown agent as follows:

- SA_libvirtgp—SF that panics nodes (guest OS)
- SA_libvirtgr—SF that resets nodes (guest OS)

The log file is stored in the following:

/var/opt/SMAWsf/log/SA_libvirtgp.log

/var/opt/SMAWsf/log/SA_libvirtgr.log

8.2.12 Sun Fire

The Sun Fire SA, SA_sunF.cfg, file is the configuration file that provides a machine shutdown facility for the RMS product for Sun Fire system controllers.

The log file is stored in the following:

/var/opt/SMAWsf/log/SA_sunF.log

8.3 SF split-brain handling

The PRIMECLUSTER product provides the ability to gracefully resolve split-brain situations as described in this section.

8.3.1 Administrative LAN

Split-brain processing makes use of Administrative LAN. For details on setting up such a LAN, see "8.2 Changing an IP Address on the Public / Administrative LAN" in the *PRIMECLUSTER Installation and Administration Guide*. The use of Administrative LAN is optional, however it is recommended for faster and more accurate split-brain handling.

8.3.2 SF split-brain handling

A *split-brain* condition is one in which one or more cluster nodes have stopped receiving heartbeats from one or more other cluster nodes, yet those nodes have been determined to still be running. Each of these distinct sets of cluster nodes is called a sub-cluster, and when a split-brain condition occurs the Shutdown Facility has a choice to make as to which sub-cluster should remain running.

Only one of the sub-clusters in a split-brain condition can survive. The SF determines which sub-cluster is most important and allows only that sub-cluster to remain. SF determines the importance of each subcluster by calculating the total node weight and application weight of each subcluster. The subcluster with the greatest total weight survives.

Node weights are defined in the SF configuration file rcsd.cfg. Typically, you use Cluster Admin's SF Wizard to set the node weights.

Application weights are defined in RMS. Each RMS userApplication object can have a ShutdownPriority defined for it. The value of the ShutdownPriority is that application's weight. RMS calculates the total application weight for a particular node by adding up the weights of all applications that are Online on that node. If an application is switched from one node to another, its weight will be transferred to the new node.

SF combines the values for the RMS ShutdownPriority attributes and the SF weight assignments to determine how to handle a split-brain condition.

8.3.2.1 RMS ShutdownPriority attribute

RMS supports the ability to set application importance in the form of a ShutdownPriority value for each userApplication object defined within the RMS configuration. These values are combined for all userApplication objects that are Online on a given cluster node to represent the total application weight of that node. When a userApplication object is switched from one node to another, the value of that userApplication object's ShutdownPriority is transferred to the new node.

The higher the value of the ShutdownPriority attribute, the more important the application.

8.3.2.2 Shutdown Facility weight assignment

The Shutdown Facility supports the ability to define node importance in the form of a weight setting in the configuration file. This value represents a node weight for the cluster node.

The higher the node weight value, the more important the node.

Although SF takes into consideration both SF node weights and RMS application weights while performing split-brain handling, it is recommended to use only one of the weights for simplicity and ease of use. When both weights are used, split-brain handling results are much more complex.

It is recommended that you follow the guidelines in the Section "Configuration notes" for help you with the configuration.

8.3.2.3 Disabling split-brain handling

Some applications require a fast failover; however, SF split-brain handling can cause a failover delay. For such applications, it is recommended that you disable the split-brain handling in the SMAWsf software.

To disable split-brain handing, the */etc/opt/SMAW/SMAWsf/nsbm.cfg* file must be present consistently on all cluster hosts and readable by the root user. The contents of this file does not matter; however, it must be present or absent consistently on all cluster hosts.

8.3.3 Runtime processing

Spit-brain handling may be performed by the following element of the Shutdown Facility:

- The Shutdown Facility internal algorithm

This method uses the node weight calculation to determine which sub-cluster is of greater importance. The total node weight is equal to the value of the defined Shutdown Facility node weight added to the total application weight of the `Online` applications for this node as calculated within RMS.

SF internal algorithm

When the SF is selected as the split-brain resolution manager, the SF uses the node weight internally.

The SF on each cluster node identifies which cluster nodes are outside its sub-cluster and adds each one of them to an internal shutdown list. This shutdown list, along with the local nodes node weight, is advertised to the SF instances running on all other cluster nodes (both in the local sub-cluster and outside the local sub-cluster) via the admIP network defined in the SF configuration file. After the SFs on each cluster node receive the advertisements, they each calculate the heaviest sub-cluster. The heaviest sub-cluster shuts down all lower weight sub-clusters.

In addition to handling well-coordinated shutdown activities defined by the contents of the advertisements, the SF internal algorithm will also resolve split-brain if the advertisements fail to be received. If the advertisements are not received then the split-brain will still be resolved, but it may take a bit more time as some amount of delay will have to be incurred.

The split-brain resolution done by the SF in situations where advertisements have failed depends on a variable delay based on the inverse of the percentage of the available cluster weight the local sub-cluster contains. The more weight it contains the less it delays. After the delay expires (assuming the sub-cluster has not been shut down by a higher-weight sub-cluster) the SF in the sub-cluster begins shutting down all other nodes in all other sub-clusters.

If a sub-cluster contains greater than 50 percent of the available cluster weight, then the SF in that sub-cluster will immediately start shutting down all other nodes in all other sub-clusters.

8.3.4 Configuration notes

When configuring the Shutdown Facility, RMS, and defining the various weights, the administrator should consider what the eventual goal of a split-brain situation should be.

Typical scenarios that are implemented are as follows:

- Largest Sub-cluster Survival (LSS)
- Specific Hardware Survival (SHS)
- Specific Application Survival (SAS)

The weights applied to both cluster nodes and to defined applications allow considerable flexibility in defining what parts of a cluster configuration should survive a split-brain condition. Using the settings outlined below, administrators can advise the Shutdown Facility about what should be preserved during split-brain resolution.

Largest Sub-cluster Survival

In this scenario, the administrator does not care which physical nodes survive the split, just that the maximum number of nodes survive. If RMS is used to control applications, it will move the applications to the surviving cluster nodes after split-brain resolution has succeeded.

This scenario is achieved as follows:

- By means of Cluster Admin, set the SF node weight values to 1. 1 is the default value for this attribute, so new cluster installations may simply ignore it.
- By means of the RMS Wizard Tools, set the RMS attribute ShutdownPriority of all userApplications to 0. 0 is the default value for this attribute, so if you are creating new applications you may simply ignore this setting.

As can be seen from the default values of both the SF weight and the RMS ShutdownPriority, if no specific action is taken by the administrator to define a split-brain resolution outcome, LSS is selected by default.

Specific Hardware Survival

In this scenario, the administrator has determined that one or more nodes contain hardware that is critical to the successful functioning of the cluster as a whole.

This scenario is achieved as follows:

- Using Cluster Admin, set the SF node weight of the cluster nodes containing the critical hardware to values more than double the combined value of cluster nodes not containing the critical hardware.
- Using PCS or the RMS Wizard Tools, set the RMS attribute ShutdownPriority of all userApplications to 0. 0 is the default value for this attribute so if you are creating new applications you may simply ignore this setting.

As an example, in a four-node cluster in which two of the nodes contain critical hardware, set the SF weight of those critical nodes to 10 and set the SF weight of the non-critical nodes to 1. With these settings, the combined weights of both non-critical nodes will never exceed even a single critical node.

Specific Application Survival

In this scenario, the administrator has determined that application survival on the node where the application is currently Online is more important than node survival. This can only be implemented if RMS is used to control the application(s) under discussion. This can get complex if more than one application is deemed to be critical and those applications are running on different cluster nodes. In some split-brain situations, all applications will not survive and will need to be switched over by RMS after the split-brain has been resolved.

This scenario is achieved as follows:

- Using Cluster Admin, set the SF node weight values to 1. 1 is the default value for this attribute, so new cluster installations may simply ignore it.
- Using PCS or the RMS Wizard Tools, set the RMS attribute ShutdownPriority of the critical applications to more than double the combined values of all non-critical applications, plus any SF node weight.

As an example, in a four-node cluster there are three applications. Set the SF weight of all nodes to 1, and set the ShutdownPriority of the three applications to 50, 10, 10. This would define that the application with a ShutdownPriority of 50 would survive no matter what, and further that the sub-cluster containing the node on which this application was running would survive the split no matter what. To clarify this example, if the cluster nodes were A, B, C and D all with a weight of 1, and App1, App2 and App3 had ShutdownPriority of 50, 10 and 10 respectively, even in the worst-case split that node D with App1 was split from nodes A, B and C which had applications App2 and App3 the weights of the sub-clusters would be D with 51 and A,B,C with 23. The heaviest sub-cluster (D) would win.

8.4 Configuring the Shutdown Facility

This section describes how to use Cluster Admin and the CLI to configure the Shutdown Facility (SF).

8.4.1 Invoking the Configuration Wizard

This section describes how to use Cluster Admin to configure SF.

Use the *Tools* pull-down menu to select *Shutdown Facility*, and then choose *Configuration Wizard* to invoke the SF Configuration Wizard (Figure 65).

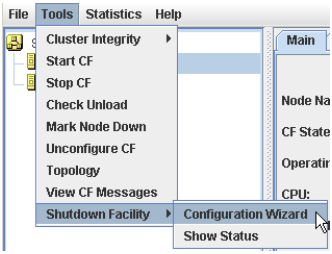

Figure 65: Starting the SF Configuration Wizard

Choose *Create* and click *Next*.

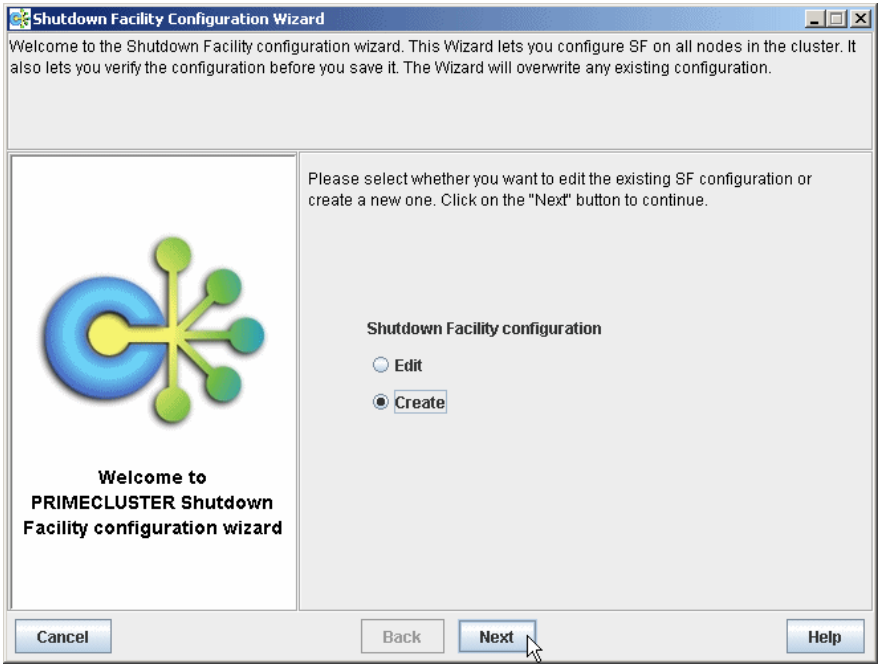

Figure 66: Creating the SF configuration

Select a configuration with the same set of SAs for all the nodes or different SAs for the individual nodes as shown in Figure 67. Click *Next*.

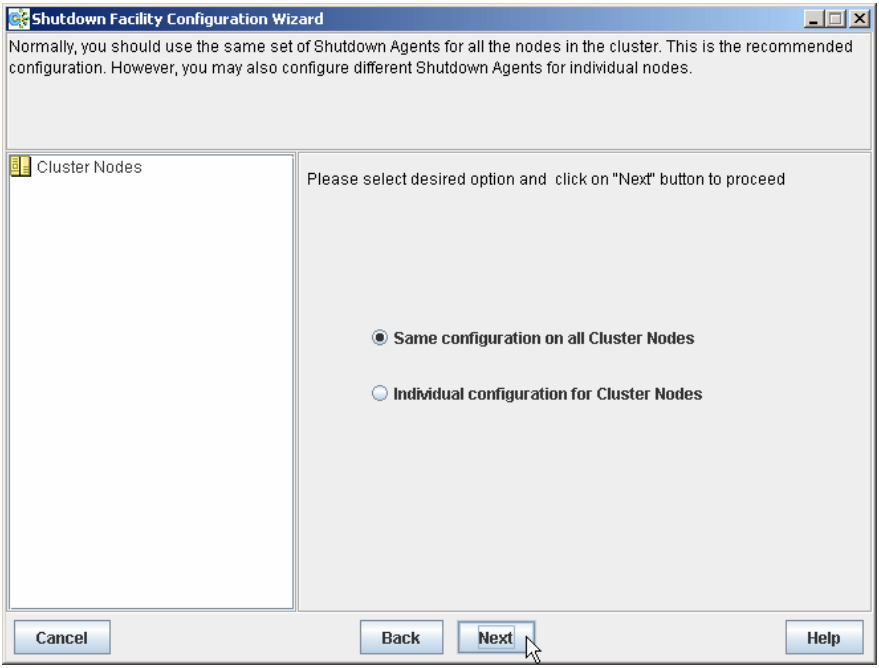

Figure 67: Choosing a common configuration for all nodes

If you choose *Same configuration on all Cluster Nodes* and click *Next*, a window such as Figure 69 appears. If you choose *Individual configuration for Cluster Nodes*, then a window such as Figure 68 appears. In this case, you can configure SF individually at a later time for each of the nodes or groups of nodes.

Currently, it is recommended that you have the same configuration on all cluster nodes.

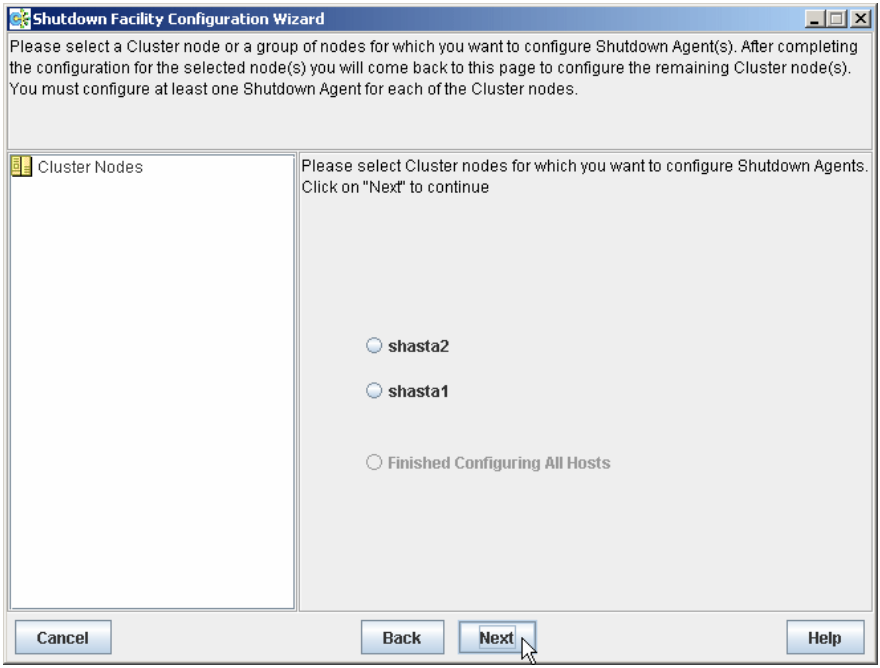

Figure 68: Selecting nodes to configure Shutdown Agents

Choose the cluster node that you want to configure and click *Next*. Note that the left panel in the window displays the cluster nodes and will progressively show the SAs configured for each node.

If you choose *Same configuration on all Cluster Nodes* in Figure 67 and clicked *Next*, a window such as Figure 69 appears.

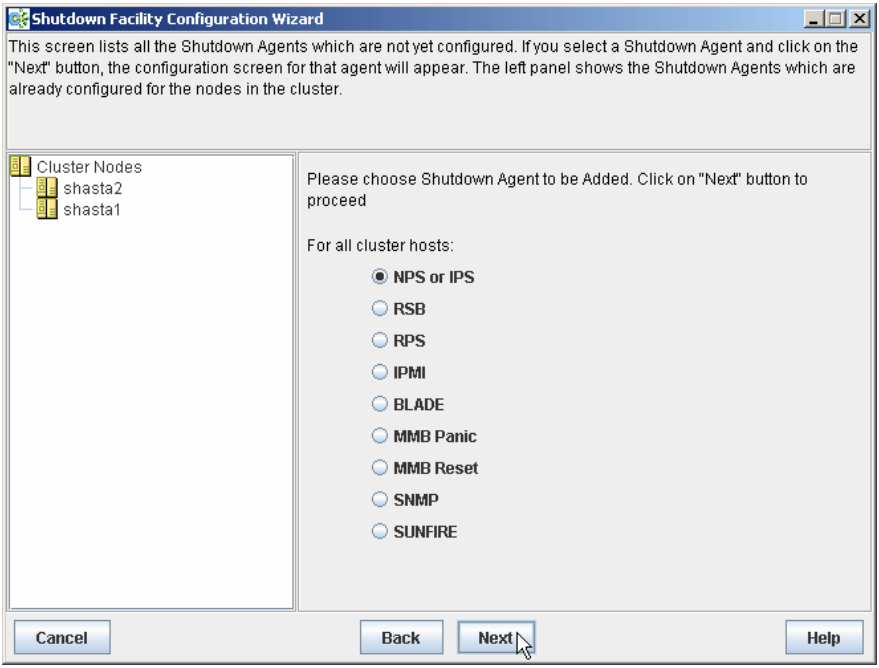

Figure 69: Choose Shutdown Agent to be added

Choose an SA from the given list and click on the *Next* button. From here you will be taken to the individual SA's configuration window, depending on your selection.

Figure 70 is the window in which to enter the *NPS or IPS Shutdown Agent* details. Enter *CF Name*, *IP Name*, *Password*, *Confirm*, and choose the *Action*. For *Action*, you can choose the value *cycle* or *leave-off*. Then click *Next*.

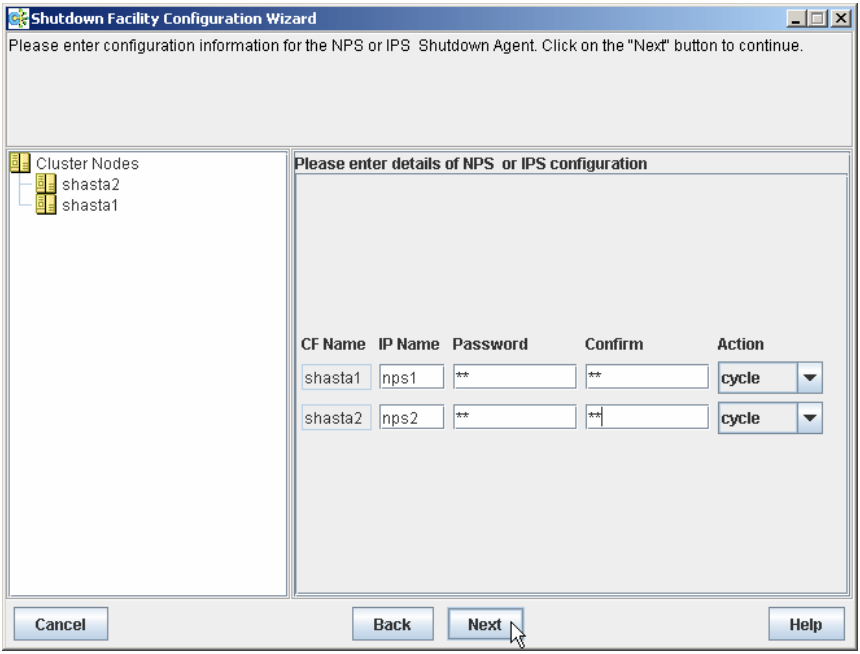

Figure 70: Configuring the IPS Shutdown Agent

The action is, by default, *cycle*, which means that the node is power cycled after shutdown.

If you choose *leave-off*, then manual action is required to turn the system back on after a IPS shutdown occurs.

Figure 71 is the window in which to enter the *SNMP Shutdown Agent* details. The top-right window is divided into two parts, global information and individual CF node information. The two settings in the global portion are as follows:

- SNMP deviceChoose RSA from the pulldown menu.
- Community StringEnter the SNMP community string in the text box. The default is *public*.

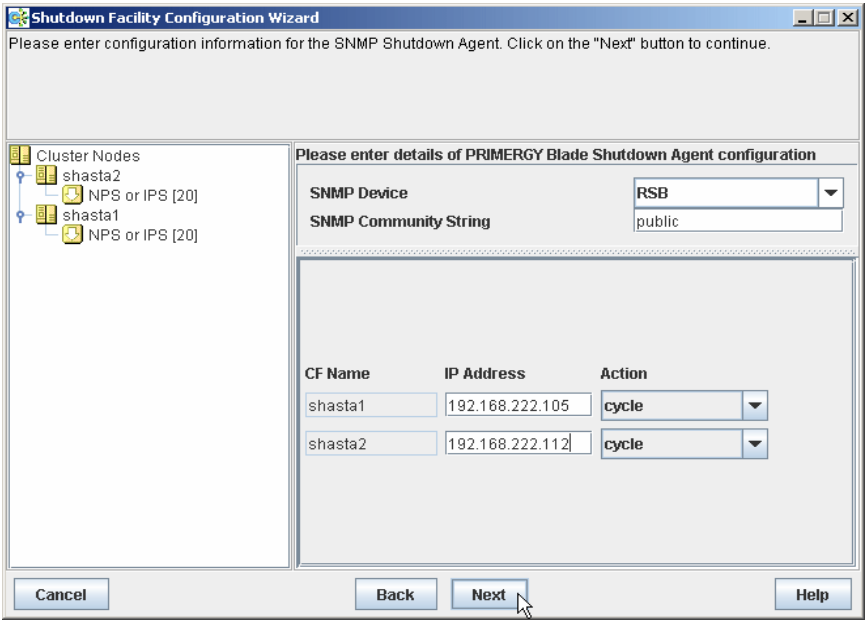

Figure 71: Selecting the SNMP Shutdown Agent options

Enter *CF Name*, *IP Address*, and choose the *Action*. For *Action*, you can choose the value *cycle* or *leave-off*. Then click *Next*.

The action is, by default, *cycle*, which means that the node is power cycled after shutdown. If you choose *leave-off*, then manual action is required to turn the system back on after a SNMP shutdown occurs.

Click the *Next* button.

If you choose IPMI, the window shown in Figure 72 appears. Enter the details for each of the cluster nodes, namely the CF name, IP of the IPMI unit, user, password, password confirmation, unit type, and the action.

For *Type*, you select either *IPMI1.5* or *IPMI2.0* (IPMI version 1.5 or 2.0). For *Action*, you can choose the value *cycle* or *leave-off*.

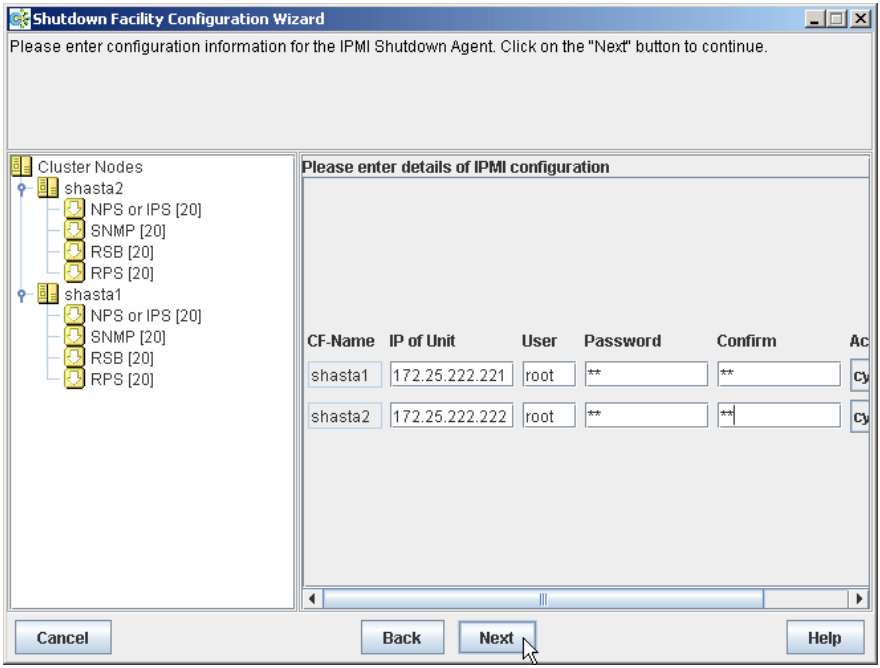

Figure 72: Configuring the IPMI Shutdown Agent

Click the *Next* button.

If you choose Blade, the window for entering the details for the Blade SA configuration appears (Figure 73).

Enter the number of Management Blade IPs and press the [Enter] or [Return] key to enter more than one Management Blade in the *Management Blade IPs* field.

If more than one Management Blade is specified, then additional *Management Blade IP* fields appear.

Specify the Management Blade IP address or addresses, the Management Blade type, and the SNMP community string for the cluster in the fields in the upper portion of the window. For each of the CF nodes, enter values for the following fields:

- CF Name—Name of the PRIMERGY Blade node
- Slot No.—Slot number of the PRIMERGY Blade node
- Action—Action can either be cycle or leave-off

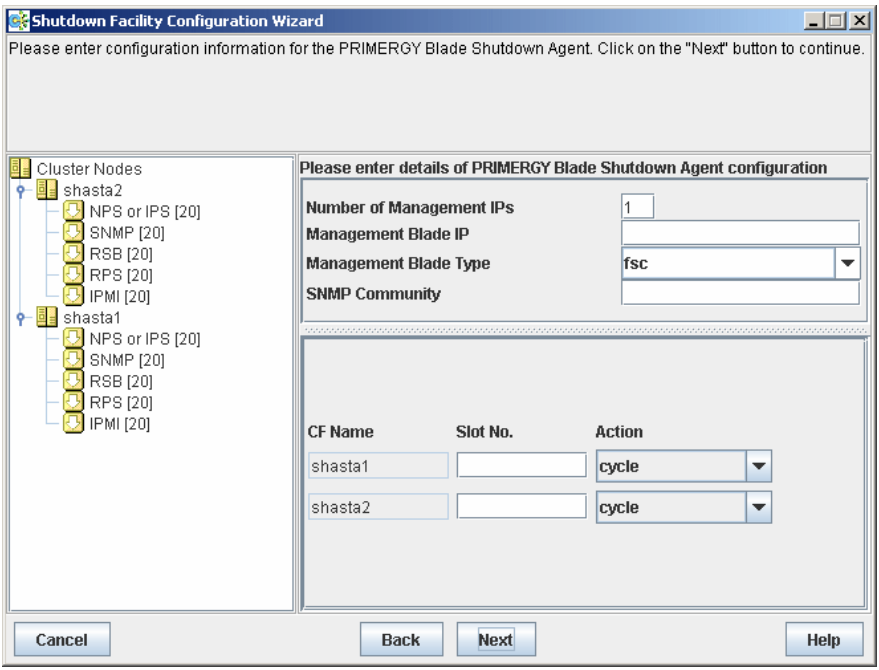

Figure 73: Configuring the Blade Shutdown Agent

Click on the *Next* button.

If you choose *MMB Panic* or *MMB Reset*, a window similar to Figure 74 appears. Enter the details for each of the cluster nodes, namely the MMB user name, password, and password confirmation.

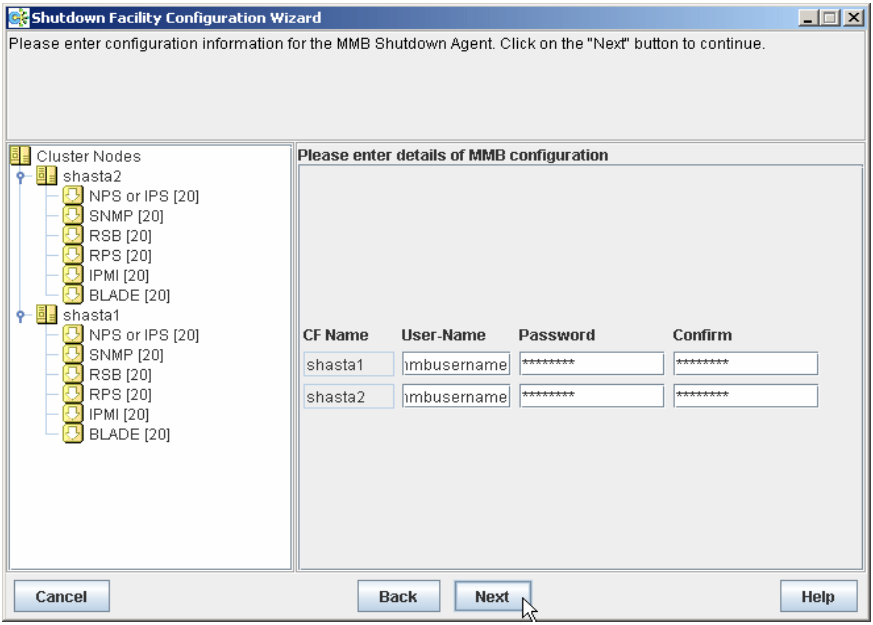

Figure 74: Configuring the MMB Shutdown Agents

You can choose both *MMB Panic* and *MMB Reset*. If you choose both, the window shown in Figure 74 does not reappear.

Click the *Next* button.

If you choose *SUNFIRE*, the window shown in Figure 75 appears. Enter the following details for each of the cluster nodes:

- *CF name*-CF name of one of the cluster nodes. This field is pre-filled and non-editable.
- *Name*-System controller host name.
- *User*-System controller login name.
- Password and password confirmation-Password for the system controller.

- *Unit Type*-Type of system controller. You can choose from the following:
  - *system-controller-15k*
  - *system-controller-6800*
  - *system-controller-alom*
  - *system-controller-alom-2k*
- *Domain ID*-Type of network domain. Choose either *Class A*, *Class B*, or *N/A*. For *system-controller-alom*, set to this value to *N/A* since *Domain ID* is not valid in an ALOM configuration.
- Domain password and password confirmation-Password for the domain.

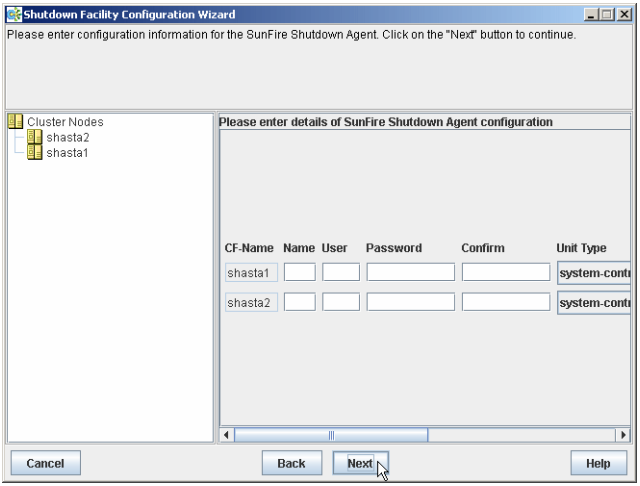

Figure 75: Configuring the SunFire Shutdown Agent

Click the *Next* button.

You can continue to *Add*, *Delete*, or *Edit* the SAs as shown in Figure 76. If there are no more SAs to add, then the *Add* option is greyed out.

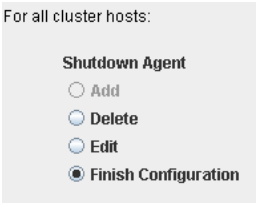

Figure 76: Adding, deleting, editing Shutdown Agents

If you have finished, select *Finish Configuration* and click on *Next* (Figure 77).

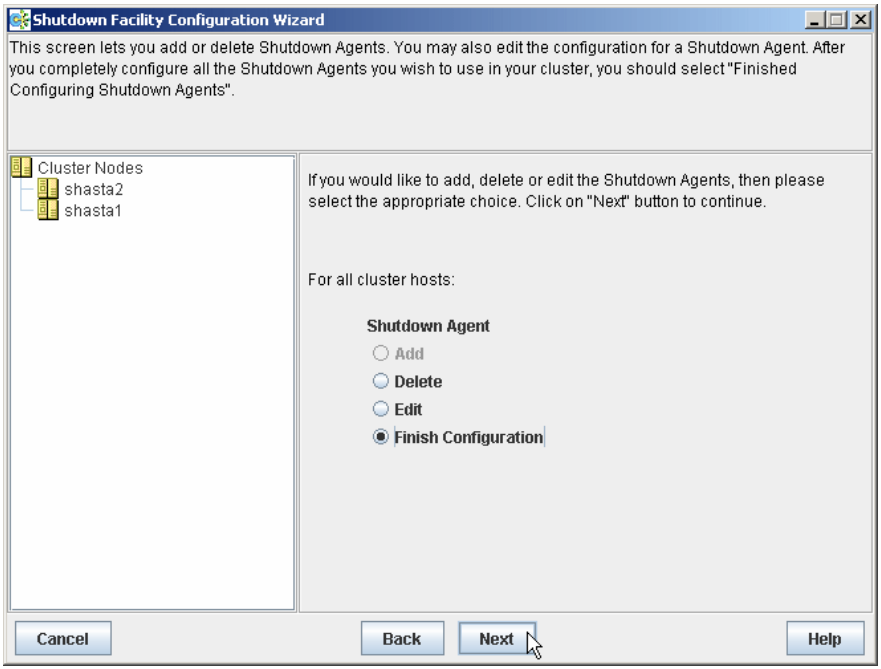

Figure 77: Finishing configuration

Next use the *UP* or *DOWN* buttons to arrange the order of the SAs (Figure 78). The SA on the top of the list is the primary SA and will be invoked first if SF needs to eliminate a node. Click on *DEFAULT* to use the recommended order for the SAs. Click on *Next*.

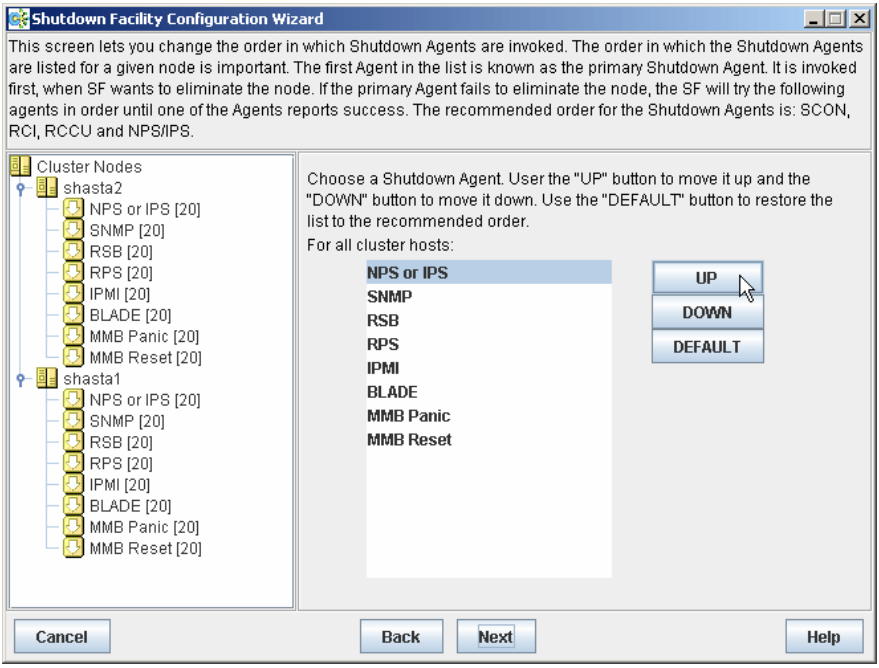

Figure 78: Changing the Shutdown Agent order

The following window lets you enter the timeout values for the configured SAs for each node (Figure 79). Enter timeout values for all nodes and for each SA or click on the *Use Defaults* button. Select *Next* to go to the next window.

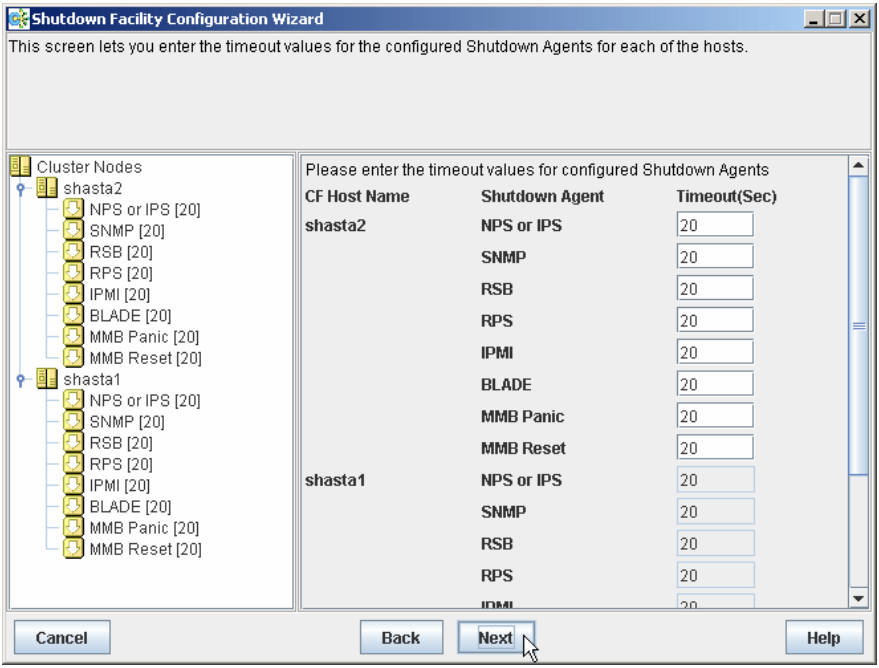

Figure 79: Specifying timeout values

The window for entering node weights and administrative IP addresses appears (Figure 80). Node weights should be an integer value greater than 0. You can select the *Admin IP* from the list of choices or enter your own. Enter node weights and *Admin IP* addresses for all CF nodes.

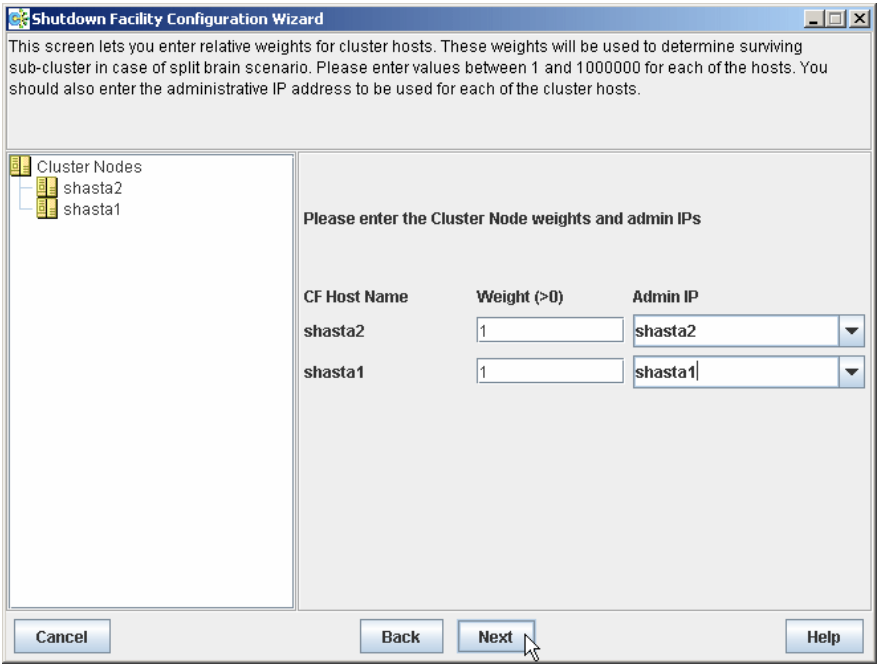

Figure 80: Entering node weights and administrative IP addresses

For our cluster we will give each node an equal node weight of 1 (refer to the Section "SF split-brain handling" for more details on node weights).

Set the *Admin IP* fields to the CF node's interface on the administrative LAN. By convention, these IP interfaces are named *node*ADM; although, this is not mandatory. If you don't have an administrative LAN, then enter the address to the public LAN. Click on *Next*.

The list of configuration files, created or edited, by the Wizard are shown in Figure 81. Click *Next* to save the configuration files or click *Back* to change the configuration.

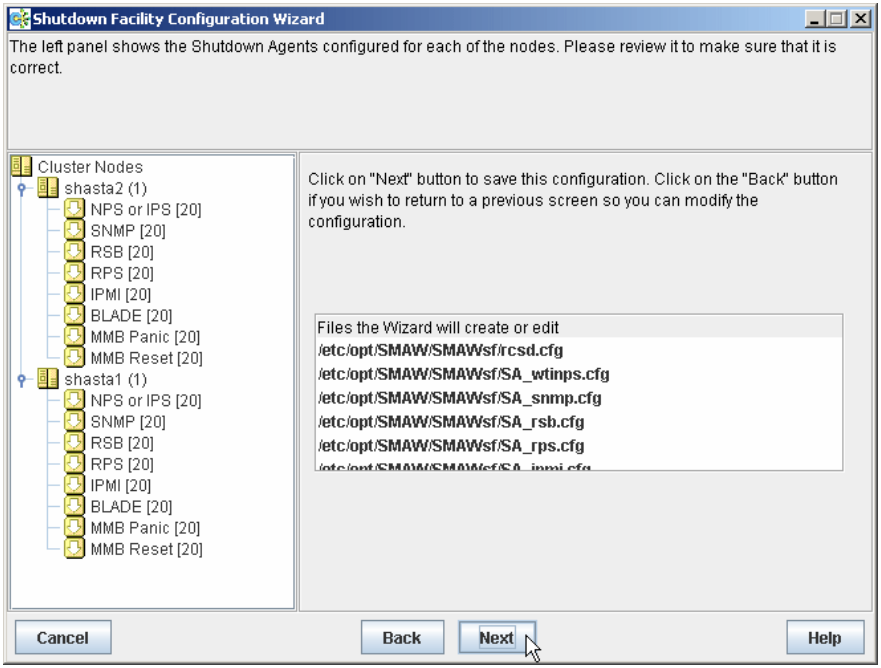

Figure 81: Confirming configuration file changes

Choose *Yes* in the confirmation popup to save the configuration (Figure 82).

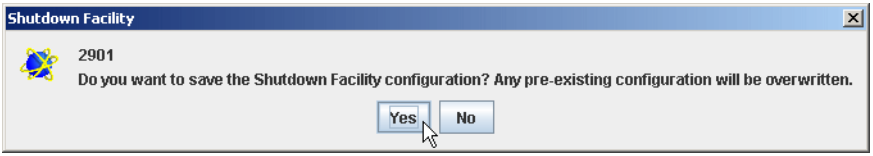

Figure 82: Saving SF configuration

The window displaying the configuration status of the shutdown agents appears (Figure 83). You can also use the *Tools* pull-down menu, and choose *Show Status* in the *Shutdown Facility* selection.

SF has a test mechanism built into it. SF periodically has each shutdown agent verify that it can shut down cluster nodes. The shutdown agent does this by going through all the steps to shut down a node, except the very last one which would actually cause the node to go down. It then reports if the test was successful. This test is run for each node that a particular agent is configured to potentially shut down.

The table in Figure 83 shows, among other things, the results of these tests. The columns *Cluster Host*, *Agent*, *SA State*, *Shut State*, *Test State*, and *Init State* when taken together in a single row, represent a test result.

If the word InitFailed appears in the InitState column, then the agent found a problem when initializing that particular shutdown agent.

If the words TestFailed appear in red in the *Test State* column, then it means that the agent found a problem when testing to see if it could shut down the node listed in the *Cluster Host* column. This indicates some sort of problem with the software, hardware, or networking resources used by that agent.

If the word Unknown appears in the *Shut State*, *Test State*, or the *Init State* columns, it means that SF has not attempted to shut down, test, or initialize those SAs. For the *Test State* and the *Init State* columns, the Unknown state is usually a temporary state that disappears when the actual state is known.

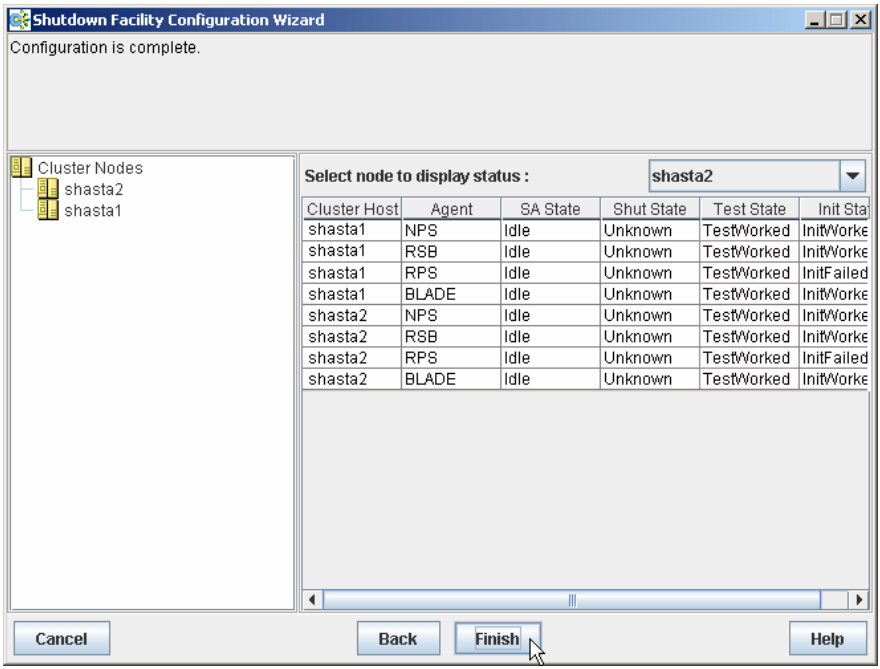

Figure 83: Status of Shutdown Agents

If you see TestFailed or InitFailed, look at the SA log file or in /var/log/messages. The log files show debugging information on why the SA's test or initialization failed. Once the problem is corrected, and SF is restarted, the status should change to InitWorked or TestWorked.

Click on the *Finish* button to exit the SF Wizard. A confirmation popup appears and asks if you really want to exit the Wizard (Figure 84). If you click on *Yes*, then the SF Wizard disappears, and you see the base Cluster Admin window.

If you click on the *Back* button in the SF Wizard instead of the *Finish* button, then you can go back and re-edit the SF configuration.

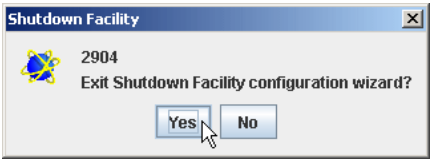

Figure 84: Exiting SF configuration wizard

8.4.2 Configuration via CLI

This section describes the setup and configuration via Command Line Interface (CLI).

Note that the format of the configuration file is presented for information purposes only. The preferred method of configuring the shutdown facility and all SAs is to use the Cluster Admin GUI (refer to the Section "Configuring the Shutdown Facility").

8.4.2.1 Shutdown Daemon

To configure the Shutdown Daemon (SD), you will need to modify the file /etc/opt/SMAW/SMAWsf/rcsd.cfg on every node in the cluster.

A file, rcsd.cfg.template, is provided under the /etc/opt/SMAW/SMAWsf directory, which is a sample configuration file for the Shutdown Daemon using fictitious nodes and agents.

It is important that the rcsd.cfg file is identical on all cluster nodes; care should be taken in administration to ensure that this is true.

An example configuration for SD (which is created by editing the sample rcsd.cfg.template) follows:

```
fuji3,weight=1,admIP=fuji3ADM:agent=SA_lkcd,timeout=20:agent=SA_blade,timeout=20
fuji2,weight=1,admIP=fuji2ADM:agent=SA_lkcd,timeout=20:agent=SA_blade,timeout=20
```

The configuration file must be created in the /etc/opt/SMAW/SMAWsf directory and must use rcsd.cfg as the file name.

The format of the configuration file is as follows:

```
cluster-node1 [,weight=w1][, admIP=admIP1]:agent=SA1,
timeout=t1[:agent=SA2, timeout=T2]...
cluster-node2 [,weight=w2][,admIP=admIP2]:agent=SA1,
timeout=t1[:agent=SA2, timeout=T2]...
...
```

- *cluster-nodeN* is the *cfname* of a node within the cluster.
- *agent* and *timeout* are reserved words.
- SA*N* is the command name of a SA.
- t*N* is the maximum time in seconds that are allowed for the associated SA to run before assuming failure.
- w*N* is the node weight.
- admIP*N* is the admin interface on the Administrative LAN on this cluster node. Available IP addresses are IPv4 and IPv6 address. IPv6 link local addresses are not available. When specifying the IPv6 address, enclose it in blankets "[ ]".
  (Example: [1080:2090:30a0:40b0:50c0:60d0:70e0:80f0])

The order of the SAs in the configuration file should be such that the first SA in the list is the preferred SA. If this preferred SA is issued a shutdown request and if its response indicates a failure to shut down, the secondary SA is issued the shutdown request. This request/response is repeated until either an SA responds with a successful shutdown, or all SAs have been tried. If no SA is able to successfully shut down a cluster node, then operator intervention is required and the node is left in the LEFTCLUSTER state.

The location of the log file will be /var/opt/SMAWsf/log/rcsd.log.

8.4.2.2 Shutdown Agents

This section contains information on how to configure the following SAs with CLI.

- IPS
- SNMP
- Blade
- IPMI
- vmSP
- ICMP
- VMCHKHOST
- libvirt
- Sun Fire

IPS

To configure IPS, you will need to create the following file:
/etc/opt/SMAW/SMAWsf/SA_wtinps.cfg

A sample configuration file can be found in the following directory:
/etc/opt/SMAW/SMAWsf/SA_wtinps.cfg.template

The configuration file SA_wtinps.cfg contains lines that are in one of two formats: a line defining an attribute and value pair or a line defining a plug set up:

- Lines defining attribute value pairs

  Attributes are similar to global variables, as they are values that are not modifiable for each IPS unit, or each cluster node. Each line contains two fields:

  Attribute-name Attribute-value

  The currently supported attribute/value pairs are as follows:

  Initial-connect-attempts *positive integer*

  This sets the number of connect retries until the first connection to an IPS unit is made. The default value for the numbers of connect retries is 12.

  wti-model-type IPS

- Lines defining a plug set up

  Each line contains four fields:

Plug-ID IP-name Password Action

The four fields are:

– *Plug-ID*: The Plug-ID of the WTI IPS unit, which should correspond to a cluster node. The *CF_name* of the cluster node must be used here.

The *Plug-ID* defined in the `SA_wtinps.cfg` file must be defined on the WTI IPS unit.

– *IP-name*: The IP name of the WTI IPS unit.

– *Password*: The password to access the WTI IPS unit.

– *Action*: The action can either be cycle or leave-off. If it is `cycle`, the node will be powered on again after powering off. If it is `leave-off`, manual action is required to turn the system back on.

The permissions of the SA_wtinps.cfg file are read/write by root only. This is to protect the password to the WTI IPS unit.

IPS is not supported in all regions. Please check with your sales representative to see if the IPS is supported in your area.

The following is an example of the SA_wtinps configuration file:

```
# Configuration for Shutdown Agent for the WTI IPS
# Each line of the file has the format:
#
#Attribute-name Attribute-value
# - or -
#Plug-ID IP-name-of-WTI-box password {cycle|leave-off}
#
# Sample:
# initial-connect-attempts 12
# fuji2wtinps1.mycompany.com wtipwd cycle
# fuji3wtinps1.mycompany.com wtipwd leave-off
# fuji4wtinps2.mycompany.com newpwd cycle
# fuji5wtinps2.mycompany.com newpwd leave-off
#
# Note:
#The Plug-ID's that are specified here must be
#configured on the named WTI IPS unit.
#
# Note:
#The permissions on the file should be read/write
```

```
#only for root.  This is to protect the password
#of the WTI IPS unit.
#
fuji2 ips6 mypassword cycle
fuji3 ips6 mypassword cycle
```

The log file is stored in the following:

/var/opt/SMAWsf/log/SA_wtinps.log

SNMP

To configure SNMP, you will need to create or modify the following file:
/etc/opt/SMAW/SMAWsf/SA_snmp.cfg

A sample configuration file can be found in the following directory:
/etc/opt/SMAW/SMAWsf/SA_snmp.cfg.template

The format of the SA_snmp.cfg file is as follows:

snmp-device <*snmp device name*>
community-string <*SNMP community string*>
cfname ip-of-snmp-device Action

snmp-device, community-string, cycle and leave-off are reserved words and must be in lowercase letters.

The editable fields are defined as follows:

- *snmp device name*: The name of the SNMP device this cluster uses.

  Known devices include rsa. If the device is other, the next nine lines must contain the description of the SNMP device. No comment lines or empty lines are allowed inside this description. The meaning of the nine items is not easy to describe, so this option is restricted to experts evaluating a new device only. Please contact field engineers in this case.

- *SNMP community string*: The SNMP community string with read/write permissions for the SNMP device

- *cfname*: The name of the node in the CF cluster.

- *ip-of-snmp-device*: The IP name or address in dot notation of the SNMP device

- *Action*: The action may either be cycle or leave-off. If it is cycle, it will be powered on again after being powered off.

For each node in the cluster, there should be a line of the following form in the configuration file: *cfname ip-of-snmp-device Action*.

The following is an example of the SA_snmp configuration file:

```
snmp-device rsa
community-string public
shasta1 123.45.56.81 cycle
shasta2 123.45.56.82 cycle
shasta3 123.45.56.83 cycle
shasta4 123.45.56.84 leave-off
```

The log file is stored in the following:

/var/opt/SMAWsf/log/SA_snmp.log

Blade

To configure the Blade shutdown agent, you can create or modify the following file:

/etc/opt/SMAW/SMAWsf/SA_blade.cfg

A sample configuration file can be found at the following location:

/etc/opt/SMAW/SMAWsf/SA_blade.cfg.template

The format of the SA_blade.cfg file is as follows:

```
community-string          SNMP community string
management-blade-type     type
management-blade-ip       ip-address
cfname                    slot-no       Action
```

Additionally, you can add the IP address of the management Blade for a second Blade chassis. In this case, you would need to add one or more lines to specify the Blades in that chassis.

management-blade-type, community-string, management-blade-ip, *cfname*, cycle and leave-off are reserved words and must be in lowercase letters.

The editable fields are defined as follows:

- *SNMP community string* is the SNMP community string with read/write permissions for the server Blades. This string is the same value as the SNMP community string in the Management Blade SNMP configuration. By default, this string is usually set to public.

- *type*—Valid options are acta-intelFW5, acta-intel, or fsc. If this line is missing, then the default of fsc is used.
- *IP-address*—The IP address of the Management Blade. Available IP addresses are IPv4 and IPv6 address. IPv6 link local addresses are not available. When specifying the IPv6 address, enclose it in blankets "[ ]". (Example: [1080:2090:30a0:40b0:50c0:60d0:70e0:80f0])
- *cfname*—The name of the node in the CF cluster.
- *slot-no*—The slot number of the Blade server.
- *Action*—The action can either be cycle or leave-off. If it is cycle, the node will be powered on again after powering off. If it is leave-off, manual action is required to turn the system back on.

For each node in the cluster, there should be a line of the following form in the configuration file:

cfname slot-no Action

The permissions of the SA_blade.cfg file are read/write by root only. This is to protect the community string used in the snmp command.

The following is an example of the SA_blade configuration file:

```
community-string public
management-blade-type atca-intel
management-blade-ip 123.45.56.78
shasta1 1 cycle
shasta2 3 leave-off
management-blade-ip 123.45.56.79
shasta3 1 cycle
shasta4 2 cycle
```

The log file is stored in the following:

/var/opt/SMAWsf/log/SA_blade.log

IPMI

To configure the IPMI SA, you need to create or modify the following file:

/etc/opt/SMAW/SMAWsf/SA_ipmi.cfg

A sample configuration file can be found at the following location:

/etc/opt/SMAW/SMAWsf/SA_ipmi.cfg.template

The SA_ipmi.cfg configuration file contains lines with four fields (and some subfields) on each line. Each line defines a node in the cluster than can be powered off (leaving it off) or powered off and then on again (power cycle). The fields are as follows:

- *cfname*—The name of the node in the CF cluster.

  cfname must correspond to the IP address of the IPMI compliant onboard LAN interface in these nodes.

- *Access-Information*—The access information is of the following format:

  ip-address-of-unit:*user*:*password*

  For *ip-address-of-unit*, specify the IP address of IPMI (BMC and iRMC). Available IP addresses are IPv4 and IPv6 address. IPv6 link local addresses are not available. When specifying the IPv6 address, enclose it in blankets "[ ]".

  (Example: [1080:2090:30a0:40b0:50c0:60d0:70e0:80f0])

- *Action*—The action can either be cycle or leave-off. If it is cycle, the node will be powered on again after powering off. If it is leave-off, manual action is required to turn the system back on.

- *IPMI version*—The version can either be ipmi2.0 or ipmi1.5. These are the supported versions of IPMI. If you are using IPMI 2, you must enter the ipmi2.0 keyword here. However, if no keyword exists, the SF defaults to IPMI1.5.

The permissions of the SA_ipmi.cfg file are read/write by root only. This is to protect the password to the BMC/iRMC unit.

The following is an example of the SA_ipmi configuration file:

```
fuji2 172.25.222.221:root:ipmipwd cycle
fuji3 172.25.222.222:root:ipmipwd cycle
fuji4 172.25.222.223:root:ipmipwd leave-off
fuji5 172.25.222.224:root:ipmipwd leave-off
```

The log file is stored in the following:

/var/opt/SMAWsf/log/SA_ipmi.log

vmSP

To configure the Shutdown Agent (SA_vmSPgp) for PANIC or Shutdown Agent (SA_vmSPgr) for RESET of Virtual Machine Service Provider (vmSP), you must create or modify the following files respectively.

/etc/opt/SMAW/SMAWsf/SA_vmSPgp.cfg

/etc/opt/SMAW/SMAWsf/SA_vmSPgr.cfg

Templates of the configuration file are stored in the following:

/etc/opt/SMAW/SMAWsf/SA_vmSPgp.cfg.template

/etc/opt/SMAW/SMAWsf/SA_vmSPgr.cfg.template

Each line of the configuration files, SA_vmSPgp.cfg and SA_vmSPgr.cfg, includes the following 5 fields:

- *cfname*—Node name on the CF cluster.
- *domain*—Domain name of a guest OS.
- *ip-address*—IP address of the Host OS. Available IP address is IPv4 address.
- *user*—Account of the Host OS. FJSVvmSP is specified with "fixed."
- *password*—Login password for the account FJSVvmSP of the Host OS. A password encrypted by the encryption command, sfcipher(8). For details on sfcipher(8), see "Manual pages."

The following is an example of the SA_vmSPgp configuration file:

```
fuji2 domain2 192.168.200.1 FJSVvmSP 3CA1wxVXKD8a93077BaEkA==
fuji3 domain3 192.168.200.2 FJSVvmSP 3CA1wxVXKD8a93077BaEkA==
```

The following is an example of the SA_vmSPgr configuration file:

```
fuji2 domain2 192.168.200.1 FJSVvmSP 3CA1wxVXKD8a93077BaEkA==
fuji3 domain3 192.168.200.2 FJSVvmSP 3CA1wxVXKD8a93077BaEkA==
```

The log file is stored in the following:

/var/opt/SMAWsf/log/SA_vmSPgp.log

/var/opt/SMAWsf/log/SA_vmSPgr.log

ICMP

To configure the ICMP SA (SA_icmp), you need to create or modify the following file:

/etc/opt/SMAW/SMAWsf/SA_icmp.cfg

The format of the SA_icmp.cfg file is as follows:

TIME_OUT=*value*
cfname:ip-address-of-node:NIC-name1,NIC-name2

The editable fields are defined as follows:

- *value*—Specify the time in seconds to check for the existence of a node. The recommended value is 5 seconds.
- *cfname*—Specify a CF node name.
- *ip-address-of-node*—Specify any of the following IP addresses of *cfname*. Available IP addresses are IPv4 and IPv6 address. IPv6 link local addresses are not available. When specifying the IPv6 address, enclose it in blankets "[ ]".
  (Example: [1080:2090:30a0:40b0:50c0:60d0:70e0:80f0])
  – Administrative LAN
  – Public LAN
  – Cluster interconnect

  You must describe one or more IP addresses for all nodes in the cluster. Add a new line to specify more than one LAN routes. It is recommended that you specify more than one routes to check the LAN routes.
- *NIC-nameX*—For *ip-address-of-node*, specify a network interface used to check the existence of the node. If there are more than one network interfaces, separate them with a comma.

 Note the following when you describe network interfaces:

- When duplicating the network by GLS, you need to describe all redundant network interfaces. (For example, eth0,eth1)
- When bonding NICs, you need to describe a bonding device after the IP address. (For example, bond0)
- When describing cluster interconnects, you need to describe all network interfaces used in all paths of the cluster interconnects. (For example, eth2,eth3)

The following is an example of the SA_icmp configuration file:

```
TIME_OUT=5
node1:10.20.30.100:eth0,eth1
node1:10.20.40.200:eth2
```

node2:10.20.30.101:eth0,eth1

node2:10.20.40.201:eth2

The log file is stored in the following:

/opt/SMAW/SMAsf/log/SA_icmp.log

VMCHKHOST

To configure the Shutdown Agent (SA_vmchkhost) to switch the HOST OS in the event of an error in the virtual machine environment, you must create or modify the following file:

/etc/opt/SMAW/SMAWsf/SA_vmchkhost.cfg

Templates of the configuration file are stored in the following:

/etc/opt/SMAW/SMAWsf/SA_vmchkhost.cfg.template

Each line of the configuration file SA_vmchkhost includes the following 5 fields:

- *guest-cfname*—CF node name of a guest OS.
- *host-cfname*—CF node name of the Host OS.
- *ip-address*—IP address of the Host OS. Available IP addresses are IPv4 and IPv6 address. IPv6 link local addresses are not available.
- *user*—Account of the Host OS. FJSVvmSP is specified with "fixed."
- *password*—Login password for the account FJSVvmSP of the Host OS. A password encrypted by the encryption command, sfcipher(8). For details on sfcipher(8), see "Manual pages."

The following is an example of the SA_vmchkhost configuration file:

fuji2 hostos2 192.168.200.1 FJSVvmSP 3CA1wxVXKD8a93077BaEkA==

fuji3 hostos3 192.168.200.2 FJSVvmSP 3CA1wxVXKD8a93077BaEkA==

The log file is stored in the following:

/var/opt/SMAWsf/log/SA_vmchkhost.log

libvirt

To configure the Shutdown Agent (SA_libvirtgp) for PANIC or Shutdown Agent (SA_libvirtgr) for RESET of libvirt, you must create or modify the following files respectively.

/etc/opt/SMAW/SMAWsf/SA_libvirtgp.cfg

/etc/opt/SMAW/SMAWsf/SA_libvirtgr.cfg

Templates of the configuration file are stored in the following:

/etc/opt/SMAW/SMAWsf/SA_libvirtgp.cfg.template

/etc/opt/SMAW/SMAWsf/SA_libvirtgr.cfg.template

Each line of the configuration files, SA_libvirtgp.cfg and SA_libvirtgr.cfg, includes the following 5 fields:

- *cfname*—Node name on the CF cluster.
- *domain*—Domain name of a guest OS.
- *ip-address*—IP address of the hypervisor. Available IP addresses are IPv4 and IPv6 address. IPv6 link local addresses are not available.
- *user*—Account of the hypervisor. The user for shutdown facility is specified.
- *password*—Login password for the account specified in "user" field. A password encrypted by the encryption command, sfcipher(8). For details on sfcipher(8), see "Manual pages."

The following is an example of the SA_libvirtgp configuration file:

fuji2 domain2 192.168.200.1 user 3CA1wxVXKD8a93077BaEkA==

fuji3 domain3 192.168.200.2 user 3CA1wxVXKD8a93077BaEkA==

The following is an example of the SA_libvirtgr configuration file:

fuji2 domain2 192.168.200.1 user 3CA1wxVXKD8a93077BaEkA==

fuji3 domain3 192.168.200.2 user 3CA1wxVXKD8a93077BaEkA==

The log file is stored in the following:

/var/opt/SMAWsf/log/SA_libvirtgp.log

/var/opt/SMAWsf/log/SA_libvirtgr.log

Sun Fire

To configure the Sun Fire SA, you will need to create the following file:

/etc/opt/SMAW/SMAWsf/SA_sunF.cfg

A sample configuration file can be found at the following location:

/etc/opt/SMAW/SMAWsf/SA_sunF.cfg.template

The SA_sunF.cfg configuration file contains two lines for each CF node. The fields are:

- *ScTag*—Type of system controller unit. The following options are available:
  - system-controller-15k

  - system-controller-6800
  - system-controller-alom
- *ScHost*—System controller host name.
- *ScLogin*—System controller login name.
- *PWord*—Password for the system controller. If *PWord* is not configured for system controller, use the value `void`.
- *PWord*—Password for the domain. If *PWord* is not configured for the domain, use the value `void`
- *DomainID*—Type of network domain. The following options are available:
  - B—Class B network
  - C—Class C network
  - void—Use this option for ALOM configurations.
- *cfname*—Name of the node in the CF cluster.

The following is an example of the SA_sunF configuration file:

```
system-controller-15k name1 user1 pass1
pass2 void shasta1
system-controller-15k   void
void void shasta2
```

The log file is stored in the following:

/var/opt/SMAWsf/log/SA_sunF.log

8.5 SF administration

This section provides information on administering SF. SF can be administered with the CLI or Cluster Admin. It is recommended to use Cluster Admin.

8.5.1 Starting and stopping SF

This section describes the following administrative procedures for starting and stopping SF:

- Manually via the CLI
- Automatically via the rc script interface

8.5.1.1 Starting and stopping SF manually

SF may be manually started or stopped by using the sdtool(1M) command. The sdtool(1M) command. Refer to the Chapter "Manual pages" for more information on CLI commands.

8.5.1.2 Starting and stopping SF automatically

SF can be started automatically using the S64sf RC-script available under the /etc/rc2.d directory. The rc start/stop script for SF is installed as /etc/init.d/RC_sf.

8.6 Logging

Whenever there is a recurring problem where the cause cannot be easily detected, turn on the debugger with the following command:

 # sdtool -d on

This will write the debugging information into the log file: /var/opt/SMAWsf/log/rscd.log, which will provide additional information to find the cause of the problem. You can also use the sdtool -d off command to turn off debugging.

Note that the rcsd log file does not contain logging information from any SA. Refer to the SA specific log files for logging information from a specific SA.

9 Diagnostics and troubleshooting

This chapter provides help for troubleshooting and problem resolution for PRIMECLUSTER Cluster Foundation. This chapter will help identify the causes of problems and possible solutions. If a problem is in another component of the PRIMECLUSTER suite, the reader will be referred to the appropriate manual. This chapter assumes that the installation and verification of the cluster have been completed as described in the *Software Release Guide PRIMECLUSTER* and the *PRIMECLUSTER Installation Guide*.

This chapter discusses the following:

- The Section "Beginning the process" discusses collecting information used in the troubleshooting process.
- The Section "Symptoms and solutions" is a list of common symptoms and the solutions to the problems.
- The Section "Collecting Troubleshooting Information" gives steps and procedures for collecting troubleshooting information.

9.1 Beginning the process

Start the troubleshooting process by gathering information to help identify the causes of problems. You can use the CF log viewer facility from the Cluster Admin GUI, look for messages on the console, or look for messages in the /var/log/messages file. You can use the cftool(1M) command for checking states, configuration information. To use the CF log viewer click on the *Tools* pull-down menu and select *View Syslog messages* (refer to the Section "Using PRIMECLUSTER log viewer" for more details). The log messages are displayed. You can search the logs using a date/time filter or scan for messages based on severity levels. To search based on date/time, use the date/time filter and press the *Filter* button. To search based on severity levels, click on the *Severity* button and select the desired severity level. You can use keyword also to search the log. To detach the CF log viewer window, click on the *Detach* button; click on the *Attach* button to attach it again.

Collect information as follows:

- Look for messages on the console that contain the identifier CF.
- Look for messages in /var/log/messages. You might have to look in multiple files (/var/log/messages.*N*).

- Use cftool as follows:
  - cftool -l: Check local node state
  - cftool -d: Check device configuration
  - cftool -n: Check cluster node states
  - cftool -r: Check the route status

Error log messages from CF are always placed in the /var/log/messages file; some messages may be replicated on the console. Other device drivers and system software may only print errors on the console. To have a complete understanding of the errors on a system, both console and error log messages should be examined. The Section "Alphabetical list of messages" contains messages that can be found in the /var/log/messages file. This list of messages gives a description of the cause of the error. This information is a good starting point for further diagnosis.

All of the parts of the system put error messages in this file or on the console and it is important to look at all of the messages, not just those from the PRIME-CLUSTER suite. The following is an example of a CF error message from the /var/log/messages file:

Aug 26 13:31:05 fuji2 kernel: LOG3.0429320 1080024 100014 0 1.0 CF: Giving UP Mastering (Cluster already Running)

The parts of this message are as follows:

The first 80 bytes are the log3 prefix:

Aug 26 13:31:05 fuji2 kernel: LOG3. .0429320 1080024 100014 0 1.0 cf:elmlog

This parts of the message is a standard prefix on each CF message in the log file that gives the date and time, the node name, and log3 specific information. Only the date, time, and node name are important in this context. The remainder is the error message from CF as follows:

CF: Giving UP Mastering (Cluster already Running).

When the node detects a joined server, and it enters an existing cluster instead of making a new cluster, this message is output. Refer to Section "Alphabetical list of messages" for details of the message.

Several options for the command cftool(1M) are available as sources for information. The following is an example:

root@fuji2> **cftool -l**

```
Node    Number  State     Os      Cpu      Flags
fuji2      2      UP     Linux   Pentium    0
```

This shows that the local node has joined a cluster as node number 2 and is currently UP. This is the normal state when the cluster is operational. Another possible response is as follows:

root@fuji2> **cftool -l**

```
Node    Number      State        Os  Cpu   Flags
fuji2     --     COMINGUP       --   --
```

This indicates that the CF driver is loaded and that the node is attempting to join a cluster. If the node stays in this state for more than a few minutes, then something is wrong and we need to examine the /var/log/messages file. In this case, we see as follows:

root@fuji2> **tail /var/log/messages**

```
Aug 28 10:38:25 fuji2  kernel:  CF: (TRACE): Load: Complete.
Aug 28 10:38:25 fuji2  kernel:  CF: (TRACE): JoinServer: Startup.
Aug 28 10:38:25 fuji2  kernel:  CF: Giving UP Mastering (Cluster already Running).
Aug 28 10:38:25 fuji2 kernel:   CF: fuji2: busy: local node not DOWN:
retrying.
```

We see that this node is in the LEFTCLUSTER state on another node (fuji4). To resolve this condition, see Chapter "LEFTCLUSTER state" for a description of the LEFTCLUSTER state and the instructions for resolving the state.

The next option to cftool(1M) shows the device states as follows:

root@fuji2> **cftool -d**

```
Number Device Type Speed    Mtu  State Configured     Address
  1      eth0    4    100   1432  UP      YES      00.03.47.c2.a8.82
  2      eth1    4    100   1432  UP      YES      00.02.b3.88.09.f1
  3      eth2    4    100   1432  UP       NO      00.02.b3.88.09.ea
```

Here we can see that there are two interconnects configured for the cluster (the lines with YES in the Configured column). This information shows the names of the devices and the device numbers for use in further troubleshooting steps.

The cftool -n command displays the states of all the nodes in the cluster. The node must be a member of a cluster and UP in the cftool -l output before this command will succeed:

root@fuji2> **cftool -n**

```
Node    Number State     Os      Cpu
fuji2        1    UP     Linux   Pentium
fuji3        2    UP     Linux   Pentium
```

This indicates that the cluster consists of two nodes fuji2 and fuji3, both of which are UP. If the node has not joined a cluster, the command will wait until the join succeeds.

cftool -r lists the routes and the current status of the routes as follows:

root@fuji2> **cftool -r**

```
Node    Number Srcdev Dstdev Type State    Destaddr
fuji2        1      1      4    4    UP   00.03.47.c2.a8.82
fuji2        1      1      5    5    UP   00.03.47.c2.a8.cc
fuji3        2      2      4    4    UP   00.03.47.d1.af.ec
fuji3        2      2      5    5    UP   00.03.47.d1.af.ef
```

This shows that all of the routes are UP. If a route shows a DOWN state, then the step above where we examined the error log should have found an error message associated with the device. At least the CF error noting the route is down should occur in the error log. If there is not an associated error from the device driver, then the diagnosis steps are covered below.

The last route to a node is never marked DOWN, it stays in the UP state so that the software can continue to try to access the node. If a node has left the cluster or gone down, there will still be an entry for the node in the route table and one of the routes will still show as UP. Only the cftool -n output shows the state of the nodes. The following example shows:

root@fuji2> **cftool -r**

```
Node    Number Srcdev Dstdev Type State Destaddr
fuji3        2      3      2    4    UP  00.03.47.d1.af.ec
fuji2        1      3      3    4    UP  00.03.47.c2.a8.82
```

root@fuji2> **cftool -n**

```
Node    Number State              Os      Cpu
fuji3        1   LEFTCLUSTER Linux   Pentium
fuji2        2   UP                  Linux   Pentium
```

9.2 Symptoms and solutions

The previous section discussed the collection of data. This section discusses symptoms and gives guidance for troubleshooting and resolving the problems. The problems dealt with in this section are divided into two categories: problems with joining a cluster and problems with routes, either partial or complete loss of routes. The solutions given here are either to correct configuration problems or to correct interconnect problems. Problems outside of these categories or solutions to problems outside of this range of solutions are beyond the scope of this manual and are either covered in another product's manual or require technical support from field engineers. Samples from the error log (/var/log/messages) have the log3 header stripped from them in this section.

9.2.1 Join-related problems

Join problems occur when a node is attempting to become a part of a cluster. The problems covered here are for a node that has previously successfully joined a cluster. If this is the first time that a node is joining a cluster, the *Software Release Guide PRIMECLUSTER* and the *PRIMECLUSTER Installation Guide* section on verification covers the issues of initial startup. If this node has previously been a part of the cluster and is now failing to rejoin the cluster, here are some initial steps in identifying the problem.

9.2.1.1 Identifying join-related problems

First, look in the error log and at the console messages for any clue to the problem. Have the Ethernet drivers reported any errors? Any other unusual errors? If there are errors in other parts of the system, the first step is to correct those errors. Once the other errors are corrected, or if there were no errors in other parts of the system, proceed as follows.

Is the CF device driver loaded? The device driver puts a message in the log file when it loads and the cftool -l command will indicate the state of the driver. The logfile message looks as follows:

CF: (TRACE): JoinServer: Startup.

cftool -l prints the state of the node as in the following:

root@fuji2> **cftool -l**

```
Node    Number State          Os
fuji2       --   COMINGUP     --
```

This indicates that the driver is loaded and that the node is trying to join a cluster. If the errorlog message above does not appear in the logfile or the cftool -l command fails, then the device driver is not loading. If there is no indication in the /var/log/messages file or on the console why the CF device driver is not loading, it could be that the CF kernel binaries or commands are corrupted, and you might need uninstall and reinstall CF. Before any further steps can be taken, the device driver must be loaded.

After the CF device driver is loaded, it attempts to join a cluster as indicated by the following message:

CF: (TRACE): JoinServer: Startup

The join server will attempt to contact another node on the configured interconnects. If one or more other nodes have already started a cluster, this node will attempt to join that cluster. The following message in the error log indicates that this has occurred:

CF: Giving UP Mastering (Cluster already Running).

If this message does not appear in the error log, then the node did not see any other node communicating on the configured interconnects and it will start a cluster of its own. The following two messages will indicate that a node has formed its own cluster as follows:

CF: Local Node fuji2 Created Cluster FUJI. (#0000 1)
CF: Node fuji2 Joined Cluster FUJI. (#0000 1)

At this point, we have verified that the CF device driver is loading and the node is attempting to join a cluster. In the following list, problems are described with corrective actions. Find the problem description that most closely matches the symptoms of the node being investigated and follow the steps outlined there.

9.2.1.2 Solving join-related problems

Problem

The following are typical join problems.

The node does not join an existing cluster; it forms a cluster of its own.

Diagnosis

The error log shows the following messages:

```
CF: (TRACE): JoinServer: Startup.
CF: Local Node fuji2 Created Cluster FUJI. (#0000 1)
CF: Node fuji2 Joined Cluster FUJI. (#0000 1)
```

This indicates that the CF devices are all operating normally and suggests that the problem is occurring some place in the interconnect. The first step is to determine if the node can see the other nodes in the cluster over the interconnect. Use cftool(1M) to send an echo request to all the nodes of the cluster:

```
root@fuji2> cftool -e
```

Localdev	Srcdev	Address	Cluster	Node	Number	Joinstate
3	2	00.03.47.c2.a8.82	FUJI	fuji2	2	6
3	3	00.03.47.d1.af.ec	FUJI	fuji3	1	6

This shows that node fuji3 sees node fuji2 using interconnect device 3 (Localdev) on fuji3 and device 2 (Srcdev) on fuji2. If the cftool -e shows only the node itself continue on in this section." If some or all of the expected cluster nodes appear in the list, attempt to rejoin the cluster by unloading the CF driver and then reloading the driver as follows:

```
root@fuji2> cfconfig -u
root@fuji2> cfconfig -l
```

There is no output from either of these commands, only error messages in the error log.

Problem

The node does not join the cluster and some or all nodes respond to cftool -e.

Diagnosis

At this point, we know that the CF device is loading properly and that this node can communicate with at least one other node in the cluster. We should suspect at this point that the interconnect is missing messages. One way to test this hypothesis is to repeatedly send echo requests and see if the result changes over time, for example:

root@fuji2> **cftool -e**

Localdev	Srcdev	Address	Cluster	Node	Number	Joinstate
3	2	00.03.47.c2.aa.f9	FUJI	fuji2	3	6
3	2	00.03.47.c2.a8.82	FUJI	fuji3	2	6
3	3	00.03.47.d1.af.ec	FUJI	fuji4	1	6

root@fuji2> **cftool -e**

Localdev	Srcdev	Address	Cluster	Node	Number	Joinstate
3	2	00.03.47.c2.aa.f9	FUJI	fuji2	3	6
3	2	00.03.47.c2.a8.82	FUJI	fuji3	2	6
3	3	00.03.47.d1.af.ec	FUJI	fuji4	1	6
3	3	00.03.47.d1.ae.f9	FUJI	fuji5	1	6

root@fuji2> **cftool -e**

Localdev	Srcdev	Address	Cluster	Node	Number	Joinstate
3	2	00.03.47.c2.aa.f9	FUJI	fuji2	3	6
3	2	00.03.47.c2.a8.82	FUJI	fuji3	2	6
3	3	00.03.47.d1.af.ec	FUJI	fuji4	1	6

root@fuji2> **cftool -e**

Localdev	Srcdev	Address	Cluster	Node	Number	Joinstate
3	2	00.03.47.c2.aa.f9	FUJI	fuji2	3	6
3	2	00.03.47.c2.a8.82	FUJI	fuji3	2	6
3	3	00.03.47.d1.af.ec	FUJI	fuji4	1	6
3	3	00.03.47.d1.ae.f9	FUJI	fuji5	1	6

root@fuji2> **cftool -e**

Localdev	Srcdev	Address	Cluster	Node	Number	Joinstate
3	2	00.03.47.c2.aa.f9	FUJI	fuji2	3	6
3	2	00.03.47.c2.a8.82	FUJI	fuji3	2	6
3	3	00.03.47.d1.af.ec	FUJI	fuji4	1	6
3	3	00.03.47.d1.ae.f9	FUJI	fuji5	1	6

root@fuji2> **cftool -e**

Localdev	Srcdev	Address	Cluster	Node	Number	Joinstate
3	2	00.03.47.c2.aa.f9	FUJI	fuji2	3	6
3	2	00.03.47.c2.a8.82	FUJI	fuji3	2	6
3	3	00.03.47.d1.af.ec	FUJI	fuji4	1	6
3	3	00.03.47.d1.ae.f9	FUJI	fuji5	1	6

Notice that the node fuji5 does not show up in each of the echo requests. This indicates that the connection to the node fuji5 is having errors. Because only this node is exhibiting the symptoms, we focus on that node. First, we need to examine the node to see if the Ethernet utilities on that node show any errors. We log on to fuji5 and use the netstat(8) utility to find out the network interface information and errors.

The netstat(8) utility in Linux reports information about the network interfaces.

Further resolution of the problem consists of trying each of the following steps:

- Ensure that the Ethernet cable is securely inserted at each end.
- Try repeated cftool -e and look at the netstat -i. If the results of the cftool(1M) are always the same and the input errors are gone or greatly reduced, the problem is solved.
- Replace the Ethernet cable.
- Try a different port in the Ethernet hub or switch or replace the hub or switch, or temporarily use a cross-connect cable.
- Replace the Ethernet adapter in the node.

If none of these steps resolves the problem, then field engineers will have to further diagnose the problem.

Problem

The following console message appears on node fuji3 while node fuji2 is trying to join the cluster with node fuji3:

Aug 30 21:31:35 fuji3 kernel: CF: Local node is missing a route from node: fuji2.
Aug 30 21:31:35 fuji3 kernel: CF: missing route on local device: eth1.
Aug 30 21:31:35 fuji3 kernel: CF: Node fuji2 Joined Cluster FUJI. (#0000 3)

Diagnosis

Look in /var/log/messages on node fuji2.
Same message as on console.

No console messages on node fuji3.

Look in /var/log/messages on node fuji3.

fuji3:**cftool -d**

Number	Device	Type	Speed	Mtu	State	Configured	Address
1	eth0	4	100	1432	UP	YES	00.03.47.c2.a8.82
2	eth1	4	100	1432	UP	YES	00.02.b3.88.09.f1
3	eth2	4	100	1432	UP	NO	00.02.b3.88.09.ea

fuji2:**cftool -d**

Number	Device	Type	Speed	Mtu	State	Configured	Address
1	eth0	4	100	1432	UP	YES	00.03.47.c2.a8.3c
2	eth1	4	100	1432	UP	NO	00.02.b3.88.b8.89
3	eth2	4	100	1432	UP	NO	00.02.b3.88.b7.46

Problem

eth1 is not configured are on node fuji2:

Diagnosis

Look in /var/log/messages on node fuji3

Aug 27 16:05:59 fuji3 kernel: e100: eth1 NIC Link is Down
Aug 27 16:06:08 fuji3 kernel: CF: Icf Error: (service err_type route_src route_dst). (#0000 0 2 1 1)
Aug 27 16:06:08 fuji3 kernel: CF: (TRACE): CFSF failure detected: no SFopen: passed to ENS: fuji2. (#0000 1)
Aug 27 16:06:08 fuji3 kernel: CF: Node fuji2 Left Cluster FUJI. (#00001)

Problem

The eth1 device or interconnect temporarily failed. It could be the NIC on either of the cluster nodes or a cable or the following hub problem.

Node in LEFTCLUSTER state

Node fuji2 panicked and has rebooted. The following console message appears on node fuji2:

Aug 28 10:38:25 fuji2 kernel: CF: fuji2: busy: local node not DOWN: retrying

Diagnosis

Look in /var/log/messages on node fuji2

```
Aug 28 10:38:   fuji2  kernel: CF: (TRACE): JoinServer: Startup.
Aug 28 10:38:25 fuji2  kernel: CF: Giving UP Mastering (Cluster already Running).
Aug 28 10:38:25 fuji2  kernel: CF: fuji3: busy: local node not DOWN: retrying
```

… last message repeats

No new messages on console or in /var/log/messages on fuji3

fuji3:**cftool -n**

```
Node    Number   State                Os       Cpu
fuji2        1       LEFTCLUSTER Linux   Pentium
fuji3        2       UP                   Linux   Pentium
```

Problem

Node fuji2 has left the cluster and has not been declared DOWN.

Fix

cftool -k

This option will declare a node down. Declaring an operational node down can result in catastrophic consequences, including loss of data in the worst case. If you do not wish to declare a node down, quit this program now.

```
Enter node number: 1
Enter name for node #1: fuji2
cftool(down): declaring node #1 (fuji2) down
cftool(down): node fuji2 is down
```

The following console messages then appear on node fuji3:

```
Aug 28 10:47:39 fuji5 kernel: CF: FUJI: fuji2  is Down. (#0000 2)
Aug 28 10:49:09 fuji5 kernel: CF: Node fuji2  Joined Cluster FUJI. (#0000 2)
```

The following console message appears on node fuji2:

```
Aug 28 10:49:00 fuji2  kernel: CF: Local Node fuji2
Created Cluster  FUJI. (#0000 1)
```

9.3 Collecting Troubleshooting Information

If an error occurs in the PRIMECLUSTER system, collect the information required for the error investigation from all nodes that construct the cluster and the cluster management servers. Then, contact field engineers.

1. PRIMECLUSTER investigation information
   - Use fjsnap to collect information.
   - Retrieve the system dump.
   - Collect the Java Console on the clients.

     Refer to the Java console documentation in the *PRIMECLUSTER Web-Based Admin View Operation Guide*.
   - Collect a hard copy of the client screens.

     Refer to the screen hard copy documentation in the *PRIMECLUSTER Web-Based Admin View Operation Guide*.
2. Investigation information for the failed application
3. Crash Dump

   If the failed node is restartable, manually collect a crash dump before restarting it. Crash dump will be useful for troubleshooting if the failure is OS related.

   Example: If the failover occurred due to an unexpected resource failure
   After the failover of the cluster application is complete, collect a crash dump on the node where the resource failure occurred.

   For details on the crash dump, see Section "Crash Dump".
4. Error reproduction procedure description if the error can be reproduced

- When reporting a problem, collect the information required for an error investigation. If you do not provide information for problem checking and error reproduction execution, it may take a long time to reproduce and diagnose the problem or it may become impossible to do so.
- Collect investigation material promptly from all nodes of the PRIMECLUSTER system. Necessary information may become lost if a long time elapses after the error occurs. This applies especially to information collected by fjsnap.

9.3.1 Executing the fjsnap Command

The fjsnap command is a tool with which you can collect information for troubleshooting of PRIMECLUSTER. If a failure occurs in the PRIMECLUSTER system, this tool enables you to collect required information to pursue an investigation into the cause of the problem. You can execute this command as follows:

1. Log in with system administrator authority.
2. Execute the fjsnap command.

   /opt/FJSVpclsnap/bin/fjsnap -a output

   or

   /opt/FJSVpclsnap/bin/fjsnap -h output

   – If -a is specified, the amount of data becomes large because all detailed information is collected. If -h is specified, only cluster control information is collected.

   – Specify a special file name or an output file name (ex: /dev/st0) to output information. The specified name is the file to which collected information is output when the fjsnap command is executed.

     If you a specifying a relative path from the current directory to an output file name that contains a directory, begin the path specification with "./".

For details on the fjsnap command, see the README file included in the FJSVpclsnap package.

Execution timings for the fjsnap command

- For problems that occur during operation, for example, if an error message is output, execute the fjsnap command immediately after the problem occurs.
- If the fjsnap command cannot be executed because the system hangs, collect a crash dump. Then start the system in single user mode, and execute the fjsnap command. For information on how to collect a crash dump, see Section "Crash Dump".
- After an error occurs, if a node restarts automatically (the node could not be started in single-user mode) or if the node is mistakenly started in multi-user mode, execute the fjsnap command.
- If investigation information cannot be collected because the fjsnap command results in an error or the fjsnap command does not return, then collect a system dump.

Free space required for the execution of the fjsnap command

The approximate amount of free space required for the execution of the fjsnap command is listed in the following table:

	Default directory	**Free space (approximate) (MB)**
Output directory	Current directory during the execution of the command	300
Temporary directory	/tmp	500

Caution

The listed values for the amount of free space (300 MB, 500 MB) may be insufficient depending on the system environment.
If troubleshooting information cannot be collected successfully due to there being insufficient directory space, the fjsnap command outputs an error message or a warning message upon the termination of the execution. In this case, re-execute the command according to the corrective action given below:

Corrective action when the amount of free space in the output directory is insufficient

The fjsnap command outputs the error message shown below when the creation of the output file has failed:

ERROR: failed to generate the output file "xxx".
DIAG: ...

Corrective action:

Change the output directory to one with a large amount of free space, and then re-execute the command.

Example:

When the output directory is be changed to /var/crash

 # /opt/FJSVpclsnap/bin/fjsnap -a /var/crash/output

When there is insufficient free space in the temporary directory

The fjsnap command may output the following warning message upon the termination of the command execution:

WARNING: The output file "xxx" may not contain some data files.
DIAG: ...

The output of this warning message indicates that the output file of the fjsnap command has been created. However, part of the information to be collected may not be included in the output file.

Corrective action:

Change the temporary directory to one with sufficient free space, and then re-execute the command.

Example:

When the temporary directory is to be changed to /var/crash

 # /opt/FJSVpclsnap/bin/fjsnap -a -T/var/crash output

If the same warning message continues to be output even after you change the temporary directory, the error may be caused by one of the following:

(1) A timeout occurs for a specific information collecting command due to the state of the system.
(2) The file from which information is to be collected is larger than the amount of free space in the temporary directory.

In the case of (1), the log of timeout occurrence is recorded in the fjsnap.elog file that is contained in the fjsnap output file. If possible, collect a crash dump from the fjsnap.elog file and from the fjsnap output file.

In the case of (2), confirm that the sizes of (a) and (b), below, are not larger than the amount of free space in the temporary directory:

(a) Log file size

- /var/log/messages
- Log files (SMAWsf/log/rcsd.log etc.) placed under
- /var/opt/SMAW*/log/

(b) Total size of the core files

- GFS core file
  /var/opt/FJSVsfcfs/cores/*
- GDS core file
  /var/opt/FJSVsdx/*core/*

If these are larger than the amount of free space in the temporary directory, move the relevant files to another partition that contains neither the output directory nor the temporary directory, and then re-execute the fjsnap command. Do not delete the moved files. Instead, save them.

9.3.2 Crash Dump

Check the crash dump directory for a crash dump created after the switchover had occurred. The time the dump was written can be found by referring to the time stamp using, for example, ls(1) command.

- If a crash dump after the switchover is found

  Save the crash dump.

- If a crash dump after the switchover is not found

  If the failed node is restartable, manually collect a crash dump before restarting it.

Crash dump directory

A crash dump is stored as a file on the node in which the error occurred.

10 CF over IP

This chapter describes CF over IP and how it is configured.

This chapter discusses the following:

- The Section "Overview" introduces CF over IP and describes its use.
- The Section "Configuring CF over IP" details how to configure CF over IP.

10.1 Overview

All IP configuration must be done prior to using CF over IP. The devices must be initialized with a unique IP address and a broadcast mask. IP must be configured to use these devices. If the configuration is not done, cfconfig(1M) will fail to load CF, and CF will not start.

The devices used for CF over IP must not be controlled by an RMS userApplication that could unconfigure a device due to Offline processing.

CF communications are based on the use of interconnects. An interconnect is a communications medium which can carry CF's link-level traffic between the CF nodes. A properly configured interconnect will have connections to all of the nodes in the cluster through some type of device. This is illustrated in Figure 85.

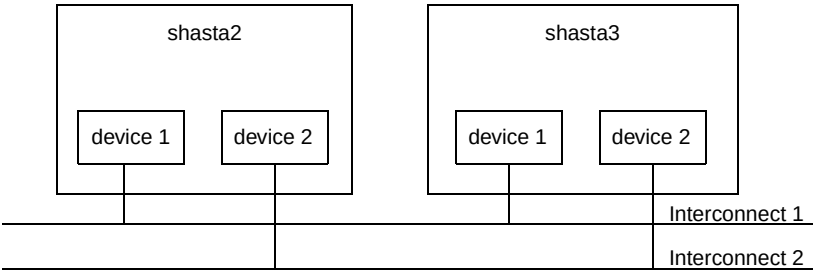

Figure 85: Conceptual view of CF interconnects

When CF is used over Ethernet, Ethernet devices are used as the interfaces to the interconnects. The interconnects themselves are typically Ethernet hubs or switches. An example of this is shown in Figure 86.

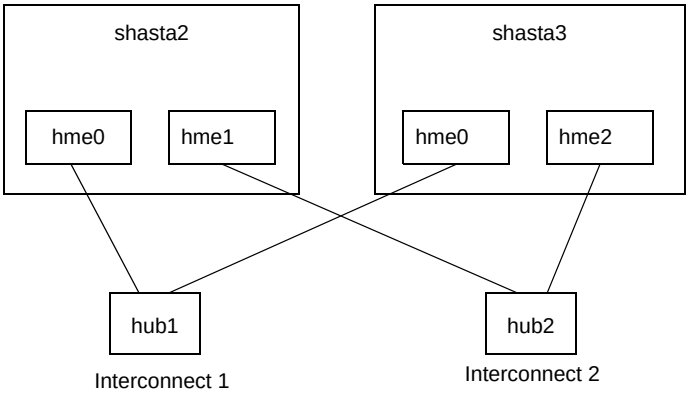

Figure 86: CF with Ethernet interconnects

When CF is run over IP, IP interfaces are the devices used to connect to the interconnect. The interconnect is an IP subnetwork. Multiple IP subnetworks may be used for the sake of redundancy. Figure 87 shows a CF over IP configuration.

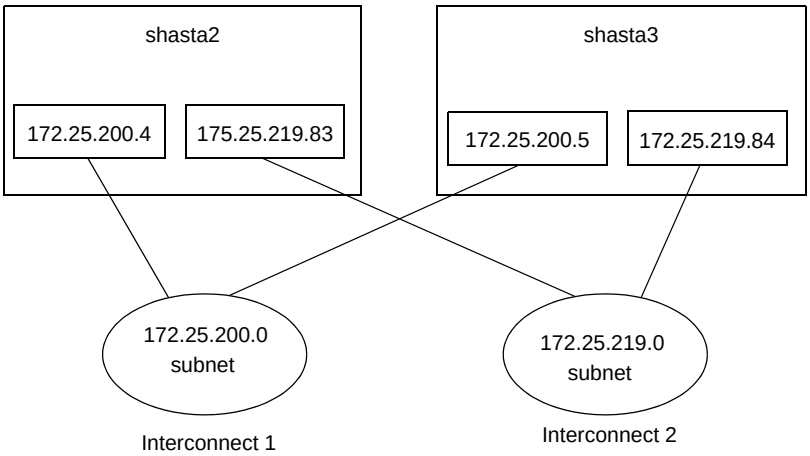

Figure 87: CF with IP interconnects

It is also possible to use mixed configurations in which CF is run over both Ethernet devices and IP subnetworks.

When using CF over IP, you should make sure that each node in the cluster has an IP interface on each subnetwork used as an interconnect. You should also make sure that all the interfaces for a particular subnetwork use the same IP broadcast address and the same netmask on all cluster nodes. This is particularly important since CF depends on an IP broadcast on each subnet to do its initial cluster join processing.

- IPv4 address is used for CF over IP.
- CF is not allowed to reach nodes that are on different subnets.

Caution
When selecting a subnetwork to use for CF, you should use a private subnetwork that only cluster nodes can access. CF security is based on access to its interconnects. Any node that can access an interconnect can join the cluster and acquire root privileges on any cluster node. When CF over IP is used, this means that any node on the subnetworks used by CF must be trusted. You should not use the public interface to a cluster node for CF over IP traffic unless you trust every node on your public network.

10.2 Configuring CF over IP

To configure CF over IP, you should do the following:

- Designate which subnetworks you want to use for CF over IP. Up to four subnetworks can be used.
- Make sure that each node that is to be in the cluster has IP interfaces properly configured for each subnetwork. Make sure the IP broadcast and netmasks are correct and consistent on all nodes for the subnetworks.
- Make sure that all of these IP interfaces are up and running.
- Run the CF Wizard in Cluster Admin.

The CF Wizard has a window which allows CF over IP to be configured. The Wizard will probe all the nodes that will be in the cluster, find out what IP interfaces are available on each, and then offer them as choices in the CF over IP window. It will also try to group the choices for each node by subnetworks. See Section "CF, CIP, and CIM configuration" for details.

CF uses special IP devices to keep track of CF over IP configuration. There are four of these devices named as follows:

/dev/ip0
/dev/ip1
/dev/ip2
/dev/ip3

These devices do not actually correspond to any device files under /dev in the Solaris. Instead, they are just place holders for CF over IP configuration information within the CF product. Any of these devices can have an IP address and broadcast address assigned by the cfconfig(1M) command (or by Cluster Admin which invokes the cfconfig(1M) command in the Wizard).

If you run cfconfig(1M) by hand, you may specify any of these devices to indicate you want to run CF over IP. The IP device should be followed by an IP address and broadcast address of an interface on the local node. The addresses must be in Internet dotted-decimal notation. For example, to configure CF on shasta2 in Figure 87, the cfconfig(1M) command would be as follows:

shasta2 # **cfconfig -S clustername /dev/ip0** 172.25.200.4 172.25.200.255 /dev/ip1 172.25.219.83 172.25.219.87

It really does not matter which IP device you use. The above command could equally have used /dev/ip2 and /dev/ip3.

The cfconfig(1M) command does not do any checks to make sure that the IP addresses are valid.

The IP devices chosen in the configuration will appear in other commands such as cftool -d and cftool -r.

IP interfaces will not show up in CF pings using cftool -p unless they are configured for use with CF and the CF driver is loaded.

cftool -d shows a relative speed number for each device, which is used to establish priority for the message send. If the configured device is IP, the relative speed 100 is used. This is the desired priority for the logical IP device. If a Gigabit Ethernet hardware device is also configured, it will have priority.

11 CF messages and codes

This chapter is a printed version of information that can be found on the PRIMECLUSTER CD. Messages on newer CDs might differ slightly.

This chapter discusses the following:

- The Section "cfconfig messages" discusses the cfconfig(1M) command and it's error messages.
- The Section "cipconfig messages" describes the cipconfig(1M) command and it's messages.
- The Section "cftool messages" details the cftool(1M) command and it's messages.
- The Section "rcqconfig messages" discusses the rcqconfig(1M) command and it's messages.
- The Section "rcqquery messages" describes the rcqquery(1M) command and it's messages.
- The Section "CF runtime messages" discusses Cluster Foundation (CF) runtime messages.
- The Section "CF Reason Code table" lists CF reason codes.
- The Section "Error messages for different systems" provides a pointer for accessing error messages for different systems.
- The Section "Linux ERRNO table" lists error messages for Linux by number.
- The Section "Cluster Resource Management Facility Messages" explains the Resource Database messages.
- The Section "Shutdown Facility" lists messages, causes, and actions for the Shutdown Facility (SF).
- The Section "CCBR messages" provides information on Cluster Configuration Backup and Restore (CCBR) messages.

The following lexicographic conventions are used in this chapter:

- Messages that will be generated on stdout or stderr are shown on the first line(s).
- Explanatory text is given after the message.
- Messages that will be generated in the system-log file and may optionally appear on the console are listed after the explanation.

- Message text tokens shown in a italic font style are placeholders for substituted text.
- Many messages include a token of the form #0407, which always denotes a hexadecimal reason code. The Section "CF Reason Code table" has a complete list of these codes.

11.1 cfconfig messages

The cfconfig(1M) command will generate an error message on stderr if an error occurs. Additional messages giving more detailed information about this error may be generated by the support routines in the libcf library. However, these additional messages will only be written to the system log file, and will not appear on stdout or stderr.

to the cfconfig(1M) manual page for an explanation of the command options and the associated functionality. The cfconfig(1M) manual page also describes the format of all non-error related command output.

11.1.1 Usage message

A usage message will be generated if:

- Multiple cfconfig(1M) options are specified (all options are mutually exclusive).
- An invalid cfconfig(1M) option is specified.
- No cfconfig(1M) option is specified.
- The -h option is specified.

Usage:

cfconfig [-d|-G|-g|-h|-L|-l|-S *nodename clustername device* [*device* [...]] |-s *clustername device* [*device* [...]]|-u]

-d delete configuration
-g get configuration
-G get configuration including address information
-h help
-L fast load (use configured devicelist)
-l load

-S set configuration (including nodename)
-s set configuration
-u unload

A device can ether be a network device or an IP device like /dev/ip[0-3] followed by the IP-Address and Broadcast Address number.

11.1.2 Error messages

cfconfig –l

cfconfig: cannot load: #0423: generic: permission denied

The CF startup routine has failed. This error message usually indicates that an unprivileged user has attempted to start CF. You must have administrative privileges to start, stop, and configure CF. An additional error message, for this case, will also be generated in the system-log file:

OSDU_start: failed to open /dev/cf (EACCES)

cfconfig: cannot load: #041f: generic: no such file or directory
cfconfig: check that configuration has been specified

The CF startup routine has failed. This error message usually indicates that the CF configuration file /etc/default/cluster cannot be found. Additional error messages, for this case, may also be generated in the system-log file:

OSDU_getconfig: failed to open config file (*errno*)
OSDU_getconfig: failed to stat config file (*errno*)

cfconfig: cannot load: #0405: generic: no such device/resource
cfconfig: check if configuration entries match node's device list

The CF startup routine has failed. This error message usually indicates that the CF configuration file does not match the physical hardware (network interfaces) installed in/on the node.

cfconfig: cannot load: #04*xx*: generic: *reason_text*

The CF startup routine has failed. One cause of an error message of this pattern is that the CF cluster configuration file has been damaged or is missing. If you think this is the case, delete and then re-specify your cluster configuration information, and try the command again. If the same error persists, see below.

Additional error messages, for this case, will also be generated in the system-log file:

OSDU_getconfig: corrupted config file
OSDU_getconfig: failed to open config file (*errno*)
OSDU_getconfig: failed to stat config file (*errno*)
OSDU_getconfig: read failed (*errno*)

Another cause of an error message of this pattern is that the CF driver and/or other kernel components may have somehow been damaged. Remove and then re-install the CF package. If this does not resolve the problem, contact field engineers. Additional error messages, for this case, will also be generated in the system-log file:

OSDU_getconfig: malloc failed
OSDU_getstatus: mconn status ioctl failed (*errno*)
OSDU_nodename: malloc failed
OSDU_nodename: uname failed (*errno*)
OSDU_start: failed to get configuration
OSDU_start: failed to get nodename
OSDU_start: failed to kick off join
OSDU_start: failed to open /dev/cf (*errno*)
OSDU_start: failed to open /dev/mconn (*errno*)
OSDU_start: failed to select devices
OSDU_start: failed to set clustername
OSDU_start: failed to set nodename
OSDU_start: icf_devices_init failed

OSDU_start: icf_devices_setup failed
OSDU_start: IOC_SOSD_DEVSELECTED ioctl failed
OSDU_start: netinit failed

If the device driver for any of the network interfaces to be used by CF responds in an unexpected way to DLPI messages, additional message output (in the system-log) may occur, with no associated command error message. These messages may be considered as warnings, unless a desired network interface cannot be configured as a cluster interconnect. These messages are:

dl_attach: DL_ACCESS error
dl_attach: DL_ATTACH_REQ putmsg failed (*errno*)
dl_attach: DL_BADPPA error
dl_attach: DL_OUTSTATE error
dl_attach: DL_SYSERR error
dl_attach: getmsg for DL_ATTACH response failed (*errno*)
dl_attach: unknown error
dl_attach: unknown error *hexvalue*
dl_bind: DL_ACCESS error
dl_bind: DL_BADADDR error
dl_bind: DL_BIND_REQ putmsg failed (*errno*)
dl_bind: DL_BOUND error
dl_bind: DL_INITFAILED error
dl_bind: DL_NOADDR error
dl_bind: DL_NOAUTO error
dl_bind: DL_NOTESTAUTO error
dl_bind: DL_NOTINIT error
dl_bind: DL_NOXIDAUTO error
dl_bind: DL_OUTSTATE error
dl_bind: DL_SYSERR error
dl_bind: DL_UNSUPPORTED error
dl_bind: getmsg for DL_BIND response failed (*errno*)
dl_bind: unknown error

dl_bind: unknown error *hexvalue*
dl_info: DL_INFO_REQ putmsg failed (*errno*)
dl_info: getmsg for DL_INFO_ACK failed (*errno*)

It is also possible that while CF is examining the kernel device tree, looking for eligible network interfaces, that a device or streams responds in an unexpected way. This may trigger additional message output in the system-log, with no associated command error message. These messages may be considered as warnings, unless a desired network interface cannot be configured as a cluster interconnect. These messages are:

get_net_dev: cannot determine driver name of *nodename* device
get_net_dev: cannot determine instance number of *nodename* device
get_net_dev: device table overflow - ignoring /dev/*drivernameN*
get_net_dev: dl_attach failed: /dev/*drivernameN*
get_net_dev: dl_bind failed: /dev/*drivernameN*
get_net_dev: dl_info failed: /dev/*drivername*
get_net_dev: failed to open device: /dev/*drivername* (*errno*)
get_net_dev: not an ethernet device: /dev/*drivername*
get_net_dev: not DL_STYLE2 device: /dev/*drivername*
icf_devices_init: cannot determine instance number of *drivername* device
icf_devices_init: device table overflow - ignoring /dev/scin
icf_devices_init: di_init failed
icf_devices_init: di_prom_init failed
icf_devices_init: dl_bind failed: /dev/scin
icf_devices_init: failed to open device: /dev/scin (*errno*)
icf_devices_init: no devices found
icf_devices_select: *devname* device not found
icf_devices_select: fstat of mc1x device failed: /devices/pseudo/icf*n* - *devname* (*errno*)
icf_devices_select: mc1_select_dev failed: /devices/pseudo/icf*n* – *devname* (*errno*)
icf_devices_select: open of mc1x device failed: /devices/pseudo/icf*n* – *devname* (*errno*)
icf_devices_setup: calloc failed: *devname*
icf_devices_setup: failed to create mc1x dev: /devices/pseudo/icf*n* – *devname* (*errno*)
icf_devices_setup: failed to open /dev/kstat (*errno*)
icf_devices_setup: failed to open mc1x device: /devices/pseudo/icf*n* – *devname* (*errno*)
icf_devices_setup: failed to stat mc1x device: /dev/mc1x (*errno*)
icf_devices_setup: failed to stat mc1x device: /devices/pseudo/icf*n* – *devname* (*errno*)
icf_devices_setup: I_LIST failed: *devname* (*errno*)icf_devices_setup: I_LIST 0 failed: *devname* (*errno*)

icf_devices_setup: I_PLINK failed: /devices/pseudo/icf*n* – *devname*(*errno*)
icf_devices_setup: I_POP failed: *devname* (*errno*)
icf_devices_setup: I_PUSH failed: *devname* (*errno*)
icf_devices_setup: mc1_set_device_id failed: /devices/pseudo/icf*n* – *devname* (*errno*)
icf_devices_setup: mc1x_get_device_info failed: /devices/pseudo/icf*n* – *devname* (*errno*)
icf_devices_setup: mc1x device already linked: /devices/pseudo/icf*n* – *devname*(*errno*)
icf_devices_setup: mc1x not a device
mc1_select_device: MC1_IOC_SEL_DEV ioctl failed (*errno*)
mc1_set_device_id: MC1_IOC_SET_ID ioctl failed (*errno*)
mc1x_get_device_info: MC1X_IOC_GET_INFO ioctl failed (*errno*)

cfconfig –u

cfconfig: cannot unload: #0406: generic: resource is busy
cfconfig: check if dependent service-layer module(s) active

The CF shutdown routine has failed. This error message is generated if a PRIMECLUSTER Layered Service still has a CF resource active/allocated. RMS, OPS, CIP, and so forth, need to be stopped before CF can be unloaded. Please to the layered-products software README file on how to stop these software. An additional error message, for this case, will also be generated in the system-log file:

OSDU_stop: failed to unload cf_drv

In the special case where the cfconfig(1M) command was called by a shutdown script that is rebooting the system, the following additional error message is generated in the system-log file:

OSDU_stop: runlevel now n: sent EVENT_NODE_LEAVING_CLUSTER (#*xxxx*)

cfconfig: cannot unload: #0423: generic: permission denied

The CF shutdown routine has failed. This error message usually indicates that an unprivileged user has attempted to stop CF. You must have administrative privileges to start, stop, and configure CF. An additional error message, for this case, will also be generated in the system-log file:

OSDU_stop: failed to open /dev/cf (EACCES)

cfconfig: cannot unload: #04*xx*: generic: *reason_text*

The cause of an error message of this pattern is that the CF driver and/or other kernel components may have somehow been damaged. Remove and then re-install the CF package. If this does not resolve the problem, contact field engineers. Additional error messages, for this case, will also be generated in the system-log file:

mc1x_get_device_info: MC1X_IOC_GET_INFO ioctl failed (*errno*)
OSDU_stop: disable unload failed
OSDU_stop: enable unload failed
OSDU_stop: failed to open /dev/cf (*errno*)
OSDU_stop: failed to open mc1x device: /devices/pseudo/icf*n* (*errno*)
OSDU_stop: failed to unlink mc1x device: /devices/pseudo/icf*n* (*errno*)
OSDU_stop: failed to unload cf_drv
OSDU_stop: failed to unload mc1 module
OSDU_stop: failed to unload mc1x driver
OSDU_stop: mc1x_get_device_info failed: /devices/pseudo/icfn

cfconfig –s
cfconfig -S

cfconfig: specified nodename: bad length: #407: generic: invalid parameter

This usually indicates that nodename is too long. The maximum length is 31 characters.

cfconfig: invalid nodename: #407: generic: invalid parameter

This indicates that nodename contains one or more non-printable characters.

cfconfig: node already configured: #0406: generic: resource is busy

This error message usually indicates that there is an existing CF configuration. To change the configuration of a node, you must first delete (cfconfig –d) any pre-existing configuration. Also, you must have administrative privileges to start, stop, and configure CF. A rare cause of this error would be that the CF driver and/or other kernel components have somehow been damaged. If you believe this is the case, remove and then re-install the CF package. If this does not resolve the problem, contact field engineers. Additional error messages may also be generated in the system-log file:

OSDU_getconfig: corrupted config file
OSDU_getconfig: failed to open config file (*errno*)
OSDU_getconfig: failed to stat config file(*errno*)
OSDU_getconfig: malloc failed
OSDU_getconfig: read failed(*errno*)

cfconfig: too many devices specified: #0407: generic: invalid parameter

Too many devices have been specified on the command line. The current limit is set to 255.

cfconfig: clustername cannot be a device: #0407: generic: invalid parameter

This error message indicates that "clustername," is a CF-eligible device. This usually means that the clustername has accidentally been omitted.

cfconfig: invalid clustername: #0407: generic: invalid parameter

This error message indicates that clustername is a CF-eligible device.

cfconfig: duplicate device names specified: #0407: generic: invalid parameter

This error message indicates that duplicate device names have been specified on the command line. This is usually a typographical error, and it is not permitted to submit a device name more than once.

cfconfig: device [device […]]: #0405: generic: no such device/resource

This error message indicates that the specified device names are not CF-eligible devices. Only those devices displayed by cftool -d are CF-eligible devices.

cfconfig: cannot open mconn: #04*xx*: generic: *reason_text*

This message should not occur unless the CF driver and/or other kernel components have somehow been damaged. Remove and then re-install the CF package. If the problem persists, contact field engineers.

cfconfig: cannot set configuration:
#04*xx*: generic: *reason_text*

This message can occur if concurrent cfconfig -s or cfconfig -S commands are being run. Otherwise, it should not occur unless the CF driver and/or other kernel components have somehow been damaged. If this is the case, remove and then re-install the CF package. If the problem persists, contact field engineers. Additional error messages may also be generated in the system-log file:

OSDU_setconfig: config file exists
OSDU_setconfig: failed to create config file (*errno*)
OSDU_setconfig: write failed (*errno*)

cfconfig: cannot get new configuration: #04*xx*: generic: *reason_text*

This message indicates that the saved configuration cannot be read back. This may occur if concurrent cfconfig -s or cfconfig -S commands are being run, or if disk hardware errors are reported. Otherwise, it should not occur unless the CF driver and/or other kernel components have somehow been damaged. If this is the case, remove and then re-install the CF package. If the problem persists, contact field engineers. Additional error messages may also be generated in the system-log file:

OSDU_getconfig: corrupted config file
OSDU_getconfig: failed to open config file (*errno*)
OSDU_getconfig: failed to stat config file (*errno*)
OSDU_getconfig: malloc failed
OSDU_getconfig: read failed (*errno*)

cfconfig: cannot load: #04
xx: generic: *reason_text*

This error message indicates that the device discovery portion of the CF startup routine has failed. (See error messages associated with cfconfig -l above).

cfconfig –g

cfconfig: cannot get configuration: #04*xx*: generic: *reason_text*

This message indicates that the CF configuration cannot be read. This may occur if concurrent cfconfig(1M) commands are being run, or if disk hardware errors are reported. Otherwise, it should not occur unless the CF driver and/or other kernel components have somehow been damaged. If this is the case, remove and then re-install the CF package. If the problem persists, contact field engineers. Additional error messages may also be generated in the system-log file:

OSDU_getconfig: corrupted config file
OSDU_getconfig: failed to open config file (*errno*)
OSDU_getconfig: failed to stat config file (*errno*)
OSDU_getconfig: malloc failed
OSDU_getconfig: read failed (*errno*)

cfconfig –d

cfconfig: cannot get joinstate: #0407: generic: invalid parameter

This error message usually indicates that the CF driver and/or other kernel components have somehow been damaged. Remove and then re-install the CF package. If this does not resolve the problem, contact field engineers.

cfconfig: cannot delete configuration: #0406: generic: resource is busy

This error message is generated if CF is still active (i.e., if CF resource(s) are active/allocated). The configuration node may not be deleted while it is an active cluster member.

cfconfig: cannot delete configuration: #04*xx*: generic: *reason_text*

You must have administrative privileges to start, stop, and configure CF. A rare cause of this error would be that the CF driver and/or other kernel components have somehow been damaged. If you believe this is the case, remove and then re-install the CF package. If this does not resolve the problem, contact field engineers. An additional error message will also be generated in the system-log file:

OSDU_delconfig: failed to delete config file (*errno*)

11.2 cipconfig messages

The cipconfig(1M) command will generate an error message on stderr if an error occurs. Additional error messages giving more detailed information about the error may be generated by the support routines of the libcf library. However, these additional messages will only be written to the system-log file, and will not appear on stdout or stderr.

to the cipconfig(1M) manual page for an explanation of the command options and associated functionality. The cipconfig(1M) manual page also describes the format of all non-error related command output.

11.2.1 Usage message

A usage message will be generated if:

- Multiple cipconfig(1M) options are specified (all options are mutually exclusive).
- An invalid cipconfig(1M) option is specified.
- No cipconfig(1M) option is specified.
- The -h option is specified.

usage: cipconfig [-l|-u|-h]

-l start/load
-u stop/unload
-h help

11.2.2 Error messages

cipconfig –l

cipconfig: could not start CIP - detected a problem with CF.
cipconfig: cannot open mconn: #04*xx*: generic: *reason_text*

These messages should not occur unless the CF driver and/or other kernel components have somehow been damaged. Remove and then re-install the CF package. If the problem persists, contact field engineers.

cipconfig: cannot setup cip: #04*xx*: generic: *reason_text*

The cip startup routine(s) have failed. There may be problems with the configuration file. Additional error messages will be generated in the system-log file:

OSDU_cip_start: cip kickoff failed (*errno*)
OSDU_cip_start: dl_attach failed: *devpathn*
OSDU_cip_start: dl_bind failed: *devpathn*
OSDU_cip_start: dl_info failed: *devpath*
OSDU_cip_start: failed to open device: /dev/cip (*errno*)
OSDU_cip_start: failed to open device: *devpath* (*errno*)
OSDU_cip_start: I_PLINK failed: *devpath* (*errno*)
OSDU_cip_start: POPing module failed: *errno*
OSDU_cip_start: ppa n is not valid: *devpath*
OSDU_cip_start: setup controller/speed failed: *devpath* (*errno*)

If the device driver for any of the network interfaces used by CIP responds in an unexpected way to DLPI messages, additional message output may occur:

dl_info: DL_INFO_REQ putmsg failed (*errno*)
dl_info: getmsg for DL_INFO_ACK failed (*errno*)
dl_attach: DL_ACCESS error
dl_attach: DL_ATTACH_REQ putmsg failed (*errno*)
dl_attach: DL_BADPPA error
dl_attach: DL_OUTSTATE error
dl_attach: DL_SYSERR error
dl_attach: getmsg for DL_ATTACH response failed (*errno*)
dl_attach: unknown error
dl_attach: unknown error *hexvalue*
dl_bind: DL_ACCESS error
dl_bind: DL_BADADDR error
dl_bind: DL_BIND_REQ putmsg failed (*errno*)
dl_bind: DL_BOUND error
dl_bind: DL_INITFAILED error
dl_bind: DL_NOADDR error

dl_bind: DL_NOAUTO error
dl_bind: DL_NOTESTAUTO error
dl_bind: DL_NOTINIT error
dl_bind: DL_NOXIDAUTO error
dl_bind: DL_OUTSTATE error
dl_bind: DL_SYSERR error
dl_bind: DL_UNSUPPORTED error
dl_bind: getmsg for DL_BIND response failed (*errno*)
dl_bind: unknown error
dl_bind: unknown error *hexvalue*

If these messages appear and they do not seem to be associated with problems in your CIP configuration file, contact field engineers.

cipconfig –u

cipconfig: cannot unload cip: #04*xx*: generic: *reason_text*

The CIP shutdown routine has failed. Usually this mean that another PRIMECLUSTER Layered Service has a CIP interface open (active). It must be stopped first. Additional error messages may be generated in the system-log file:

OSDU_cip_stop: failed to unload cip driver
OSDU_cip_stop: failed to open device: /dev/cip (*errno*)

11.3 cftool messages

The cftool(1M) command will generate an error message on stderr if an error condition is detected. Additional messages, giving more detailed information about this error, may be generated by the support routines of the libcf library. Note that these additional error messages will only be written to the system-log file, and will not appear on stdout or stderr.

to the cftool(1M) manual page for an explanation of the command options and the associated functionality. The cftool(1M) manual page also describes the format of all non-error related command output.

11.3.1 Usage message

A usage message will be generated if:

- Conflicting cftool(1M) options are specified (some options are mutually exclusive).

- An invalid cftool(1M) option is specified.
- No cftool(1M) option is specified.
- The -h option is specified.

usage: cftool [-c][-l][-n][-r][-d][-v][-p][-e][-i *nodename*][-A *cluster*][-T *timeout*][-F][-C *count*][-I *nodename*][-E xx.xx.xx.xx.xx.xx][-P][-m][-u][-k][-q][-h]

-c		clustername
-l		local nodeinfo
-n		nodeinfo
-r		routes
-d		devinfo
-v		version
-p		ping
-e		echo
-i		icf stats for nodename
-m		mac stats
-u		clear all stats
-k		set node status to down
-q		quiet mode
-h		help
-F		flush ping queue. Be careful, please
-T	timeout	millisecond ping timeout
-I		raw ping test by node name
-P		raw ping
-A	cluster	ping all interfaces in one cluster
-E	xx.xx.xx.xx.xx.xx	raw ping by 48-bit physical address
-C	count	stop after sending count raw ping messages

A device can either be a network device or an IP device like /dev/ip[0-3] followed by IP address and broadcast address.

11.3.2 Error messages

cftool –c

cftool: failed to get cluster name: #*xxxx*: service: *reason_text*

This message should not occur unless the CF driver and/or other kernel components have somehow been damaged. Remove and then re-install the CF package. If the problem persists, contact field engineers.

cftool –d

cftool: cannot open mconn: #04*xx*: generic: *reason_text*

This message should not occur unless the CF driver and/or other kernel components have somehow been damaged. Remove and then re-install the CF package. If the problem persists, contact field engineers.

cftool –e

cftool: cannot open mconn: #04*xx*: generic: *reason_text*

This message should not occur unless the CF driver and/or other kernel components have somehow been damaged. Remove and then re-install the CF package. If the problem persists, contact field engineers.

cftool –i nodename

cftool: *nodename*: No such node
cftool: cannot get node details: #*xxxx*: service: *reason_tex*t

Either of these messages indicates that the specified nodename is not an active cluster node at this time.

cftool: cannot open mconn: #04*xx*: generic: *reason_text*

This message should not occur unless the CF driver and/or other kernel components have somehow been damaged. Remove and then re-install the CF package. If the problem persists, contact field engineers.

cftool –k

cftool(down): illegal node number

This message indicates that the specified node number is non-numeric or is out of allowable range (1–64).

cftool(down): not executing on active cluster node

This message is generated if the command is executed either on a node that is not an active cluster node or on the specified LEFTCLUSTER node itself.

cftool(down): cannot declare node down: #0426: generic: invalid node name
cftool(down): cannot declare node down: #0427: generic: invalid node number
cftool(down): cannot declare node down: #0428: generic: node is not in LEFTCLUSTER state

One of these messages will be generated if the supplied information does not match an existing cluster node in LEFTCLUSTER state.

cftool(down): cannot declare node down: #*xxxx*: service: *reason_text*

Other variations of this message should not occur unless the CF driver and/or other kernel components have somehow been damaged. Remove and then re-install the CF package. If the problem persists, contact field engineers.

cftool –l

cftool: cannot get nodename: #04*xx*: generic: *reason_text*
cftool: cannot get the state of the local node: #04*xx*: generic: *reason_text*

These messages should not occur unless the CF driver and/or other kernel components have somehow been damaged. Remove and then re-install the CF package. If the problem persists, contact field engineers.

cftool –m

cftool: cannot open mconn: #04*xx*: generic: *reason_text*
cftool: cannot get icf mac statistics: #04*xx*: generic: *reason_text*

These messages should not occur unless the CF driver and/or other kernel components have somehow been damaged. Remove and then re-install the CF package. If the problem persists, contact field engineers.

cftool –n

cftool: cannot get node id: #*xxxx*: service: *reason_text*
cftool: cannot get node details: #*xxxx*: service: *reason_text*

This message should not occur unless the CF driver and/or other kernel components have somehow been damaged. Remove and then re-install the CF package. If the problem persists, contact field engineers.

cftool –p

cftool: cannot open mconn: #04*xx*: generic: *reason_text*

This message should not occur unless the CF driver and/or other kernel components have somehow been damaged. Remove and then re-install the CF package. If the problem persists, contact field engineers.

cftool –r

cftool: cannot get node details: #*xxxx*: service: *reason_text*

These messages should not occur unless the CF driver and/or other kernel components have somehow been damaged. Remove and then re-install the CF package. If the problem persists, contact field engineers.

cftool –u

cftool: cannot open mconn: #04*xx*: generic: *reason_text*
cftool: clear icf statistics: #04*xx*: generic: *reason_text*

These messages should not occur unless the CF driver and/or other kernel components have somehow been damaged. Remove and then re-install the CF package. If the problem persists, contact field engineers.

cftool –v

cftool: cannot open mconn: #04*xx*: generic: *reason_text*
cftool: unexpected error retrieving version: #04*xx*: generic: *reason_text*

These messages should not occur unless the CF driver and/or other kernel components are damaged. Remove and then re-install the CF package. If the problem persists, contact field engineers.

11.4 rcqconfig messages

The rcqconfig(1M) command will generate an error message on standard error if an error condition is detected. Additional messages, giving more detailed information about this error, may be generated by the support routines of the libcf library. Please note that these additional error messages will only be written to the system-log file during cfconfig -l, and will not appear on standard out or standard error.

to the rcqconfig(1M) manual page for an explanation of the command options and the associated functionality.

11.4.1 Usage message

A usage message will be generated if:

- Conflicting rcqconfig(1M) options are specified (some options are mutually exclusive).
- An invalid rcqconfig(1M) option is specified.
- The '-h' option is specified.

```
usage: rcqconfig [ -g | -h ] or
    rcqconfig -s or
    rcqconfig [ -v ] [ -c ]
        [ -a Add-node-1 ... Add-node-n ]
```

[-x Ignore-node-1 ... Ignore-node-n]

[-d Delete-node-1 ... Delete-node-n]

[-m quorum-method-1 ... quorum-method-n]

11.4.2 Error messages

rcqconfig -a node-1 node-2 …. node-n
-g and -a cannot exist together.

This error message usually indicates that get configuration option (-g) cannot be specified with this option (-a). to the manual pages for the correct syntax definition.

Nodename is not valid nodename.

This error message usually indicates that the length of the node is less than 1 or greater than 31 bytes. to the manual pages for the correct syntax definition.

rcqconfig : failed to start

The following errors will also be reported in standard error if rcqconfig(1M) fails to start.

rcqconfig failed to configure qsm since quorum node set is empty.

Quorum state machine (qsm) is the kernel module that collects the states of the cluster nodes specified in the quorum node set. This error message usually indicates that the quorum configuration does not exist. to the manual pages for rcqconfig(1M) for the correct syntax to configure the quorum nodes.

cfreg_start_transaction: `#2813: cfreg daemon not present`

The rcqconfig(1M) routine has failed. This error message usually indicates that the synchronization daemon is not running on the node. The cause of error messages of this pattern may be that the cfreg daemon has died and the previous error messages in the system log or console will indicate why the daemon died. Restart the daemon using cfregd -r. If it fails again, the error messages associated with it will indicate the problem. The data in the registry is most likely corrupted. If the problem persists, contact field engineers.

cfreg_start_transaction: `#2815: registry is busy`

The rcqconfig(1M) routine has failed. This error message usually indicates that the daemon is not in synchronized state or if the transaction has been started by another application. This messages should not occur. The cause of error messages of this pattern is that the registries are not in consistent state. If the problem persists, unload the cluster by using cfconfig -u and reload the cluster by using cfconfig -l. If the problem still persists, remove and then re-install the CF package. If this does not resolve the problem, contact field engineers.

cfreg_start_transaction: `#2810: an active transaction exists`

The rcqconfig(1M) routine has failed. This error message usually indicates that the application has already started a transaction. If the cluster is stable, the cause of error messages of this pattern is that different changes may be done concurrently from multiple nodes. Therefore, it might take longer time to commit. Retry the command again. If the problem persists, the cluster might not be in a stable state. The error messages in the log will indicate the problem. If this is the case, unload the cluster by using cfconfig -u and reload the cluster by using cfconfig -l. If the problem persists, remove and then re-install the CF package. If this does not resolve the problem, contact field engineers.

Too many nodenames are defined for quorum. Max node = 64

This error message usually indicates that if the number of node specified are more than 64 for which the quorum is to be configured. The following errors will also be reported in standard error if there are too many nodenames defined:

cfreg_get: `#2809: specified transaction invalid`

The rcqconfig(1M) routine has failed. This error message usually indicates that the information supplied to get the specified data from the registry is not valid (e.g. transaction aborted due to time period expiring or synchronization daemon termination, etc.). This messages should not occur. Try to unload the cluster by using cfconfig -u and reload the cluster by using cfconfig -l. If the problem persists, remove and then re-install the CF package. If this does not resolve the problem, contact field engineers.

cfreg_get: `#2819: data or key buffer too small`

The rcqconfig(1M) routine has failed. This error message usually indicates that the specified size of the data buffer is too small to hold the entire data for the entry. The cause of error messages of this pattern is that the memory image may have somehow been damaged. Try to unload the cluster by using cfconfig -u and reload the cluster by using cfconfig -l. If the problem persists, remove and then re-install the CF package. If this does not resolve the problem, contact field engineers.

Cannot add node node that is not up.

This error message usually indicates that the user is trying to add a node whose state is not up in the NSM node space. Try to bring up the down node or remove the node from the list which quorum is to be configured.

Cannot proceed. Quorum node set is empty.

This error message usually indicates that if no node is specified to this option or there is no configured node prior to this call. The following errors will also be reported in standard error if rcqconfig(1M) fails to start:

cfreg_put: `#2809: specified transaction invalid`

The rcqconfig(1M) routine has failed. This error message usually indicates that the information supplied to get the specified data from the registry is not valid (e.g. transaction aborted due to time period expiring or synchronization daemon termination, etc.). This messages should not occur. Try to unload the cluster by using cfconfig -u and reload the cluster by using cfconfig -l. If the problem persists, remove and then re-install the CF package. If this does not resolve the problem, contact field engineers.

cfreg_put: `#2820: registry entry data too large`

The rcqconfig(1M) routine has failed. This error message usually indicates that the specified size data is larger than 28K. The cause of error messages of this pattern is that the memory image may have somehow been damaged. Try to unload the cluster by using cfconfig -u and reload the cluster by using cfconfig -l. If the problem persists, remove and then re-install the CF package. If this does not resolve the problem, contact field engineers.

rcqconfig –s
stopping quorum space methods `#0408: unsuccessful`

The rcqconfig(1M) routine has failed. This error message usually indicates that there is no method specified.

rcqconfig -x ignore_node-1 … ignore_node-n

-g and -x cannot exist together.

This error message usually indicates that get configuration option (-g) cannot be specified with this option (-m). to the manual pages for the correct syntax definition.

Nodename is not valid nodename.

This error message usually indicates that the length of the node is less than 1 or greater than 31 bytes.

rcqconfig : failed to start

The following errors will also be reported in standard error if rcqconfig(1M) fails to start:

cfreg_start_transaction: `#2813: cfreg daemon not present`

The rcqconfig(1M) routine has failed. This error message usually indicates that the synchronization daemon is not running on the node. The cause of error messages of this pattern may be that the cfreg daemon has died and the previous error messages in the system log or console will indicate why the daemon died. Restart the daemon using cfregd -r. If it fails again, the error messages associated with it will indicate the problem. The data in the registry is most likely corrupted. If the problem persists, contact field engineers.

cfreg_start_transaction: `#2815: registry is busy`

The rcqconfig(1M) routine has failed. This error message usually indicates that the daemon is not in synchronized state or if the transaction has been started by another application. This messages should not occur. If the problem persists, unload the cluster by using cfconfig -u and reload the cluster by using cfconfig -l. If the problem still persists, remove and then re-install the CF package. If this does not resolve the problem, contact field engineers.

cfreg_start_transaction: `#2810: an active transaction exists`

The rcqconfig(1M) routine has failed. This error message usually indicates that the application has already started a transaction. If the cluster is stable, the cause of error messages of this pattern is that different changes may be done concurrently from multiple nodes. Therefore, it might take longer time to commit. Retry the command again. If the problem persists, the cluster might not be in a stable state. The error messages in the log will indicate the problem. If this is the case, unload the cluster by using cfconfig -u and reload the cluster by using cfconfig -l. If the problem persists, remove and then re-install the CF package. If this does not resolve the problem, contact field engineers.

Too many ignore nodenames are defined for quorum. Max node = 64

This error message usually indicates that if the number of ignore nodes specified are more than 64. The following errors will also be reported in standard error if the ignore node names exceed 64.

cfreg_get: `#2809: specified transaction invalid`

The rcqconfig(1M) routine has failed. This error message usually indicates that the information supplied to get the specified data from the registry is not valid (e.g. transaction aborted due to time period expiring or synchronization daemon termination, etc.). This messages should not occur. Try to unload the cluster by using cfconfig -u and reload the cluster by using cfconfig -l. If the problem persists, remove and then re-install the CF package. If this does not resolve the problem, contact field engineers.

cfreg_get: `#2804: entry with specified key does not exist`

The rcqconfig(1M) routine has failed. This error message usually indicates that the specified entry does not exist. The cause of error messages of this pattern is that the memory image may have somehow been damaged. Try to unload the cluster by using cfconfig -u and reload the cluster by using cfconfig -l. If the problem persists, remove and then re-install the CF package. If this does not resolve the problem, contact field engineers.

cfreg_get: `#2819: data or key buffer too small`

The rcqconfig(1M) routine has failed. This error message usually indicates that the specified size of the data buffer is too small to hold the entire data for the entry. The cause of error messages of this pattern is that the memory image may have somehow been damaged. Try to unload the cluster by using cfconfig -u and reload the cluster by using cfconfig -l. If the problem persists, remove and then re-install the CF package. If this does not resolve the problem, contact field engineers.

Can not add node node that is not up.

This error message usually indicates that the user is trying to add a node whose state is not up in the NSM node space. Try to bring up the down node or remove the node from the list which quorum is to be configured.

Can not proceed. Quorum node set is empty.

This error message usually indicates that if no node is specified to this option or there is no configured node prior to this call. The following errors will also be reported in standard error if Quorum node set is empty:

cfreg_put: `#2809: specified transaction invalid`

The rcqconfig(1M) routine has failed. This error message usually indicates that the information supplied to get the specified data from the registry is not valid (e.g. transaction aborted due to time period expiring or synchronization daemon termination, etc.). This messages should not occur. Try to unload the cluster by using cfconfig -u and reload the cluster by using cfconfig -l. If the problem persists, remove and then re-install the CF package. If this does not resolve the problem, contact field engineers.

cfreg_put: `#2820: registry entry data too large`

The rcqconfig(1M) routine has failed. This error message usually indicates that the event information (data being passed to the kernel) to be used for other sub-systems) is larger than 32K. The cause of error messages of this pattern is that the memory image may have somehow been damaged. Try to unload the cluster by using cfconfig -u and reload the cluster by using cfconfig -l. If the problem persists, remove and then re-install the CF package. If this does not resolve the problem, contact field engineers.

cfreg_put: `#2807: data file format is corrupted`

The rcqconfig(1M) routine has failed. This error message usually indicates that the registry data file format has been corrupted. The cause of error messages of this pattern is that the memory image may have somehow been damaged. Try to unload the cluster by using cfconfig -u and reload the cluster by using cfconfig -l. If the problem persists, remove and then re-install the CF package. If this does not resolve the problem, contact field engineers.

cms_post_event: `#0c01: event information is too large`

The rcqconfig(1M) routine has failed. This error message usually indicates that the event information (data being passed to the kernel) to be used for other sub-systems) is larger than 32K. The cause of error messages of this pattern is that the memory image may have somehow been damaged. Try to unload the cluster by using cfconfig -u and reload the cluster by using cfconfig -l. If the problem persists, remove and then re-install the CF package. If this does not resolve the problem, contact field engineers.

rcqconfig -m method_name-1 … method_name -n

-g and -m cannot exist together.

This error message usually indicates that get configuration option (-g) cannot be specified with this option (-x). to the manual pages for the correct syntax definition.

Methodname is not valid method name.

This error message usually indicates that the length of the node is less than 1 or greater than 31 bytes. to the manual pages for the correct syntax definition.

rcqconfig : failed to start

The following errors will also be reported in standard error if rcqconfig(1M) fails to start:

cfreg_start_transaction: `#2813: cfreg daemon not present`

The rcqconfig(1M) routine has failed. This error message usually indicates that the synchronization daemon is not running on the node. The cause of error messages of this pattern may be that the cfreg daemon has died and the previous error messages in the system log or console will indicate why the daemon died. Restart the daemon using cfregd -r. If it fails again, the error messages associated with it will indicate the problem. The data in the registry is most likely corrupted. If the problem persists, contact field engineers.

cfreg_start_transaction: `#2815: registry is busy`

The rcqconfig(1M) routine has failed. This error message usually indicates that the daemon is not in synchronized state or if the transaction has been started by another application. This message should not occur. The cause of error messages of this pattern is that the registries are not in consistent state. If the problem persists, unload the cluster by using cfconfig -u and reload the cluster by using cfconfig -l. If the problem still persists, remove and then re-install the CF package. If this does not resolve the problem, contact field engineers.

cfreg_start_transaction: `#2810: an active transaction exists`

The rcqconfig(1M) routine has failed. This error message usually indicates that the application has already started a transaction. If the cluster is stable, the cause of error messages of this pattern is that different changes may be done concurrently from multiple nodes. Therefore, it might take longer time to commit. Retry the command again. If the problem persists, the cluster might not be in a stable state. If this is the case, unload the cluster by using cfconfig -u and reload the cluster by using cfconfig -l. If the problem persists, remove and then re-install the CF package. If this does not resolve the problem, contact field engineers.

Too many method names are defined for quorum. Max method = 8

This error message usually indicates that if the number of methods specified are more than 8. The following errors will also be reported in standard error if Quorum method names exceed the limit:

cfreg_get: `#2809: specified transaction invalid`

The rcqconfig(1M) routine has failed. This error message usually indicates that the information supplied to get the specified data from the registry is not valid (e.g. transaction aborted due to time period expiring or synchronization daemon termination, etc.). This message should not occur. Try to unload the cluster by using cfconfig -u and reload the cluster by using cfconfig -l. If the problem persists, remove and then re-install the CF package. If this does not resolve the problem, contact field engineers.

cfreg_get: `#2804: entry with specified key does not exist`

The rcqconfig(1M) routine has failed. This error message usually indicates that the specified entry does not exist. The cause of error messages of this pattern is that the memory image may have somehow been damaged. Try to unload the cluster by using cfconfig -u and reload the cluster by using cfconfig -l. If the problem persists, remove and then re-install the CF package. If this does not resolve the problem, contact field engineers.

cfreg_get: `#2819: data or key buffer too small`

The rcqconfig(1M) routine has failed. This error message usually indicates that the specified size of the data buffer is too small to hold the entire data for the entry. The cause of error messages of this pattern is that the memory image may have somehow been damaged. Try to unload the cluster by using cfconfig -u and reload the cluster by using cfconfig -l. If the problem persists, remove and then re-install the CF package. If this does not resolve the problem, contact field engineers.

cfreg_put: `#2809: specified transaction invalid`

The rcqconfig(1M) routine has failed. This error message usually indicates that the information supplied to get the specified data from the registry is not valid (e.g. transaction aborted due to time period expiring or synchronization daemon termination, etc.). This messages should not occur. Try to unload the cluster by using cfconfig -u and reload the cluster by using cfconfig -l. If the problem persists, remove and then re-install the CF package. If this does not resolve the problem, contact field engineers.

cfreg_put: `#2820: registry entry data too large`

The rcqconfig(1M) routine has failed. This error message usually indicates that the event information (data being passed to the kernel) to be used for other sub-systems) is larger than 32K. The cause of error messages of this pattern is that the memory image may have somehow been damaged. Try to unload the cluster by using cfconfig -u and reload the cluster by using cfconfig -l. If the problem persists, remove and then re-install the CF package. If this does not resolve the problem, contact field engineers.

cfreg_put: `#2807: data file format is corrupted`

The rcqconfig(1M) routine has failed. This error message usually indicates that the registry data file format has been corrupted. The cause of error messages of this pattern is that the memory image may have somehow been damaged. Try to unload the cluster by using cfconfig -u and reload the cluster by using cfconfig -l. If the problem persists, remove and then re-install the CF package. If this does not resolve the problem, contact field engineers.

cms_post_event: `#0c01: event information is too large`

The rcqconfig(1M) routine has failed. This error message usually indicates that the event information (data being passed to the kernel) to be used for other sub-systems) is larger than 32K. The cause of error messages of this pattern is that the memory image may have somehow been damaged. Try to unload the cluster by using cfconfig -u and reload the cluster by using cfconfig -l. If the problem persists, remove and then re-install the CF package. If this does not resolve the problem, contact field engineers.

rcqconfig -d node-1 node-2 .... node-n

-g and -d cannot exist together.

This error message usually indicates that get configuration option (-g) cannot be specified with this option (-d). to the manual pages for the correct syntax definition.

Nodename is not valid nodename.

This error message usually indicates that the length of the node is less than 1 or greater than 31 bytes. to the manual pages for the correct syntax definition.

rcqconfig : failed to start

The following errors will also be reported in standard error if rcqconfig(1M) fails to start:

cfreg_start_transaction: `#2813: cfreg daemon not present`

The rcqconfig(1M) routine has failed. This error message usually indicates that the synchronization daemon is not running on the node. The cause of error messages of this pattern may be that the cfreg daemon has died and the previous error messages in the system log or console will indicate why the daemon died. Restart the daemon using cfregd -r. If it fails again, the error messages associated with it will indicate the problem. The data in the registry is most likely corrupted. If the problem persists, contact field engineers.

cfreg_start_transaction: `#2815: registry is busy`

The rcqconfig(1M) routine has failed. This error message usually indicates that the daemon is not in synchronized state or if the transaction has been started by another application. This messages should not occur. The cause of error messages of this pattern is that the registries are not in consistent state. If the problem persists, unload the cluster by using cfconfig -u and reload the cluster by using cfconfig -l. If the problem still persists, remove and then re-install the CF package. If this does not resolve the problem, contact field engineers.

cfreg_start_transaction: `#2810: an active transaction exists`

The rcqconfig(1M) routine has failed. This error message usually indicates that the application has already started a transaction. If the cluster is stable, the cause of error messages of this pattern is that different changes may be done concurrently from multiple nodes. Therefore, it might take longer time to commit. Retry the command again. If the problem persists, the cluster might not be in a stable state. If this is the case, unload the cluster by using cfconfig -u and reload the cluster by using cfconfig -l. If the problem persists, remove and then re-install the CF package. If this does not resolve the problem, contact field engineers.

Too many nodename are defined for quorum. Max node = 64

This error message usually indicates that if the number of node specified are more than 64 for which the quorum is to be configured. The following errors will also be reported in standard error if nodename defined exceed the maximum limit:

cfreg_get: `#2809: specified transaction invalid`

The rcqconfig(1M) routine has failed. This error message usually indicates that the information supplied to get the specified data from the registry is not valid (e.g. transaction aborted due to time period expiring or synchronization daemon termination, etc.). This message should not occur. Try to unload the cluster by using cfconfig -u and reload the cluster by using cfconfig -l. If the problem persists, remove and then re-install the CF package. If this does not resolve the problem, contact field engineers.

cfreg_get: `#2804: entry with specified key does not exist`

The rcqconfig(1M) routine has failed. This error message usually indicates that the specified entry does not exist. The cause of error messages of this pattern is that the memory image may have somehow been damaged. Try to unload the cluster by using cfconfig -u and reload the cluster by using cfconfig -l. If the problem persists, remove and then re-install the CF package. If this does not resolve the problem, contact field engineers.

cfreg_get: `#2819: data or key buffer too small`

The rcqconfig(1M) routine has failed. This error message usually indicates that the specified size of the data buffer is too small to hold the entire data for the entry. The cause of error messages of this pattern is that the memory image may have somehow been damaged. Try to unload the cluster by using cfconfig -u and reload the cluster by using cfconfig -l. If the problem persists, remove and then re-install the CF package. If this does not resolve the problem, contact field engineers.

cfreg_put: `#2809: specified transaction invalid`

The rcqconfig(1M) routine has failed. This error message usually indicates that the information supplied to get the specified data from the registry is not valid (e.g. transaction aborted due to time period expiring or synchronization daemon termination, etc.). This message should not occur. Try to unload the cluster by using cfconfig -u and reload the cluster by using cfconfig -l. If the problem persists, remove and then re-install the CF package. If this does not resolve the problem, contact field engineers.

cfreg_put: `#2820: registry entry data too large`

The rcqconfig(1M) routine has failed. This error message usually indicates that the specified size data is larger than 28K. The cause of error messages of this pattern is that the memory image may have somehow been damaged. Try to unload the cluster by using cfconfig -u and reload the cluster by using cfconfig -l. If the problem persists, remove and then re-install the CF package. If this does not resolve the problem, contact field engineers.

cfreg_put: `#2807: data file format is corrupted`

The rcqconfig(1M) routine has failed. This error message usually indicates that the registry data file format has been corrupted. The cause of error messages of this pattern is that the memory image may have somehow been damaged. Try to unload the cluster by using cfconfig -u and reload the cluster by using cfconfig -l. If the problem persists, remove and then re-install the CF package. If this does not resolve the problem, contact field engineers.

cms_post_event: `#0c01: event information is too large`

The rcqconfig(1M) routine has failed. This error message usually indicates that the event information (data being passed to the kernel) to be used for other sub-systems) is larger than 32K. The cause of error messages of this pattern is that the memory image may have somehow been damaged. Try to unload the cluster by using cfconfig -u and reload the cluster by using cfconfig -l. If the problem persists, remove and then re-install the CF package. If this does not resolve the problem, contact field engineers.

11.5 rcqquery messages

The rcqquery(1M) command will generate an error message on stderr if an error condition is detected. Additional messages, giving more detailed information about this error, may be generated by the support routines of the libcf library. Please note that these additional error messages will only be written to the system-log file, and will not appear on stdout or stderr.

to the rcqquery(1M) manual page for an explanation of the command options and the associated functionality.

11.5.1 Usage message

A usage message will be generated if:

- An invalid rcqquery(1M) option is specified.
- The -h option is specified.

```
Usage:    rcqquery [ -v ] [ -l ] [-h]
    -v verbose
    -l loop
    -h help
```

11.5.2 Error messages

```
rcqquery -v -l
failed to register user event
`# 0c0b: user level ENS event memory limit overflow`
```

The rcqquery(1M) routine has failed. It usually indicates that either the total amount of memory allocated or the amount of memory allocated for use on a per-open basis exceed the limit. Try to unload the cluster by using cfconfig -u and reload the cluster by using cfconfig -l. If the problem persists, remove and then re-install the CF package. If this does not resolve the problem, contact field engineers.

11.6 CF runtime messages

All CF runtime messages include an 80-byte ASCII log3 prefix, which includes a timestamp, component number, error type, severity, version, product name, and structure id. This header is not included in the message descriptions that follow.

All of the following messages are sent to the system-log file, and ‘node up’ and ‘node down’ messages are also sent to the console.

There are some common tokens (shown in bold italic font) substituted into the error and warning messages that follow. If necessary, any not covered by this global explanation will be explained in the text associated with the specific message text.

- *clustername* — The name of the cluster to which the node belongs (or is joining). It is specified in the cluster configuration (see cfconfig -s).
- *err_type* — Identifies the type of ICF error reported. There are three types of errors:
  1. Debug (none in released product)
  2. Heartbeat missing
  3. Service error (usually, "route down")
- *nodename* — The name by which a node is known within a cluster (usually derived from uname -n).
- *nodenum* — A unique number assigned to each and every node within a cluster.
- *route_dst* — The ICF route number (at the remote node) associated with a specific route.
- *route_src* — The ICF route number (on the local node) associated with a route. An ICF route is the logical connection established between two nodes over a cluster interconnect.
- *servername* — The nodename of the node acting as a join server for the local (client) node that is attempting to join the cluster.
- *service* — Denotes the ICF registered service number. There are currently over 30 registered ICF services.

This first set of messages are "special" in that they deal with the CF driver basic initialization and de-initialization:

CF: cf_attach Error: invalid command. (#0425 bad_cmd)
CF: cf_attach Error: invalid instance. (#0425 *cf_instance instance*)
CF: cf_attach Error: phase 1 init failure. (*#reason_code*)
CF: cf_attach Error: phase 2 init failure. (*#reason_code*)
CF: cf_attach Error: unable to create cf minor.
CF: cf_detach Error: invalid instance. (#0425 *cf_instance instance*)

These messages are associated with a CF initialization failure. They should not occur unless the CF driver and/or other kernel components have somehow been damaged. Remove and then re-install the CF package. If the problem persists, contact field engineers.

11.6.1 Alphabetical list of messages

CF: carp_broadcast_version: Failed to announce version *cip_version*

This message will occur if CIP fails to initialize successfully, indicating some sort of mismatch between CIP and CF. This message should not occur unless the CF driver and/or other kernel components have somehow been damaged. Remove and then re-install the CF package. If the problem persists, contact field engineers.

CF: carp_event: bad nodeid (#0000 *nodenum*)

This message is generated by CIP when a bad nodenumber is received.

CF: cip: Failed to register ens EVENT_CIP

This message is generated when CIP initialization cannot register for the event EVENT_CIP.

CF: cip: Failed to register ens EVENT_NODE_LEFTCLUSTER

This message is generated when CIP initialization cannot register for the event EVENT_NODE_LEFTCLUSTER.

CF: cip: Failed to register icf channel ICF_SVC_CIP_CTL

This message is generated when CIP initialization cannot register with ICF for the service ICF_SVC_CIP_CTL.

CF: cip: message SYNC_CIP_VERSION is too short

This message is generated when CIP receives a garbled message.

CF: ens_nicf_input Error: unknown msg type received. (#0000 *msgtype*)

This message is generated by ENS when a garbled message is received from ICF. The message is dropped.

CF: Giving UP Mastering (Cluster already Running).

This message is generated when a node detects a join server and joins an existing cluster, rather than forming a new one. No action is necessary.

CF: Giving UP Mastering (some other Node has Higher ID).

This message is generated when a node volunteers to be a join server, but detects an eligible join server with a higher id. No action is necessary.

CF: Icf Error: (service err_type route_src route_dst). (#0000 *service err-type route_src route_dst*)

This message is generated when ICF detects an error. It is most common to see this message in missing heartbeat and route down situations.

CF: Join client *nodename* timed out. (#0000 nodenum)

This message is generated on a node acting as a join server, when the client node does not respond in time.

CF: Join Error: Invalid configuration: multiple devs on same LAN.

This message is generated when a node is attempting to join or form a cluster. Multiple network interconnects cannot be attached to the same LAN segment.

CF: Join Error: Invalid configuration: asymmetric cluster.

This message is generated when a node is joining a cluster that has a active node that does not support asymmetric clustering, and has configured an incompatible (asymmetric) set of cluster interconnects.

CF: Join postponed: received packets out of sequence from *servername*.

This message is generated when a node is attempting to join a cluster, but is having difficulty communicating with the node acting as the join server. Both nodes will attempt to restart the join process.

CF: Join postponed, server *servername* is busy.

This message is generated when a node is attempting to join a cluster, but the join server is busy with another client node. (Only one join may be active in/on the cluster at a time.) Another reason for this message to be generated is that the client node is currently in LEFTCLUSTER state. A node cannot re-join a cluster, unless its state is DOWN. (the cftool -k manual page.)

CF: Join timed out, server *servername* did not send node number: retrying.
CF: Join timed out, server *servername* did not send nsm map: retrying.
CF: Join timed out, server *servername* did not send welcome message.

These messages are generated when a node is attempting to join a cluster, but is having difficulty communicating with the node acting as the join server. The join client node will attempt to continue the join process.

CF: Local node is missing a route from node: *nodename*
CF: missing route on local device: *devicename*

These messages are generated when an asymmetric join has occurred in a cluster, and the local node is missing a route to the new node. The nodename and devicename of the associated cluster interconnect are displayed, in case this is not the desired result.

CF: Local Node *nodename* Created Cluster *clustername*. (#0000 *nodenum*)

This message is generated when a node forms a new cluster.

CF: Local Node *nodename* Left Cluster *clustername*.

This message is generated when a node leaves a cluster.

CF: No join servers found.

This message is generated when a node cannot detect any nodes willing to act as join servers.

CF: Node *nodename* Joined Cluster *clustername*. (#0000 *nodenum*)

This message is generated when a node joins an existing cluster.

CF: Node *nodename* Left Cluster *clustername*. (#0000 *nodenum*)

This message is generated when a node leaves a cluster.

CF: Received out of sequence packets from join client: *nodename*

This message is generated when a node, acting as a join server, is having difficulty communicating with the client node. Both nodes will attempt to restart the join process.

CF: Starting Services.

This message is generated by CF as it is starting.

CF: Stopping Services.

This message is generated by CF as it is stopping.

CF: User level event memory overflow: Event dropped (#0000 *eventid*)

This message is generated when an ENS user event is received, but there is no memory for the event to be queued.

CF: *clustername*: *nodename* is Down. (#0000 *nodenum*)

This message is generated when a node has left the cluster in an orderly manner (i.e., cfconfig -u).

CF: *nodename* Error: local node has no route to node: join aborted.

This message is generated when a node is attempting to join a cluster, but detects that there is no route to one or more nodes that are already members of the cluster.

CF: *nodename* Error: no echo response from node: join aborted.

This message is generated when a node is attempting to join a cluster, but is having difficulty communicating with all the nodes in the cluster.

CF: *servername*: busy: cluster join in progress: retrying
CF: *servername*: busy: local node not DOWN: retrying
CF: *servername*: busy mastering: retrying
CF: *servername*: busy serving another client: retrying
CF: *servername*: local node's status is UP: retrying
CF: *servername*: new node number not available: join aborted

These messages are generated when a node is attempting to join a cluster, but the join server is busy with another client node. (Only one join may be active in/on the cluster at a time.) Another reason for this message to be generated is that the client node is currently in LEFTCLUSTER state. A node cannot re-join a cluster, unless its state is DOWN. (the cftool -k manual page.)

CF (TRACE): cip: Announcing version *cip_version*

This message is generated when a CIP initialization is complete.

CF (TRACE): EnsEV: Shutdown

This message is generated when the ENS event-daemon shuts down.

CF (TRACE): EnsND: Shutdown

This message is generated when the ENS node_down-daemon shuts down.

CF (TRACE): Icf: Route UP: node src dest (#0000 *nodenum route_src route_dst*)

This message is generated when an ICF route is (re-)activated.

CF (TRACE): JoinServer: Stop

This message is generated when the join server mechanism is deactivated.

CF (TRACE): JoinServer: Startup

This message is generated when the join daemon is started.

CF (TRACE): JoinServer: ShutDown

This message is generated when an active join daemon shuts down.

CF (TRACE): Load: Complete

This message is generated when CF initialization is complete.

11.7 CF Reason Code table

Code	Reason	Service	Text
0401	REASON_SUCCESS		Operation was successful
	/* generic error codes */		
0401	REASON_NOERR	generic	Request not completed
0402	REASON_ALERTED	generic	Interrupted call
0403	REASON_TIMEOUT	generic	Timedout call
0404	REASON_NO_MEMORY	generic	Out of memory
0405	REASON_NO_SUCH_DEVICE	generic	No such device/resource
0406	REASON_DEVICE_BUSY	generic	Resource is busy
0407	REASON_INVALID_PARAMETER	generic	Invalid parameter
0408	REASON_UNSUCCESSFUL	generic	Unsuccessful
0409	REASON_ADDRESS_ALREADY_EXISTS	generic	Address already exists
040a	REASON_BAD_ADDRESS	generic	Bad memory address
040b	REASON_INSUFFICIENT_RESOURCES	generic	Insufficient resources
040c	REASON_BUFFER_OVERFLOW	generic	Buffer overflow
040d	REASON_INVALID_OWNER	generic	Invalid owner
040e	REASON_INVALID_HANDLE	generic	Invalid handle
040f	REASON_DUPNAME	generic	Duplicate name
0410	REASON_USAGE	generic	Usage
0411	REASON_NODATA	generic	No data
0412	REASON_NOT_INITIALIZED	generic	Driver not initialized
0413	REASON_UNLOADING	generic	Driver unloading
0414	REASON_REASSEMBLY_DOWN	generic	Sender died while sending data
0415	REASON_WENT_DOWN	generic	Destination node went down
0416	REASON_TRANSMIT_TIMEOUT	generic	Data transmission timeout
0417	REASON_BAD_PORT	generic	Bad destination port

Code	Reason	Service	Text
0418	REASON_BAD_DEST	generic	Bad destination
0419	REASON_YANK	generic	Message transmission flushed
041a	REASON_SVC_BUSY	generic	SVC has pending transmissions
041b	REASON_SVC_UNREGISTER	generic	SVC has been unregistered
041c	REASON_INVALID_VERSION	generic	Invalid version
041d	REASON_NOT_SUPPORTED	generic	Function not supported
041e	REASON_EPERM	generic	Not super-user
041f	REASON_ENOENT	generic	No such file or directory
0420	REASON_EINTR	generic	Interrupted system call
0421	REASON_EIO	generic	I/O error
0422	REASON_ENXIO	generic	No such device or address (I/O req)
0423	REASON_EACCES	generic	Permission denied
0424	REASON_EEXIST	generic	File exists
0425	REASON_DDI_FAILURE	generic	Error in DDI/DKI routine
0426	REASON_INVALID_NODENAME	generic	Invalid node name
0427	REASON_INVALID_NODENUMBER	generic	Invalid node number
0428	REASON_NODE_NOT_LEFTC	generic	Node is not in LEFTCLUSTER state
0429	REASON_CORRUPT_CONFIG	generic	Corrupt/invalid cluster config
042a	REASON_FLUSH	generic	Messages transmission flushed
042b	REASON_MAX_ENTRY	generic	Maximum entries reached
042c	REASON_NO_CONFIGURATION	generic	No configuration exists
	/* mrpc reasons */		
0801	REASON_MRPC_CLT_SVCUNAVAIL	mrpc	Service not registered on Client

Code	Reason	Service	Text
0802	REASON_MRPC_SRV_SVCUNAVAIL	mrpc	Service not registered on Server
0803	REASON_MRPC_CLT_PROCUNAVAIL	mrpc	Service Procedure not avail on Clt
0804	REASON_MRPC_SRV_PROCUNAVAIL	mrpc	Service Procedure not avail on Srv
0805	REASON_MRPC_INARGTOOLONG	mrpc	Input argument size too big
0806	REASON_MRPC_OUTARGTOOLONG	mrpc	Output argument size too big
0807	REASON_MRPC_RETARGOVERFLOW	mrpc	Return argument size overflow
0808	REASON_MRPC_VERSMISMATCH	mrpc	Version mismatch
0809	REASON_MRPC_ICF_FAILURE	mrpc	ICF send failed
080a	REASON_MRPC_INTR	mrpc	Interrupted RPC
080b	REASON_MRPC_RECURSIVE	mrpc	Illegal recursive call
080c	REASON_MRPC_SVC_EXIST	mrpc	Service already registered
	/* ens reasons */		
0c01	REASON_ENS_INFOTOOBIG	ens	Event information is too large
0c02	REASON_ENS_TOOSOON	ens	Attempt to post event before ens_init
0c03	REASON_ENS_NODEST	ens	Remote or local not specified in howto
0c04	REASON_ENS_DAEMONNOTIFY	ens	Invalid event posting by event daemon
0c05	REASON_ENS_NOICF	ens	Attempt to post remote before ICF config
0c06	REASON_ENS_OLDACKVERS	ens	Old version kernel has acked event
0c07	REASON_ENS_IMPLICITACK	ens	Event handler did not obtain ack handle
0c08	REASON_ENS_ACKNOTREQ	ens	Event acknowledgment not required

Code	Reason	Service	Text
0c09	REASON_ENS_NOTEVHANDLER	ens	Obtainer of ack handle not event handler
0c0a	REASON_ENS_NOACKHANDLE	ens	Cannot locate event ack handle
0c0b	REASON_ENS_MEMLIMIT	ens	User level ENS event memory limit overflow
0c0c	REASON_ENS_DUPREG	ens	Duplicate event registration
0c0d	REASON_ENS_REGNOTFOUND	ens	Event registration not found
0c0e	REASON_ENS_INFOTOOSMALL	ens	Event information size too small
0c0f	REASON_ENS_BADFAILNODE	ens	Node cannot post LEFTCLUSTER or NODE DOWN for itself
	/* nsm */		
1001	REASON_NSM_BADVERSION	nsm	Data structure version mismatch
1002	REASON_NSM_NONODES	nsm	No nodes have been specified
1003	REASON_NSM_TOOMANYNODES	nsm	Too many nodes have been specified
1004	REASON_NSM_BADNODEID	nsm	Node ID out of node name space range
1005	REASON_NSM_BADNETALEN	nsm	Invalid network address length
1006	REASON_NSM_ICFCREATE	nsm	Failure trying to create ICF node
1007	REASON_NSM_ICFDELETE	nsm	Failure trying to delete ICF node
1008	REASON_NSM_BADSTARTNODE	nsm	Invalid starting node specified
1009	REASON_NSM_BADINFOLEN	nsm	Invalid event information length

Code	Reason	Service	Text
100a	REASON_NSM_BADCNODEID	nsm	Control node out of name space range
100b	REASON_NSM_BADCNSTATUS	nsm	Control node status invalid
100c	REASON_NSM_BADANODEID	nsm	Invalid node ID for node being added
100d	REASON_NSM_ADDNODEUP	nsm	Node being added is already operational
100e	REASON_NSM_NONODE	nsm	Node does not exist in the node name space
100f	REASON_NSM_NODEFAILURE	nsm	A node has been declared dead
1010	REASON_NSM_NODETIMEOUT	nsm	Heartbeat timeout has expired for a node
1011	REASON_NSM_BADOUTSIZE	nsm	Invalid value for MRPC outsize
1012	REASON_NSM_BADINSIZE	nsm	Invalid value for MRPC insize
1013	REASON_NSM_BADNDNOTIFY	nsm	Failure to post NODE DOWN event
1014	REASON_NSM_VERSIONERR	nsm	nsetinfo versioning error
	/* mrpc */		
1401	REASON_ICF_MRPC_SZSM	icfmrpc	Output argument size too small
1402	REASON_ICF_MRPC_BADNDNUM	icfmrpc	Node does not exist
1403	REASON_ICF_MRPC_BADADDR	icfmrpc	mesh address does not exist
	/* user events */		
1801	REASON_UEV_ALREADYOPEN	uev	Process already has event device open
1802	REASON_UEV_TOOMANYEVENTS	uev	Too many user events initialized

Code	Reason	Service	Text
1803	REASON_UEV_BADHANDLE	uev	Invalid user event handle specified
1804	REASON_UEV_NOTOPEN	uev	Process does not have event device open
1805	REASON_UEV_REGISTERED	uev	Duplicate user event registration
	/* node group */		
1c01	REASON_NG_DEF_SYNTAX	ng	Bad definition syntax
1c02	REASON_NG_DUPNAME	ng	Name exists already
1c03	REASON_NG_EXIST	ng	Group does not exist
1c04	REASON_NG_ND_EXIST	ng	Node does not exist
1c05	REASON_NG_NAMELEN	ng	Too long a node name
1c06	REASON_NG_STATE	ng	Unknown parser state
1c07	REASON_NG_NODEINFO	ng	Failed to get up-node info
1c08	REASON_NG_ITER_STALE	ng	Iterator is stale
1c09	REASON_NG_ITER_NOSPACE	ng	Iterator pool exhausted
1c0a	REASON_NG_ITER_NOENT	ng	The end of iteration
1c0b	REASON_NG_MEMBER	ng	Node is not a group member
1c0c	REASON_NG_NOENT	ng	No node is up
1c0d	REASON_NG_UNPACK	ng	Failed to unpack definition
1c0e	REASON_NG_DUPDEF	ng	Identical group definition
	/* distributed mount services */		
2001	REASON_DMS_INVALIDCNG	dms	Invalid client node group
2002	REASON_DMS_MNTINUSE	dms	Mount in use
2003	REASON_DMS_DEVINUSE	dms	Device in use
2004	REASON_DMS_FSCKFAILED	dms	Failover fsck failed
2005	REASON_DMS_MNTFAILED	dms	Failover mount failed

Code	Reason	Service	Text
2006	REASON_DMS_MNTBUSY	dms	Mount is busy
2007	REASON_DMS_NOMNTPT	dms	No mount point specified
2008	REASON_DMS_NODBENT	dms	Specified mount point not found
2009	REASON_DMS_BADSTATE	dms	Server is up or failover in progress
200a	REASON_DMS_SUBMOUNT	dms	Specified mount point is CFS submount
200b	REASON_MAX_REASON_VAL	dms	Last reason
	/* join */		
2401	REASON_JOIN_FAILED	join	Node has failed to join cluster
2402	REASON_JOIN_DISABLED	join	Cluster join not started
2403	REASON_JOIN_SHUTDOWN	join	Join daemon shut down
	/* cfreg */		
2801	REASON_CFREG_STOPREQUESTED	cfreg	cfreg daemon stop requested
2802	REASON_CFREG_DUPDAEMON	cfreg	cfreg daemon already running
2803	REASON_CFREG_BADCONFIG	cfreg	Internal cfreg configuration error
2804	REASON_CFREG_NOENTRY	cfreg	Entry with specified key does not exist
2805	REASON_CFREG_COMMITTED	cfreg	Specified transaction committed
2806	REASON_CFREG_NOTOPEN	cfreg	Data file not open
2807	REASON_CFREG_CORRUPTFILE	cfreg	Data file format is corrupt
2808	REASON_CFREG_NSIERR	cfreg	Internal packaging error
2809	REASON_CFREG_INVALIDTRANS	cfreg	Specified transaction invalid

Code	Reason	Service	Text
280a	REASON_CFREG_ACTIVETRANS	cfreg	An active transaction exists
280b	REASON_CFREG_NOREQUESTS	cfreg	No daemon requests available
280c	REASON_CFREG_REQOVERFLOW	cfreg	Daemon request buffer overflow
280d	REASON_CFREG_NODAEMON	cfreg	cfreg daemon not present
280e	REASON_CFREG_BADREQUEST	cfreg	Unknown daemon request
280f	REASON_CFREG_REGBUSY	cfreg	Register is busy
2810	REASON_CFREG_REGOWNED	cfreg	Registry is owned
2811	REASON_CFREG_INVALIDUPDATE	cfreg	Invalid update
2812	REASON_CFREG_INVALIDKEY	cfreg	Invalid registry key
2813	REASON_CFREG_OVERFLOW	cfreg	Data or key buffer too small
2814	REASON_CFREG_TOOBIG	cfreg	Registry entry data too large
	/* cflog Message Catalogs */		
2c01	REASON_CFLOG_NOCAT	cflog	cflog could not open message catalog
	/* qsm Message Catalogs */		
3001	REASON_QSM_DUPMETHODNAME	qsm	Duplicate quorum method name
3002	REASON_QSM_TRYAGAIN	qsm	Need to try again later
3003	REASON_QSM_BUSY	qsm	Method has been registered already
3004	REASON_QSM_IDLE	qsm	Method has not been registered
3005	REASON_QSM_STOP	qsm	qsm stop requested
	/* sens */		

Code	Reason	Service	Text
3401	REASON_SENS_BADSEQ	sens	Invalid sequence number
3402	REASON_SENS_TOOSOON	sens	SENS not initialized
3403	REASON_SENS_DUPACK	sens	Duplicate registration for completion ack
3404	REASON_SENS_NOREG	sens	Registration does not exist
3405	REASON_SENS_BADMAP	sens	Node missing from node map
3406	REASON_SENS_NOUREG	sens	User event registration does not exist
3407	REASON_SENS_NOUEVENT	sens	Event not received
	/* CFRS */		
3801	REASON_CFRS_BADFCPSRCCONF	cfrs	cfcp not configured on source node
3802	REASON_CFRS_BADFCPDSTCONF	cfrs	cfcp not configured on destination node
3803	REASON_CFRS_BADEXECSRCCONF	cfrs	cfsh not configured on source node
3804	REASON_CFRS_BADEXECDSTCONF	cfrs	cfsh not configured on execution node
3805	REASON_CFRS_BADDSTPATH	cfrs	Invalid destination file path
3806	REASON_CFRS_DSTPATHTOOLONG	cfrs	Destination file path too long
3807	REASON_CFRS_SRCACCESSERR	cfrs	Cannot access source file
3808	REASON_CFRS_SRCNOTREG	cfrs	Source file is not regular file
3809	REASON_CFRS_SRCREADERR	cfrs	Source file read error
380a	REASON_CFRS_NOCMD	cfrs	No command string specified
380b	REASON_CFRS_CMDTOOLONG	cfrs	Command string too long

Code	Reason	Service	Text
380c	REASON_CFRS_OUTPUTWRTERR	cfrs	Command output write error
380d	REASON_CFRS_NSIERROR	cfrs	Internal CFRS NSI error
380e	REASON_CFRS_DSTABORTEXEC	cfrs	Execution aborted on execution node
380f	REASON_CFRS_INVALIDIOCTL	cfrs	Invalid ioctl call
3810	REASON_CFRS_BADDSTNODE	cfrs	Destination node not in cluster
3811	REASON_CFRS_BADROPHANDLE	cfrs	Bad remote operation handle
3812	REASON_CFRS_SRCEXECABORTED	cfrs	Remote exec aborted on source node
3813	REASON_CFRS_RESPOUTTOOSMALL	cfrs	Response output buffer too small
3814	REASON_CFRS_MRPCOUTSIZE	cfrs	Unexpected MRPC outsize error
3815	REASON_CFRS_DSTNODELEFT	cfrs	Destination node has left the cluster
3816	REASON_CFRS_DSTDAEMONDOWN	cfrs	cfregd on destination node down
3817	REASON_CFRS_DSTSTATERR	cfrs	Failure to stat dst file
3818	REASON_CFRS_DSTNOTREG	cfrs	Existing dstpath not regular file
3819	REASON_CFRS_DSTTMPOPENERR	cfrs	Cannot open tmp file on dst node
381a	REASON_CFRS_DSTTMPCHOWNERR	cfrs	Cannot chown tmp file on dst node
381b	REASON_CFRS_DSTTMPCHMODERR	cfrs	Cannot chmod tmp file on dst node
381c	REASON_CFRS_DSTTMPWRITEERR	cfrs	tmp file write error on dst node
381d	REASON_CFRS_DSTTMPCLOSEERR	cfrs	tmp file close error on dst node
381e	REASON_CFRS_DSTRENAMEERR	cfrs	Failed to rename existing dstpath

Code	Reason	Service	Text
381f	REASON_CFRS_TMPRENAMEERR	cfrs	Failed to tmp file to dstpath
3820	REASON_CFRS_DUPIFC	cfrs	Duplicate remote operation handle error
3821	REASON_CFRS_STALESUBFCREQ	cfrs	Stale remote operation handle error
3822	REASON_CFRS_BADSPAWN	cfrs	Failure to spawn exec cmd on dstnode
	/* CFSF */		
4001	REASON_CFSF_PENDING	cfsf	Invalid node down request with pending ICF failure
4002	REASON_MAX_REASON_VAL		Last reason

11.8 Error messages for different systems

For most Linux versions, to the file /usr/include/sys/errno.h for the meaning of an ERRNO for a particular system.

11.9 Linux ERRNO table

No	Name	Description
1	EPERM	**Operation not permitted / not super-user** Typically this error indicates an attempt to modify a file in some way forbidden except to its owner or the super-user. It is also returned for attempts by ordinary users to do things allowed only to the super-user.
2	ENOENT	**No such file or directory** A file name is specified and the file should exist but doesn't, or one of the directories in a path name does not exist.
3	ESRCH	**No such process, LWP, or thread** No process can be found in the system that corresponds to the specified PID, LWPID_t, or thread_t.

No	Name	Description
4	EINTR	**Interrupted system call** An asynchronous signal (such as interrupt or quit), which the user has elected to catch, occurred during a system service function. If execution is resumed after processing the signal, it will appear as if the interrupted function call returned this error condition. In a multi-threaded application, EINTR may be returned whenever another thread or LWP calls fork.(2)
5	EIO	**I/O error** Some physical I/O error has occurred. This error may in some cases occur on a call following the one to which it actually applies.
6	ENXIO	**No such device or address** I/O on a special file s to a sub-device which does not exist, or exists beyond the limit of the device. It may also occur when, for example, a tape drive is not on-line or no disk pack is loaded on a drive.
7	E2BIG	**Arg list too long** An argument list longer than ARG_MAX bytes is presented to a member of the exec family of functions (exec(2)). The argument list limit is the sum of the size of the argument list plus the size of the environment's exported shell variables.
8	ENOEXEC	**Exec format error** A request is made to execute a file which, although it has the appropriate permissions, does not start with a valid format (a.out(4)).
9	EBADF	**Bad file number** Either a file descriptor s to no open file, or a read(2) (respectively, write(2)) request is made to a file that is open only for writing (respectively, reading).
10	ECHILD	**No child processes** A wait(2) function was executed by a process that had no existing or unwaited-for child processes.

No	Name	Description
11	EAGAIN	**Try again / no more processes or no more LWPs** For example, the fork(2) function failed because the system's process table is full or the user is not allowed to create any more processes, or a call failed because of insufficient memory or swap space.
12	ENOMEM	**Out of memory / not enough space** During execution of brk() or sbrk() (brk(2)), or one of the exec family of functions, a program asks for more space than the system is able to supply. This is not a temporary condition; the maximum size is a system parameter. On some architectures, the error may also occur if the arrangement of text, data, and stack segments requires too many segmentation registers, or if there is not enough swap space during the fork(2) function. If this error occurs on a resource associated with Remote File Sharing (RFS), it indicates a memory depletion which may be temporary, dependent on system activity at the time the call was invoked.
13	EACCES	**Permission denied** An attempt was made to access a file in a way forbidden by the protection system.
14	EFAULT	**Bad address** The system encountered a hardware fault in attempting to use an argument of a routine. For example, errno potentially may be set to EFAULT any time a routine that takes a pointer argument is passed an invalid address, if the system can detect the condition. Because systems will differ in their ability to reliably detect a bad address, on some implementations passing a bad address to a routine will result in undefined behavior.
15	ENOTBLK	**Block device required** A non-block device or file was mentioned where a block device was required (for example, in a call to the mount(2) function).

No	Name	Description
16	EBUSY	**Device or resource busy** An attempt was made to mount a device that was already mounted or an attempt was made to unmount a device on which there is an active file (open file, current directory, mounted-on file, active text segment). It will also occur if an attempt is made to enable accounting when it is already enabled. The device or resource is currently unavailable. EBUSY is also used by mutexes, semaphores, condition variables, and read-write locks, to indicate that a lock is held, and by the processor control function P_ONLINE.
17	EEXIST	**File exists** An existing file was mentioned in an inappropriate context (for example, call to the link(2) function).
18	EXDEV	**Cross-device link** A hard link to a file on another device was attempted.
19	ENODEV	**No such device** An attempt was made to apply an inappropriate operation to a device (for example, read a write-only device.
20	ENOTDIR	**Not a directory** A non-directory was specified where a directory is required (for example, in a path prefix or as an argument to the chdir(2) function).
21	EISDIR	**Is a directory** An attempt was made to write on a directory.
22	EINVAL	**Invalid argument** An invalid argument was specified (for example, unmounting a non-mounted device), mentioning an undefined signal in a call to the signal(3C) or kill(2) function.
23	ENFILE	**File table overflow** The system file table is full (that is, SYS_OPEN files are open, and temporarily no more files can be opened).

No	Name	Description
24	EMFILE	**Too many open files** No process may have more than OPEN_MAX file descriptors open at a time.
25	ENOTTY	**Not a TTY - inappropriate ioctl for device** A call was made to the ioctl(2) function specifying a file that is not a special character device.
26	ETXTBSY	**Text file busy (obsolete)** An attempt was made to execute a pure-procedure program that is currently open for writing. Also an attempt to open for writing or to remove a pure-procedure program that is being executed.
27	EFBIG	**File too large** The size of the file exceeded the limit specified by resource RLIMIT_FSIZE; the file size exceeds the maximum supported by the file system; or the file size exceeds the offset maximum of the file descriptor.
28	ENOSPC	**No space left on device** While writing an ordinary file or creating a directory entry, there is no free space left on the device. In the fcntl(2) function, the setting or removing of record locks on a file cannot be accomplished because there are no more record entries left on the system.
29	ESPIPE	**Illegal seek** A call to the lseek(2) function was issued to a pipe.
30	EROFS	**Read-only file system** An attempt to modify a file or directory was made on a device mounted read-only.
31	EMLINK	**Too many links** An attempt to make more than the maximum number of links, LINK_MAX, to a file.
32	EPIPE	**Broken pipe** A write on a pipe for which there is no process to read the data. This condition normally generates a signal; the error is returned if the signal is ignored.
33	EDOM	**Math argument out of domain of function** The argument of a function in the math package (3M) is out of the domain of the function.

No	Name	Description
34	ERANGE	**Math result not representable** The value of a function in the math package (3M) is not representable within node precision.
42	ENOMSG	**No message of desired type** An attempt was made to receive a message of a type that does not exist on the specified message queue (msgrcv(2)).
43	EIDRM	**Identifier removed** This error is returned to processes that resume execution due to the removal of an identifier from the file system's name space (msgctl(2), semctl(2), and shmctl(2)).
44	ECHRNG	Channel number out of range
45	EL2NSYNC	Level 2 not synchronized
46	EL3HLT	Level 3 halted
47	EL3RST	Level 3 reset
48	ELNRNG	Link number out of range
49	EUNATCH	Protocol driver not attached
50	ENOCSI	No CSI structure available
51	EL2HLT	Level 2 halted
35	EDEADLK	**Resource deadlock condition** A deadlock situation was detected and avoided. This error pertains to file and record locking, and also applies to mutexes, semaphores, condition variables, and read-write locks.
37	ENOLCK	**No record locks available** There are no more locks available. The system lock table is full (fcntl(2)).
125	ECANCELED	**Operation canceled** The associated asynchronous operation was canceled before completion.
95	ENOTSUP	**Not supported** This version of the system does not support this feature. Future versions of the system may provide support.

No	Name	Description
122	EDQUOT	**Disc quota exceeded** A write(2) to an ordinary file, the creation of a directory or symbolic link, or the creation of a directory entry failed because the user's quota of disk blocks was exhausted, or the allocation of an inode for a newly created file failed because the user's quota of inodes was exhausted.
52	EBADE	Invalid exchange
53	EBADR	Invalid request descriptor
54	EXFULL	Exchange full
55	ENOANO	No anode
56	EBADRQC	Invalid request code
57	EBADSLT	Invalid slot
35	EDEADLOCK	File locking deadlock error
59	EBFONT	Bad font file format
-	EOWNERDEAD	Process died with the lock
-	ENOTRECOV-ERABLE	Lock is not recoverable
60	ENOSTR	**Device not a stream** A putmsg(2) or getmsg(2) call was attempted on a file descriptor that is not a STREAMS device.
61	ENODATA	**No data available** No data (for no-delay I/O).
62	ETIME	**Timer expired** The timer set for a STREAMS ioctl(2) call has expired. The cause of this error is device-specific and could indicate either a hardware or software failure, or perhaps a timeout value that is too short for the specific operation. The status of the ioctl() operation is indeterminate. This is also returned in the case of _lwp_cond_timedwait(2) or cond_timedwait(2).
63	ENOSR	**Out of stream resources** During a STREAMS open(2) call, either no STREAMS queues or no STREAMS head data structures were available. This is a temporary condition; one may recover from it if other processes release resources.

No	Name	Description
64	ENONET	**Node is not on the network** This error is Remote File Sharing (RFS) specific. It occurs when users try to advertise, unadvertise, mount, or unmount remote resources while the node has not done the proper startup to connect to the network.
65	ENOPKG	**Package not installed** This error occurs when users attempt to use a call from a package which has not been installed.
66	EREMOTE	**Object is remote** This error is RFS-specific. It occurs when users try to advertise a resource which is not on the local node, or try to mount/unmount a device (or pathname) that is on a remote node.
67	ENOLINK	**Link has been severed** This error is RFS-specific. It occurs when the link (virtual circuit) connecting to a remote node is gone.
68	EADV	**Advertise error** This error is RFS-specific. It occurs when users try to advertise a resource which has been advertised already, or try to stop RFS while there are resources still advertised, or try to force unmount a resource when it is still advertised.
69	ESRMNT	**Srmount error** This error is RFS-specific. It occurs when an attempt is made to stop RFS while resources are still mounted by remote nodes, or when a resource is readvertised with a client list that does not include a remote node that currently has the resource mounted.
70	ECOMM	**Communication error on send** This error is RFS-specific. It occurs when the current process is waiting for a message from a remote node, and the virtual circuit fails.
71	EPROTO	**Protocol error** Some protocol error occurred. This error is device-specific, but is generally not related to a hardware failure

No	Name	Description
-	ELOCKUN-MAPPED	Locked lock was unmapped
72	EMULTIHOP	**Multihop attempted** This error is RFS-specific. It occurs when users try to access remote resources which are not directly accessible.
73	EDOTDOT	**RFS specific error** This error is RFS-specific. A way for the server to tell the client that a process has transferred back from mount point.
74	EBADMSG	**Not a data message** /* trying to read unreadable message */ During a read(2), getmsg(2), or ioctl(2) I_RECVFD call to a STREAMS device, something has come to the head of the queue that can not be processed. That something depends on the call: read(): control information or passed file descriptor. getmsg():passed file descriptor. ioctl():control or data information.
36	ENAMETOOLONG	**File name too long** The length of the path argument exceeds PATH_MAX, or the length of a path component exceeds NAME_MAX while _POSIX_NO_TRUNC is in effect; see limits(4).
75	EOVERFLOW	Value too large for defined data type
76	ENOTUNIQ	**Name not unique on network** Given log name not unique.
77	EBADFD	**File descriptor in bad state** Either a file descriptor s to no open file or a read request was made to a file that is open only for writing.
79	ELIBACC	**Cannot access a needed shared library** Trying to exec an a.out that requires a static shared library and the static shared library does not exist or the user does not have permission to use it.

No	Name	Description
80	ELIBBAD	**Accessing a corrupted shared library** Trying to exec an a.out that requires a static shared library (to be linked in) and exec could not load the static shared library. The static shared library is probably corrupted.
81	ELIBSCN	**.lib section in a.out corrupted** Trying to exec an a.out that requires a static shared library (to be linked in) and there was erroneous data in the .lib section of the a.out. The .lib section tells exec what static shared libraries are needed. The a.out is probably corrupted.
82	ELIBMAX	**Attempting to link in too many shared libraries** Trying to exec an a.out that requires more static shared libraries than is allowed on the current configuration of the system. See NFS Administration Guide.
83	ELIBEXEC	**Cannot exec a shared library directly** Attempting to exec a shared library directly.
84	EILSEQ	**Illegal byte sequence** Illegal byte sequence when trying to handle multiple characters as a single character.
38	ENOSYS	**Function not implemented / operation not applicable** Unsupported file system operation.
40	ELOOP	**Symbolic link loop** Number of symbolic links encountered during path name traversal exceeds MAXSYMLINKS.
85	ERESTART	**Restartable system call** Interrupted system call should be restarted.
86	ESTRPIPE	**Streams pipe error (not externally visible)** If pipe/FIFO, don't sleep in stream head.
39	ENOTEMPTY	Directory not empty
87	EUSERS	**Too many users** Too many users (for UFS).
88	ENOTSOCK	Socket operation on non-socket
89	EDESTADDRREQ	**Destination address required** A required address was omitted from an operation on a transport endpoint. Destination address required.

No	Name	Description
90	EMSGSIZE	**Message too long** A message sent on a transport provider was larger than the internal message buffer or some other network limit.
91	EPROTOTYPE	**Protocol wrong type for socket** A protocol was specified that does not support the semantics of the socket type requested.
92	ENOPROTOOPT	**Protocol not available** A bad option or level was specified when getting or setting options for a protocol.
93	EPROTONO-SUPPORT	**Protocol not supported** The protocol has not been configured into the system or no implementation for it exists.
94	ESOCKTNO-SUPPORT	**Socket type not supported** The support for the socket type has not been configured into the system or no implementation for it exists.
95	EOPNOTSUPP	**Operation not supported on transport end-point** For example, trying to accept a connection on a datagram transport endpoint.
96	EPFNOSUPPORT	**Protocol family not supported** The protocol family has not been configured into the system or no implementation for it exists. Used for the Internet protocols.
97	EAFNOSUPPORT	**Address family not supported by protocol** An address incompatible with the requested protocol was used.
98	EADDRINUSE	**Address already in use** User attempted to use an address already in use, and the protocol does not allow this.
99	EADDRNO-TAVAIL	**Cannot assign requested address** Results from an attempt to create a transport end-point with an address not on the current node.
100	ENETDOWN	**Network is down** Operation encountered a dead network.

No	Name	Description
101	ENETUNREACH	**Network is unreachable** Operation was attempted to an unreachable network.
102	ENETRESET	**Network dropped connection because of reset** The node you were connected to crashed and rebooted.
103	ECONNABORTED	**Software caused connection abort** A connection abort was caused internal to your node.
104	ECONNRESET	**Connection reset by peer** A connection was forcibly closed by a peer. This normally results from a loss of the connection on the remote node due to a timeout or a reboot
105	ENOBUFS	**No buffer space available** An operation on a transport endpoint or pipe was not performed because the system lacked sufficient buffer space or because a queue was full.
106	EISCONN	**Transport endpoint is already connected** A connect request was made on an already connected transport endpoint; or, a sendto(3N) or sendmsg(3N) request on a connected transport endpoint specified a destination when already connected.
107	ENOTCONN	**Transport endpoint is not connected** A request to send or receive data was disallowed because the transport endpoint is not connected and (when sending a datagram) no address was supplied.
117	EUCLEAN	Structure needs cleaning
118	ENOTNAM	Not a XENIX named type file
119	ENAVAIL	No XENIX semaphores available
120	EISNAM	Is a named type file
121	EREMOTEIO	Remote I/O error
-	EINIT	**Define EINIT 141** /* Reserved for future use* /
-	EREMDEV	**Define EREMDEV 142** /* Error 142 */

No	Name	Description
108	ESHUTDOWN	**Cannot send after transport endpoint shutdown** A request to send data was disallowed because the transport endpoint has already been shut down.
109	ETOOMANYREFS	Too many ences: cannot splice
110	ETIMEDOUT	**Connection timed out** A connect(3N) or send(3N) request failed because the connected party did not properly respond after a period of time; or a write(2) or fsync(3C) request failed because a file is on an NFS file system mounted with the soft option.
111	ECONNREFUSED	**Connection refused** No connection could be made because the target node actively refused it. This usually results from trying to connect to a service that is inactive on the remote node.
112	EHOSTDOWN	**Node is down** A transport provider operation failed because the destination node was down.
113	EHOSTUNREACH	**No route to node** A transport provider operation was attempted to an unreachable node.
114	EALREADY	**Operation already in progress** An operation was attempted on a non-blocking object that already had an operation in progress.
115	EINPROGRESS	**Operation now in progress** An operation that takes a long time to complete (such as a connect()) was attempted on a non-blocking object).
116	ESTALE	Stale NFS file handle
11	EWOULDBLOCK	Operation would block
123	ENOMEDIUM	No medium found
124	EMEDIUMTYPE	Wrong medium type

11.10 Cluster Resource Management Facility Messages

This section explains the cluster resource management facility messages. The messages are classified into the following types:

- Messages displayed by the GUI
- Messages displayed upon execution of the CLI command

11.10.1 GUI (Cluster Resource Management Facility)

The Cluster Resource Management Facility Messages are explained below. Each description explains the meaning of each message and the procedure to be applied to correct the problems:

11.10.1.1Information Messages

0700 The resource database is not configured. Please configure it by using [Tool] - [Initial setup] menu.

Corrective action
Since the resource database has not been setup, the information is not displayed.
Choose *Tool -> Initial setup* menu from the CRM main menu, and then set up the resource database.

0702 The screen cannot be displayed from the main CRM window.

Corrective action
The machine administration screen is not displayed from the cluster resource management facility main window. After closing the message dialog box by clicking the *OK* button, select the *Machine* administration menu from the *MISC* category at the top menu of Web-Based Admin View.

0703 Do you want to start up *resource_name* (rid=*rid*)?

Corrective action
Click either of the following buttons, and then request to activate the selected resource:

- *Yes*: Requests to activate the resource.
- *No*: Does not request to activate the resource.

resource_name indicates a displayed resource name while *rid* indicates a resource ID.

0704 Do you want to stop *resource_name* (rid= *rid*)

Corrective action
Click either of the following buttons, and then request to deactivate the selected resource:

- *Yes*: Requests to deactivate the resource.
- *No*: Does not request to activate the resource.

resource_name indicates a displayed resource name while *rid* indicates a resource ID.

0705 Do you want to diagnose *resource_name* (rid=*rid*)?

Corrective action
Click either of the following buttons, and then run a patrol diagnosis for the selected resource:

- *Yes*: Runs a patrol diagnosis for the resource.
- *No*: Does not run a patrol diagnosis for the resource.

resource_name indicates a displayed resource name while *rid* indicates a resource ID.

0707 Do you want to begin the proc processing?

Corrective action

Click either of the following buttons:

- *Yes*: Executes the processing indicated by *proc*.
- *No*: Terminates the operation.

The type of processing varies with the processing name displayed in *proc*, as follows:

- Initial setup

  The resource database is set up.
- Automatic configure

  The device is registered to the database.

0708 *proc1* finished.

Corrective action

The processing shown in *proc1* has been completed.
Press the *OK* button to close the message dialog box.
The type of processing varies depending on the processing name displayed in *proc1*, as follows:

- Initial setup

  The resource database is set up.
- Automatic configure

  The device is registered to the database.

0709 The configuration change function cannot be used because it is being used by another task.

Corrective action

This processing cannot be executed because the configuration change function is being used for other operation.
After closing the message dialog box by clicking the *OK* button, exit from the processing execution screen of the same client or the other client (Web browser). Then, re-execute the operation.

0710 Processing cannot be ended because the following operation instruction is not completed.

Corrective action

The operation is being requested (the command is being transmitted) to the cluster resource management facility. After a lapse of one or two minutes, re-select the *File -> Exit* menu. If this error message appears again, close the browser once and then re-open it.

0711 Can't get information from the resource database.

Corrective action

The resource database cannot be obtained. Leave the operation undone keeping the screen displayed. Once the resource database has been obtained, the resource database information will be displayed.

0712 The resource database has already been configured.

Corrective action

The resource database has been already set up. Confirm whether the cluster resource management facility has been activated. If it has not been activated, start the cluster resource management facility. If it has been activated, leave the operation undone as is. Click the *OK* button to close the message dialog box.

0713 The node which completed the settings of resource database exists.

Corrective action

A node in which the resource database is already set up is available.
After closing the message dialog box by clicking the *OK* button, initialize the resource database. Then, re-execute the initial setup.

11.10.1.2Error Messages

0760 A requested operation failed. (Error Details)

Corrective action

The operation instruction to be executed has failed.
An error message of the cluster resource management facility will be displayed in Error Details. Take corrective action according to the instructions in the message.
Click the *OK* button to close the message dialog box.

0761 An internal contradiction occurred in the main CRM window. (Error Details)

Corrective action

Either of the following symptoms might occur:

- The Java execution environment in the Web browser failed to load the Java class files that constitute the cluster resource management facility, due to the load on the network.
- A program inconsistency was detected in the CRM main window.

  After closing the message dialog box by clicking the *OK* button, restart Web-Based Admin View. Then, restart the browser and re-execute the operation. For information on how to restart Web-Based Admin View, see "Restart" in the "PRIME-CLUSTER *Web-Based Admin View Operation Guide*."

If this message is displayed, collect the Java console, a hard copy of the error dialog box, and debugging information, and then contact your field engineers.

0763 The operation cannot be executed because the resource database is not configured on all nodes, or all nodes are not communicating with Web-Based Admin View.

Corrective action

The following symptoms might occur:

- The cluster resource management facility has not yet been activated because all nodes are just been started.
- None of the nodes are connected to Web-Based Admin View.

As the cluster resource management facility is being started, click the *OK* button to close the message dialog box, wait a short while, and then re-execute the operation. If the same symptom occurs, confirm whether the node power is on and whether the cluster resource management facility is running normally on the nodes. Correct the problem if necessary, and restart Web-Based Admin View on the node on which the operation is to be executed. For information on how to start Web-Based Admin View, see "Restart" in the "PRIMECLUSTER *Web-Based Admin View Operation Guide*." If this error message is displayed again even after applying the above corrective action, collect the Java console, a hard copy of the erroneous dialog box, and debugging information, and then contact your field engineers.
Click the *OK* button to close the message dialog box.

0764 An I/O error occurred.

Corrective action

Confirm whether the symptom conforms to any of the following descriptions:

a. While the CRM main window was displayed, the operation to shut down the node was performed repeatedly.

b. The operation has been requested from the CRM main window.

c. A node or client machine is loaded, or the network is loaded.

d. While the CRM main window was displayed, the CRM main window was initialized.

If any operation instruction is being requested, close the message dialog box by clicking the *OK* button, wait a short while (3 to 5 minutes), and then re-execute the operation request.
If no operation is being requested, close the message dialog box by clicking the *OK* button and then restart the browser.
If the above action fails to solve the problem, restart Web-Based Admin View on all nodes.
For information on how to restart Web-Based Admin View, see "Restart" in the "PRIMECLUSTER *Web-Based Admin View Operation Guide*."

0765 Communication with the management server failed.

Corrective action

The management server might have been disconnected because of the events to disconnect the network such as rebooting, shutdown, emergency shutdown, or LAN cable disconnection. Otherwise, restarting of Web-Based Admin View is considered the cause.
After closing the message dialog box by clicking the *OK* button, restart the browser and then display the screen again.
If this error continues to occur, collect the Java console, a hard copy of the erroneous dialog box, and the debugging information, and then contact your field engineers.

0766 The command terminated abnormally. (Error Details)

Corrective action

A message related to the cluster resource management facility is displayed in Error Details. Take corrective action according to the instructions in the message.
Click the *OK* button, and then close the message dialog box.

0767 Command execution failed.

Corrective action

The command that was being executed did not terminate within the specified time.
Collect the Java console, a hard copy of the error dialog box, and the debugging information, and then contact your field engineers.
Click the *OK* button, and then end the message dialog box.

0768 The processing for the proc1 cannot finish normally.

Corrective action

After you close the message dialog box by clicking the *OK* button, wait a short while and then re-execute the operation.
If this error message continues to be displayed, collect the Java console, a hard copy of the error dialog box, and the debugging information, and then contact your field engineers.
The processing to be executed varies with the processing name displayed in *proc1*.

- Initial setup

  The resource database is set up.

- Automatic configure

  The device is registered to the resource database.

0769 The processing was aborted because it could not be done on all nodes. (Error Details)

Corrective action

The command cannot be executed for the node that is displayed in Error Details.
After restarting Web-Based Admin View on that node, wait a short while and then re-execute. For details on restarting Web-Based Admin View, see "Restart" in the "PRIMECLUSTER *Web-Based Admin View Operation Guide*."
If the same message continues to appear, reboot that node.
If all of above corrective actions fail, collect the Java console, a hard copy of the erroneous dialog box, and the debugging information, and then contact your field engineers.
Click the *OK* button to close the message dialog box.

0773 The initial setup of the resource database failed. (Error Details)

Corrective action

The symptom can be classified into one of the following. Take corrective action according to the symptom.

- The information on Web-Based Admin View has not yet been obtained to enable setup of the cluster resource management facility on the node that is displayed in Error Details.

  Corrective action

  After closing the message dialog box by clicking the *OK* button, restart Web-Based Admin View on the node that is displayed in Error Details and then re-execute the operation instruction. For information on how to restart Web-Based Admin View, see "Restart" in the "PRIMECLUSTER *Web-Based Admin View Operation Guide*."

- As CF has not yet been configured, the resource database cannot be set up.

  Corrective action

  After closing the message dialog box by clicking the *OK* button, configure the CF and then re-execute the operation request.

For other than the above, or if the same error continues to occur, collect the Java console, a hard copy of the erroneous dialog box, and the debugging information, and then contact your field engineers.

0774 `Initial setup failed: the resource database could not be initialized.`

Corrective action

The startup processing of the initial setup failed.
Collect the Java console, a hard copy of the erroneous dialog box, and the debugging information.
After collecting the hard copy, click the *OK* button to close the message dialog box.

0775 `CF is not running, or CF is not configured.`

Corrective action

Since CF is not configured or CF is stopped on any one of the nodes, the initial setup cannot be executed for the resource database. After you select the *cf* tab and display the CF main window, confirm the state of CF and then configure or start CF.

11.10.2 Messages displayed by CLI command

This section explains the cluster resource management facility messages. The message format is described below.

Italic indicates that the output content varies depending on the message.

FJSVcluster:*severity*:*program*:*message-number*:*message*

severity	Indicates the message severity level. On the message severity level, there are four types: Stop (HALT), Information (INFORMATION), Warning (WARNING), Error (ERROR). For details, to the table below.
program	Indicates the name of the Resource Database program that output this message.
message-number	Indicates the message number.
message	Indicates the message text.

Number	Message severity level	Meaning
0000-0999	Stop (HALT)	Message indicating an abnormal termination of the function in the Resource Database is output.
2000-3999	Information (INFORMATION)	Message providing notification of information on the Resource Database operation status is output.
4000-5999	Warning (WARNING)	Message providing notification of a minor error not leading to abnormal termination of the function in the Resource Database is output.
6000-7999	Error (ERROR)	Message providing notification of a major error leading to abnormal termination of the function in the Resource Database is output.

Table 9: Resource Database severity levels

11.10.2.1HALT messages

0100 Cluster configuration management facility terminated abnormally.

Corrective action
Correct the cause of abnormal termination, then restart the error-detected node.

Supplement
The cause of abnormal termination is indicated in the previous error message.

0101 Initialization of cluster configuration management facility terminated abnormally.

Corrective action
Correct the cause of abnormal termination, then restart the error-detected node.

Supplement
The cause of abnormal termination is indicated in the previous error message.

0102 A failure occurred in the server. It will be terminated.

Corrective action
Follow the corrective action of the error message that was displayed right before this 0102 message

11.10.2.2Information messages

2100 The resource data base has already been set. (detail:*code1-code2*)
2200 Cluster configuration management facility initialization started.
2201 Cluster configuration management facility initialization completed.
2202 Cluster configuration management facility exit processing started.
2203 Cluster configuration management facility exit processing completed.
2204 Cluster event control facility started.
2205 Cluster event control facility stopped.
3200 Cluster resource management facility initialization started.
3201 Cluster resource management facility initialization completed.
3202 Cluster resource management facility exit processing completed.
3203 Resource activation processing started.
3204 Resource activation processing completed.
3205 Resource deactivation processing started.
3206 Resource deactivation processing completed.

11.10.2.3Warning messages

4250 The line switching unit cannot be found because FJSVclswu is not installed.

Supplement

Devices other than the line switching unit register an automatic resource.

5200 There is a possibility that the resource controller does not start. (ident:*ident* command:*command*, ...)

Supplement

Notification of the completion of startup has not yet been posted from the resource controller. *indent* indicates a resource controller identifier while command indicates the startup script of the resource controller.

11.10.2.4Error messages

???? Message not found!!

Corrective action

The text of the message corresponding to the message number is not available. Copy this message and contact field engineers.

6000 An internal error occurred.(function:*function* detail:*code1-code2-code3-code4*)

Corrective action

An internal error occurred in the program.
Record this message and collect information for an investigation. Then, contact field engineers. Collect information required for trouble-shooting (to the Section "Collecting Troubleshooting Information"). *function, code1, code2, code3, code4* indicates information required for error investigation.

6001 Insufficient memory. (detail:*code1-code2*)

Corrective action

Memory resources are insufficient to operate the Resource Database. *code1*, *code2* indicates information required for error investigation. Record this message. Collect information required for troubleshooting (to the Section "Collecting Troubleshooting Information"). Review the estimating of memory resources.
If this error cannot be corrected by this operator response, contact field engineers.

6002 Insufficient disk or system resources. (detail:code1-code2)

Corrective action

This failure might be attributed to the following:

- The disk space is insufficient
- There are incorrect settings in the kernel parameter

Collect information required for troubleshooting (to the Section "Collecting Troubleshooting Information").
Check that there is enough free disk space required for PRIME-CLUSTER operation. If the disk space is insufficient, you need to reserve some free area and reboot the node. For the required disk space, to the PRIMECLUSTER *Installation Guide*.
If you still have this problem after going through the above instruction, confirm that the kernel parameter is correctly set. Modify the settings if necessary and reboot the node. Nevertheless, the above instructions are not helpful, contact field engineers. **code1** and **code2** indicate information required for troubleshooting.

6003 Error in option specification. (option:*option*)

Corrective action

Specify a correct option, and execute the command again.
option indicates an option.

6004 No system administrator authority.

Corrective action

Re-execute the processing with the system administrator authority.

6005 Insufficient shared memory. (detail:*code1*-code2)

Corrective action

Shared memory resources are insufficient for the Resource Database to operate.
Record this message. Collect information required for troubleshooting (refer to the Section "Collecting Troubleshooting Information").
Refer to the Section "Kernel parameters for Resource Database" to review the estimate of shared memory resources (kernel parameters), Reboot the nodes that have any kernel parameters that have been changed.
If this error cannot be corrected by this operator response, contact field engineers.
code1, *code2* indicates information required for error investigation.

6006 The required option option must be specified.

Corrective action
Specify the correct option, then re-execute the processing.
option indicates an option.

6007 One of the required options *option* must be specified.

Corrective action
Specify a correct option, and execute the command again.
option indicates an option.

6008 If option *option1* is specified, option *option2* is required.

Corrective action
If the option indicated by *option1* is specified, the option indicated by *option2* is required. Specify the correct option, then re-execute the processing.

6009 If option *option1* is specified, option *option2* cannot be specified.

Corrective action
If the option indicated by *option1* is specified, the option indicated by *option2* cannot be specified. Specify the correct option, then re-execute the processing.

6010 If any one of the options *option1* is specified, option *option2* cannot be specified.

Corrective action
If either option indicated by *option1* is specified, the option indicated by *option2* cannot be specified. Specify the correct option, then re-execute the processing.

6021 The *option* option(s) must be specified in the following order: *order*

Corrective action
Specify option *options* sequentially in the order of *order*. Then, retry execution.
option indicates those options that are specified in the wrong order, while *order* indicates the correct order of specification.

6025 The value of option *option* must be specified from *value1* to *value2*

Corrective action
Specify the value of the *option* in option within the range between *value1* and *value2*, and then re-execute.
option indicates the specified option while *value1*,*value2* indicate values.

6200 Cluster configuration management facility: configuration database mismatch. (name:*name* node:*node*)

Corrective action

Record this message and collect information for an investigation. Then, contact field engineers (refer to the Section "Collecting Troubleshooting Information").
Collect the investigation information in all nodes, then reactivate the faulty node.
name indicates a database name in which a mismatch occurred, while *node* indicates a node in which an error occurred.

6201 Cluster configuration management facility: internal error. (node:*node* code:*code*)

Corrective action

There might be an error in the system if the kernel parameter /etc/system (4) is not properly set up when the cluster was installed. Check if the setup is correct (refer to Section "Kernel parameters for Resource Database"). If incorrect, reset the value of /etc/system(4), and then restart the system.
If there's still any problem regardless of the fact that the value of /etc/system (4) is larger than the required by Resource Database, and the same value is shown when checked by a sysdef(1M) command, take down the message, collect information for investigation, and then contact field engineers (refer to the Section "Collecting Troubleshooting Information").
Collect the investigation information in all nodes, then reactivate the faulty node.
node indicates a node in which an error occurred while *code* indicates the code for the detailed processing performed for the error.

6202 Cluster event control facility: internal error. (detail:*code1-code2*)

Corrective action

Record this message and collect information for an investigation. Then, contact field engineers. Collect information required for trouble-shooting (refer to the Section "Collecting Troubleshooting Information").
code1, *code2* indicates information required for error investigation.

6203 Cluster configuration management facility: communication path disconnected.

Corrective action

Check the state of other nodes and path of a private LAN.

6204 Cluster configuration management facility has not been started.

Corrective action

Record this message and collect information for an investigation. Then, contact field engineers. Collect information required for troubleshooting (refer to the Section "Collecting Troubleshooting Information").

6206 Cluster configuration management facility: error in definitions used by *target* command.

Corrective action

Record this message and collect information for an investigation. Then, contact field engineers. Collect information required for troubleshooting (refer to the Section "Collecting Troubleshooting Information").
target indicates a command name.

6207 Cluster domain contains one or more inactive nodes.

Corrective action

Activate the node in the stopped state.

6208 Access denied (*target*).

Corrective action

Record this message and collect information for an investigation. Then, contact field engineers. Collect information required for troubleshooting (refer to the Section "Collecting Troubleshooting Information").
target indicates a command name.

6209 The specified file or cluster configuration database does not exist (*target*).

Corrective action

Record this message and collect information for an investigation. Then, contact field engineers. Collect information required for troubleshooting (refer to the Section "Collecting Troubleshooting Information").
target indicates a file name or a cluster configuration database name.

6210 The specified cluster configuration database is being used (*table*).

Corrective action

Record this message and collect information for an investigation. Then, contact field engineers. Collect information required for troubleshooting (refer to the Section "Collecting Troubleshooting Information").
table indicates a cluster configuration database name.

6211 A table with the same name exists (*table*).

Corrective action

Record this message and collect information for an investigation. Then, contact field engineers. Collect information required for trouble-shooting (refer to the Section"Collecting Troubleshooting Information").
table indicates a cluster configuration database name.

6212 The specified configuration change procedure is already registered (*proc*).

Corrective action

Record this message and collect information for an investigation. Then, contact field engineers. Collect information required for trouble-shooting (refer to the Section "Collecting Troubleshooting Information").
proc indicates a configuration change procedure name.

6213 The cluster configuration database contains duplicate information.

Corrective action

Record this message and collect information for an investigation. Then, contact field engineers. Collect information required for trouble-shooting (refer to the Section "Collecting Troubleshooting Information").

6214 Cluster configuration management facility: configuration database update terminated abnormally (*target*).

Corrective action

Record this message and collect information for an investigation. Then, contact field engineers. Collect information required for trouble-shooting (refer to the Section "Collecting Troubleshooting Information").
Collect the investigation information in all nodes, then reactivate all nodes.
target indicates a cluster configuration database name.

6215 Cannot exceed the maximum number of nodes.

Corrective action

Since a hot extension is required for an additional node that exceeds the maximum number of configuration nodes that is allowed with Resource Database, review the cluster system configuration so that the number of nodes becomes equal to or less than the maximum number of composing nodes.

6216 Cluster configuration management facility: configuration database mismatch occurred because another node ran out of memory. (name:*name* node:*node*)

Corrective action

Record this message and collect information for an investigation. Then, contact field engineers. Collect information required for trouble-shooting (refer to the Section "Collecting Troubleshooting Information").
After collecting data for all nodes, stop the node and start it again.
name indicates a database in which a mismatch occurred and *node* indicates a node for which a memory shortfall occurred.

6217 Cluster configuration management facility: configuration database mismatch occurred because another node ran out of disk or system resources. (name:*name* node:*node*)

Corrective action

Record this message and collect information for an investigation. Then, contact field engineers. Collect information required for trouble-shooting (refer to the Section "Collecting Troubleshooting Information").
Reexamine the estimate for the disk resources and system resources (kernel parameter) (refer to the Section "Kernel parameters for Resource Database"). When the kernel parameter is changed for a given node, restart that node. If this error cannot be corrected by this operator response, contact field engineers. After collecting data for all nodes, stop and then restart the nodes.
name indicates a database in which a mismatch occurred and *node* indicates the node in which insufficient disk resources or system resources occurred.

6218 An error occurred during distribution of file to the stopped node. (name:*name* node:*node* errno:*errno*)

Corrective action

File cannot be distributed to the stopped node from the erroneous node. Be sure to start the stopped node before the active node stops. It is not necessary to re-execute the command.
name indicates the file name that was distributed when an failure occurred, *node* indicates the node in which a failure occurred, and *errno* indicates the error number when a failure occurred.

6219 The cluster configuration management facility cannot recognize the activating node. (detail:*code1-code2*)

Corrective action

Confirm that there is no failures in Cluster Foundation (CF) or cluster interconnect. If a failure occurs in CF, take the corrective action of the CF message. If a failure occurs in cluster interconnect, check that NIC is connected to the network.
If you still have this problem after going through the above instruction, contact field engineers. Collect information required for troubleshooting (refer to the Section "Collecting Troubleshooting Information").
code1 and *code2* indicate information required for troubleshooting.

6220 The communication failed between nodes or processes in the cluster configuration management facility. (detail:*code1-code2*)

Corrective action

Confirm that there is no failures in cluster interconnect. If a failure occurs in cluster interconnect, check that NIC is connected to the network.
If you still have this problem after going through the above instruction, contact field engineers. Collect information required for troubleshooting (refer to the Section "Collecting Troubleshooting Information").
code1 and *code2* Indicate information required for troubleshooting.

6221 Invalid kernel parameter used by cluster configuration database. (detail:*code1-code2*)

Corrective action

The kernel parameter used for the Resource Database is not correctly set up. Modify the settings, referring to Section "Kernel parameters for Resource Database", and reboot the node.
If you still have this problem after going through the above instruction, contact field engineers.
code1 and *code2* indicate information required for troubleshooting.

6222 network service used by the cluster configuration management facility is not available.(detail:*code1-code2*)

Corrective action

Confirm the /etc/inet/services file is linked to the /etc/services file. If not, you need to create a symbolic link to the /etc/services file. When setup process is done, confirm the following network services are set up in the /etc/inet/services file. If any of the following should fail to be set up, you need to add the missing.

```
dcmcom 9331/tcp# FJSVcldbm package
dcmsync 9379/tcp# FJSVcldbm package
dcmlck 9378/tcp# FJSVcldbm package
dcmfcp 9377/tcp# FJSVcldbm package
dcmmst 9375/tcp# FJSVcldbm package
dcmevm 9376/tcp# FJSVcldbm package
```

If this process is successfully done, confirm that the services of the /etc/nsswitch.conf file are defined as services: files nisplus. If not, you need to define them and reboot the node.

```
services: files nisplus
```

If you still have this problem after going through the above instruction, contact field engineers. Collect information required for trouble-shooting (refer to the Section "Collecting Troubleshooting Information").
code1 and *code2* indicate information required for troubleshooting.

6223 A failure occurred in the specified command. (command: *command*, detail:*code1-code2*)

Corrective action

Confirm that you can run the program specified as an option of the clexec(1M) command.If you still have this problem after going through the above instruction, contact field engineers. Collect information required for troubleshooting (refer to the Section "Collecting Trouble-shooting Information").
code1 and *code2* indicate information required for troubleshooting.

6226 The kernel parameter setup is not sufficient to operate the cluster control facility. (detail:*code*)

Corrective action

The kernel parameter used for the Resource Database is not correctly setup. Modify the settings, referring to the Section "Kernel parameters for Resource Database", and reboot the node.
Then, execute the clinitreset(1M) command, reboot the node, and initialize the Resource Database again. Confirm that you can run the program specified as an option of the clexec(1M) command.
If you still have this problem after going through the above instruction, contact field engineers. Collect information required for trouble-shooting (refer to the Section "Collecting Troubleshooting Information").
code indicates a parameter type and its recommended value.

6250 Cannot run this command because FJSVclswu is not installed.

Corrective action

Install the FJSVclswu package before executing the command. Refer to the PRIMECLUSTER *Installation Guide* for further details.

6300 Failed in setting the resource data base (insufficient user authority).

Corrective action

No CIP is set up in the Cluster Foundation. Reset CIP, and execute again after rebooting all nodes. Refer to the Section "CF, CIP, and CIM configuration" for the setup method.
If you still have this problem after going through the above instruction, contact field engineers. Collect information required for trouble-shooting (refer to the Section "Collecting Troubleshooting Information").
code1 and *code2* represents information for investigation.

6301 The resource data base has already been set (insufficient user authority).

Corrective action

The setup for Resource Database is not necessary. If you need to reset the setup, execute the clinitreset(1M) command on all nodes, initialize the Resource Database, and then reboot all nodes. For details, refer to the manual of the clinitreset(1M) command. *code1* and *code2* represents information for investigation.

6302 Failed to create a backup of the resource database information. (detail:*code1-code2*)

Corrective action

The disk space might be insufficient. You need to reserve 1 MB or more of free disk space, and back up the Resource Database information again.
If you still have this problem after going through the above instruction, contact field engineers. Collect information required for trouble-shooting (refer to the Section "Collecting Troubleshooting Information").
code1 and code2 indicate information required for troubleshooting.

6303 Failed restoration of the resource database information.(detail:*code1-code2*)

Corrective action

The disk space might be insufficient. You need to reserve 1 MB or more of free disk space, and restore the Resource Database information again.
If you still have this problem after going through the above instruction, contact field engineers. Collect information required for trouble-shooting (refer to the Section "Collecting Troubleshooting Information").
code1 and code2 indicate information required for troubleshooting.

6600 Cannot manipulate the specified resource. (insufficient user authority)

Corrective action

Re-execute the specified resource with registered user authority.

6601 Cannot delete the specified resource. (resource: *resource* rid:*rid*)

Corrective action

Specify the resource correctly, and then re-execute it.
resource indicates the resource name of the specified resource. *rid* indicates the resource ID of the specified resource.

6602 The specified resource does not exist. (detail:*code1-code2*)

Corrective action

Specify the correct resource, then re-execute the processing.
code1, *code2* indicates information required for error investigation.

6603 The specified file does not exist.

Corrective action

Specify the correct file, then re-execute the processing.

6604 The specified resource class does not exist.

Corrective action
Specify the correct resource class, and then re-execute the processing.
A specifiable resource class is a file name itself that is under /etc/opt/FJSVcluster/classes. Confirm that there is no error in the character strings that have been specified as the resource class.

6606 Operation cannot be performed on the specified resource because the corresponding cluster service is not in the stopped state. (detail:*code1*-*code2*)

Corrective action
Stop the cluster service, then re-execute the processing.
code1, *code2* indicates information required for error investigation.

6607 The specified node cannot be found.

Corrective action
Specify the node correctly. Then, execute again.

6608 Operation disabled because the resource information of the specified resource is being updated. (detail:*code1*-*code2*)

Corrective action
Re-execute the processing.
code1, *code2* indicates information required for error investigation.

6611 The specified resource has already been registered. (detail:*code1*-*code2*)

Corrective action
If this message appears when the resource is registered, it indicates that the specified resource has been already registered. There is no need to register it again.
If this message appears when changing a display name, specify a display name that is not available because the specified display name has already been registered.
code1, *code2* indicates information required for error investigation.

6614 Cluster configuration management facility: internal error.(detail:*code1*-*code2*)

Corrective action
Record this message, and contact field engineers. Collect information required for troubleshooting (refer to the Section "Collecting Troubleshooting Information").
code1, *code2* indicates information required for error investigation.

6615 The cluster configuration management facility is not running. (detail:*code1-code2*)

Corrective action
Reactivate the Resource Database by restarting the node. If the message is redisplayed, record this message and collect related information for investigation. Then, contact field engineers. Collect information required for troubleshooting (refer to the Section "Collecting Troubleshooting Information").
code1, *code2* indicates information required for error investigation.

6616 Cluster configuration management facility: error in the communication routine. (detail:*code1-code2*)

Corrective action
Record this message, and contact field engineers. Collect information required for troubleshooting (refer to the Section "Collecting Troubleshooting Information").
code1, *code2* indicates information required for error investigation.

6653 Operation cannot be performed on the specified resource.

Corrective action
userApplication in which the specified resource is registered is not in the Deact state. You need to bring this UserApplication Deact.

6661 Cluster control is not running. (detail:*code*)

Corrective action
Confirm that the Resource Database is running by executing the clgettree(1) command. If not, reboot the node.
If you still have this problem after going through the above instruction, contact field engineers. Collect information required for troubleshooting (refer to the Section "Collecting Troubleshooting Information").
code indicates information required for troubleshooting.

6665 The directory was specified incorrectly.

Corrective action
Specify the correct directory.

6668 Cannot run this command in single-user mode.

Corrective action
Boot the node in multi-user mode.

6675 Cannot run this command because *product_name* has already been set up.

Corrective action
Cancel the setting of the Resource Database *product_name*. Refer to appropriate manual for *product_name*.

6680 The specified directory does not exist.

Corrective action
Specify the existent directory.

6900 Automatic resource registration processing terminated abnormally. (detail: *reason*)

Corrective action

There might be incorrect settings in the shared disk definition file that was specified by the -f option of the clautoconfig(1M) command. Check the following. For details about the shared disk definition file, refer to the "Register shared disk units" of "PRIMECLUSTER *Global Disk Services Configuration and Administration Guide.*"

reason indicates the command that was abnormally terminated or the returned value.

- The resource key name, the device name, and the node identifier name are specified in each line.
- The resource key name begins with shd.
- The device name begins with /dev/.
- The node that has the specified node identifier name exists. You can check by executing the clgettree(1) command.
- There is a conflict with RCVM. RCVM and clautoconfig both use the volume name VTOC. Therefore, if you use RCVM, you must set the /etc/opt/FJSVcluster/etc/targetdisk file or the /etc/opt/FJSVcluster/etc/exceptdisk file, not the diskinfo file. Set either of these files as follows:
  - targetdisk file. Set disk name to targetdisk file, excluding disk name defined in definition file of RCVM, for all nodes in the cluster as follows: 1) Log in to the system as a root user. 2) Set target disks of clautoconfig to the targetdisk file, for example the /etc/opt/FJSVcluster/etc/targetdisk appears as follows:

    c1t1d0
    c1t2d0
    c1t3d0

    From any node registered in the cluster execute clautoconfig -r
  - exceptdisk file. Set disk name defined in definition file of RCVM to exceptdisk file for all nodes registered in the cluster. 1) Log in to the system as a root user. Modify the shared disk definition file if necessary, and then execute the clautoconfig(1M) command.

Modify the shared disk definition file if necessary, and then execute the clautoconfig(1M) command.

6901 Automatic resource registration processing is aborted due to one or more of the stopping nodes in the cluster domain.

Corrective action

Start all nodes and perform automatic resource registration.

6902 Automatic resource registration processing is aborted due to cluster domain configuration manager not running.

Corrective action

Cancel the automatic resource registration processing since the configuration of Resource Database is not working. Take down this message and collect the information needed for an investigation. Then, contact field engineers (refer to the Section "Collecting Troubleshooting Information").
Failures may be recovered by restarting all nodes after collecting investigation information.

6903 Failed to create logical path. (*node dev1 dev2*)

Corrective action

Contact field engineers to confirm that a logical path can be created in the share disk unit.
If you still have this problem after going through the above instruction, contact field engineers. Collect information required for trouble-shooting (refer to the Section "Collecting Troubleshooting Information").
node indicates an identification name of the node where the logical path failed to be created. *dev1* indicates the logical path (mplb2048), and *dev2* indicates a tangible path (c1t0d0 and c2t0d0) corresponding to the logical path.

6904 Fail to register resource. (detail: *reason*)

Corrective action

Failed to register resource during the automatic registration processing. This might happen when the disk resource and system resource are not properly set up. Check the system setting of kernel parameter, disk size, etc.
If you still have this problem after going through the above instruction, contact field engineers. Collect information required for trouble-shooting (refer to the Section "Collecting Troubleshooting Information").
reason indicates the reason why a direction was invalidated.

6905 Automatic resource registration processing is aborted due to mismatch instance number of logical device between nodes.

Corrective action

This message appears when the logical path of the multi-path disk is created before registering the automatic resource.
If this message appears during registering the automatic resource after adding on disks and nodes, the registration command might fail to access the logical path of the multi-path disk and check the instance number. This happens in the following conditions:

- The same logical path name is created on multiple nodes
- This path cannot be accessed from all nodes

The PRIMECLUSTER automatic resource registration has a feature to provide a same environment to all applications. If the instance number (indicates 2048 of mplb2048) of the logical path in the same disk is different between nodes, this message appears, and the automatic resource registration process is aborted. You need to check the logical path of all nodes. Recreate the logical path if necessary. The instance number should be the same. Then, register the automatic resource again.
If the cause is the failure of accessing the logical path of the multi-path disk, there might be a failure in the disk, or the disk is disconnected to the node.
Take the corrective action and register the automatic resource again.
If you still have this problem after going through the above instruction, contact field engineers. Collect information required for troubleshooting (refer to the Section "Collecting Troubleshooting Information").

6906 Automatic resource registration processing is aborted due to mismatch setting of disk device path between nodes.

Corrective action

This failure might be due to one of the following incorrect settings:

- Among the nodes connected to the same shared disk, the package of the multi-path disk control is not installed on all nodes.
- The detection mode of the shared disk is different between nodes.
- The number of paths to the shared disk is different between nodes.

Take the corrective action and register the automatic resource again.
If you still have this problem after going through the above instruction, contact field engineers. Collect information required for trouble-shooting (refer to the Section "Collecting Troubleshooting Information").

6907 Automatic resource registration processing is aborted due to mismatch construction of disk device between nodes.

Corrective action

When the same shared disk was mistakenly connected to other cluster system, the volume label might have been overridden. Check the disk configuration. If there's no problem with the configuration, collect information required for troubleshooting (refer to the Section "Collecting Troubleshooting Information").

6910 It must be restart the specified node to execute automatic resource registration. (node: *node_name*...)

Corrective action

The nodes constituting the cluster system must be restarted. Restart the nodes constituting the cluster system. After that, perform the necessary resource registration again.
node_name indicates a node identifier for which a restart is necessary. If multiple nodes are displayed with *node_name*, these node identifiers are delimited with commas. If *node_name* is All, restart all the nodes constituting the cluster system.

6911 It must be matched device number information in all nodes of the cluster system executing automatic resource registration. (dev: *dev_name*...)

Corrective action

Take down this message, and contact field engineers. Field engineers will take care of matching transaction for the information on the disk device.
dev_name represents information for investigation.

7500 Cluster resource management facility: internal error. (function:*function* detail:*code1-code2*)

Corrective action

Record this message, and contact field engineers. Collect information required for troubleshooting (refer to the Section "Collecting Troubleshooting Information").
function, *code1*, *code2* indicates information required for error investigation.

7501 Cluster resource management facility: insufficient memory. (function:*function* detail:*code1*)

Corrective action

Check the memory resource allocation estimate. For the memory required by Resource Database, refer to the PRIMECLUSTER *Installation Guide*. If this error cannot be corrected by this operator response, record this message, and contact field engineers. Collect information required for troubleshooting (refer to the Section "Collecting Troubleshooting Information").
function, *code1* indicates information required for error investigation.

7502 Cluster resource management facility: insufficient disk or system resources. (function:*function* detail:*code1*)

Corrective action

Referring to Section "Kernel parameters for Resource Database", review the estimate of the disk resource and system resource (kernel parameter). If the kernel parameters have been changed, reboot the node for which the kernel parameters have been changed. If this error cannot be corrected by this operator response, record this message, and contact field engineers. Collect information required for troubleshooting (refer to the Section "Collecting Troubleshooting Information").
function, *code1* indicates information required for error investigation.

7503 The event cannot be notified because of an abnormal communication. (type:*type* rid:*rid* detail:c*ode1*)

Corrective action

Record this message, and contact field engineers. Collect information required for troubleshooting (refer to the Section "Collecting Troubleshooting Information").
After this event is generated, restart all the nodes within a cluster domain.
type, *rid* indicates event information and *code1* indicates information for investigation.

7504 The event notification is stopped because of an abnormal communication. (type:*type* rid:*rid* detail:*code1*)

Corrective action

Record this message, and contact field engineers. Collect information required for troubleshooting (refer to the Section "Collecting Troubleshooting Information").
After this event is generated, restart all the nodes within a cluster domain.
type, *rid* indicates event information and *code1* indicates information for investigation.

7505 The node (node) is stopped because event cannot be notified by abnormal communication. (type:*type* rid:*rid* detail:*code1*)

Corrective action

Record this message and collect information for an investigation. Then, contact field engineers. Start the stopped node in a single user mode to collect investigation information (refer to the Section "Collecting Troubleshooting Information").
node indicates the node identifier of the node to be stopped, type, *rid* the event information, and *code1* the information for investigation.

7506 The node (*node*) is forcibly stopped because event cannot be notified by abnormal communication. (type:*type* rid:*rid* detail:*code1*)

Corrective action

Record this message and collect information for an investigation. Then, contact field engineers. Start the forcibly stopped node in a single user mode to collect the investigation information (refer to the Section "Collecting Troubleshooting Information").
node indicates the node identifier of the node to be stopped, *type*, *rid* the event information, and *code1* the information for investigation.

7507 Resource activation processing cannot be executed because of an abnormal communication. (resource:resource rid:rid detail:*code1*)

Corrective action

Record this message and collect information for an investigation. Then, contact field engineers. For details about collecting investigation information (refer to the Section "Collecting Troubleshooting Information").
After this phenomena occurs, restart the node to which the resource (*resource*) belongs. *resource* indicates the resource name for which activation processing was disabled, *rid* the resource ID, and *code1* the information for investigation.

7508 Resource (*resource1* resource ID:*rid1*, ...) activation processing is stopped because of an abnormal communication.
(resource:*resource2* rid:*rid2* detail:*code1*)

Corrective action

Record this message and collect information for an investigation. Then, contact field engineers. For details about collecting investigation information (refer to the Section "Collecting Troubleshooting Information").
After this phenomena occurs, restart the node to which the resource (*resource2*) belongs.
resource2 indicates the resource name for which activation processing was not performed, *rid2* the resource ID, resource1 the resource name for which activation processing is not performed, *rid1* the resource ID, and *code1* the information for investigation.

7509 Resource deactivation processing cannot be executed because of an abnormal communication. (resource:*resource* rid:*rid* detail:*code1*)

Corrective action

Record this message and collect information for an investigation. Then, contact field engineers (refer to the Section "Collecting Troubleshooting Information").
After this phenomena occurs, restart the node to which the resource (*resource*) belongs.
resource indicates the resource name for which deactivation processing was not performed, *rid* the resource ID, and *code1* the information for investigation.

7510 Resource (*resource1* resource ID:*rid1*, ...) deactivation processing is aborted because of an abnormal communication. (resource:*resource2* rid:*rid2* detail:*code1*)

Corrective action

Record this message and collect information for an investigation. Then, contact field engineers (refer to the Section "Collecting Troubleshooting Information").
After this phenomena occurs, restart the node to which the resource (*resource2*) belongs.
resource2 indicates the resource name for which deactivation processing was not performed, *rid2* the resource ID, *resource1* the resource name for which deactivation processing is not performed, *rid1* the resource ID, and *code1* the information for investigation.

7511 An error occurred by the event processing of the resource controller. (type:*type* rid:*rid* pclass:*pclass* prid:*prid* detail:*code1*)

Corrective action

Record this message and collect information for an investigation. Then, contact field engineers (refer to the Section "Collecting Troubleshooting Information").
After this phenomena occurs, restart the node in which the message was displayed.
type, *rid* indicates the event information, *pclass*, *prid* indicates resource controller information, and *code1* the information for investigation.

7512 The event notification is stopped because an error occurred in the resource controller. (type:*type* rid:*rid* pclass:*pclass* prid:*prid* detail:*code1*)

Corrective action

Record this message and collect information for an investigation. Then, contact field engineers (refer to the Section "Collecting Troubleshooting Information").
After this phenomena occurs, restart the node in which the message was displayed.
type, *rid* indicates the event information, *pclass*, *prid* indicates resource controller information, and *code1* the information for investigation.

7513 The node(*node*) is stopped because an error occurred in the resource controller. (type:*type* rid:*rid* pclass:*pclass* prid:*prid* detail:*code1*)

Corrective action

Record this message and collect information for an investigation. Then, contact field engineers (refer to the Section "Collecting Troubleshooting Information").
Start up the stopped node in a single user mode to collect investigation information.
node indicates the node identifier of the node to be stopped, *type*,*rid* the event information, *pclass*, *prid* the resource controller information, and *code1* the information for investigation.

7514 The node (*node*) is forcibly stopped because an error occurred in the resource controller. (type:*type* rid:*rid* pclass:*pclass* prid:*prid* detail:*code1*)

Corrective action

Record this message and collect information for an investigation. Then, contact field engineers (refer to the Section "Collecting Troubleshooting Information").
Start up the forcibly stopped node in a single user mode to collect investigation information.
node indicates the node identifier of the node to be forcibly stopped, *type*, *rid* the event information, *pclass*, *prid* the resource controller information, and *code1* the information for investigation.

7515 An error occurred by the resource activation processing (resource:*resource* rid:*rid* detail:*code1*)

Corrective action

Record this message and collect information for an investigation. Then, contact field engineers (refer to the Section "Collecting Troubleshooting Information").
After this phenomena occurs, restart the node to which the resource (*resource*) belongs. An error occurs in the resource activation processing and activation of the resource (*resource*) cannot be performed.
resource indicates the resource name in which an error occurred in the activation processing, *rid* the resource ID, and *code1* the information for investigation.

7516 An error occurred by the resource deactivation processing. (resource:*resource* rid:*rid* detail:*code1*)

Corrective action

Record this message and collect information for an investigation. Then, contact field engineers (refer to the Section "Collecting Troubleshooting Information").
After this phenomena occurs, restart the node to which the resource (*resource*) belongs. An error occurs in the resource deactivation processing and deactivation of the resource (*resource*) cannot be performed.
resource indicates the resource name in which an error occurred in the activation processing, *rid* the resource ID, and *code1* the information for investigation.

7517 Resource (*resource1* resource ID:*rid1*, ...) activation processing is stopped because an error occurred by the resource activation processing. (resource:*resource2* rid:*rid2* detail:*code1*)

Corrective action

Record this message and collect information for an investigation. Then, contact field engineers (refer to the Section "Collecting Troubleshooting Information").
After this phenomena occurs, restart the node to which the resource (*resource2*) belongs.
Resource2 indicates the resource name in which an error occurred in the activation processing, *rid2* the resource ID, *resource1* the resource name in which activation processing is not performed, *rid1* the resource ID, and *code1* the information for investigation.

7518 Resource (*resource1* resource ID:*rid1*, ...) deactivation processing is aborted because an error occurred by the resource deactivation processing. (resource:*resource2* rid:*rid2* detail:*code1*)

Corrective action

Record this message and collect information for an investigation. Then, contact field engineers (refer to the Section "Collecting Troubleshooting Information").
After this phenomena occurs, restart the node to which the resource (*resource2*) belongs.
resource2 indicates the resource name in which deactivation processing was disabled, rid2 the resource ID, *resource1* the resource name in which deactivation processing is not performed, *rid1* the resource ID, and *code1* the information for investigation.

7519 Cluster resource management facility: error in exit processing. (node:*node* function:*function* detail:*code1*)

Corrective action

Record this message and collect information for an investigation. Then, contact field engineers (refer to the Section "Collecting Troubleshooting Information").
node indicates the node in which an error occurred and *function*, *code1* the information for investigation.

7520 The specified resource (resource ID:*rid*) does not exist or be not able to set the dependence relation.

Corrective action

Specify the correct resource, then re-execute the processing.
rid indicates a resource ID of the specified resource.

7521 The specified resource (class:*rclass* resource:*rname*) does not exist or be not able to set the dependence relation.

Corrective action
Specify the correct resource, then re-execute the processing.
rname indicates the specified resource name and *rclass* the class name.

7522 It is necessary to specify the resource which belongs to the same node.

Corrective action
The resource belonging to other node is specified. Specify a resource that belongs to the same node and re-execute it.

7535 An error occurred by the resource activation processing. The resource controller does not exist. (*resource* resource ID:*rid*)

Corrective action
As the resource controller is not available in the resource processing, resource (*resource*) activation was not performed.
Record this message and collect information for an investigation. Then, contact field engineers (refer to the Section "Collecting Troubleshooting Information").
resource indicates the resource name for which activation processing was disabled, and *rid* a resource ID.

7536 An error occurred by the resource deactivation processing. The resource controller does not exist. (*resource* resource ID:*rid*)

Corrective action
As the resource controller is not available in the resource deactivation processing, resource (*resource*) deactivation was not performed.
Record this message and collect information for an investigation. Then, contact field engineers (refer to the Section "Collecting Troubleshooting Information").
resource indicates the resource name for which deactivation processing could not be performed, and *rid* the resource ID.

7537 Command cannot be executed during resource activation processing.

Corrective action
After activation processing of the resource completes, re-execute it.
Resource activation processing completion can be confirmed with 3204 message that is displayed on the console of the node to which the resource belongs.

7538 Command cannot be executed during resource deactivation processing.

Corrective action

After deactivation processing of the resource completes, re-execute it. Resource deactivation processing completion can be confirmed with 3206 message that is displayed on the console of the node to which the resource belongs.

7539 Resource activation processing timed out. (code:*code* detail:*detail*)

Corrective action

Record this message and collect information for an investigation. Then, contact field engineers (refer to the Section "Collecting Troubleshooting Information").

7540 Resource deactivation processing timed out. (code:code detail:*detail*)

Corrective action

Record this message and collect information for an investigation. Then, contact field engineers (refer to the Section "Collecting Troubleshooting Information").

7541 Setting related to dependence failed.

Corrective action

After confirming the specified resource, re-execute it.

7542 Resource activation processing cannot be executed because node (*node*) is stopping.

Corrective action

As the node *node* to which the resource to be activated belongs is stopped, the resource activation processing cannot be performed. After starting up the node to which resource to be activated belongs, re-execute it again.
node indicates the node identifier of the node where the connection is broken.

7543 Resource deactivation processing cannot be executed because node (*node*) is stopping.

Corrective action
As the node *node* to which the resource to be deactivated belongs is stopped, the resource deactivation processing cannot be performed. After starting up the node to which resource to be deactivated belongs, re-execute it again.
node indicates the node identifier of the node where the connection is broken.

7545 Resource activation processing failed.

Corrective action
Refer to the measures in the error message displayed between activation processing start message (3203) and completion message (3204), which are displayed when this command is executed.

7546 Resource deactivation processing failed.

Corrective action
Refer to the measures in the error message displayed between deactivation processing start message (3205) and completion message (3206), which are displayed when this command is executed.

11.11 Shutdown Facility

(SMAWsf, 10, 2) : %s of %s failed, errno %d

Cause: Internal problem.

Action: Check if there are related error messages following.
If yes, take action from there.
Otherwise, call field engineers.

(SMAWsf, 10, 3) : Unknown command from sd_tool, command %d

Cause: Using illegal sdtool command line.

Action: Choose the correct argument when sdtool is invoked.

(SMAWsf, 10, 4) : Failed to open CLI response pipe for PID %d, errno %d

Cause: rcsd daemon could not open the pipe to response to sdtool.

Action: Call field engineers.

(SMAWsf, 10, 6) : Failed to create a signal handler for SIGCHLD

Cause: Internal problem.

Action: Call field engineers.

(SMAWsf, 10, 7) : The shutdown agent %s has exceeded its configured timeout, pid %d terminated

Cause: The shutdown agent does not return in 'timeout' seconds, which is configured in rcsd.cfg.

Action: If increasing timeout does not help, most likely shutdown agent does not work. check the shutdown agent log and call field engineers.

(SMAWsf, 10, 8) : A shutdown request has come in during a test cycle, test of %s pid %d terminated

Cause: sdtool -k was invoked while rcsd was running a shutdown agent testing.

Action: No harm. Just ignore it.

(SMAWsf, 10, 9) : A request to reconfigure came in during a shutdown cycle, this request was ignored

Cause: When rcsd is eliminating a node, reconfiguration (sdtool -r) is not allowed.

Action: Try again after the node elimination is done.

(SMAWsf, 10, 10) : Could not correctly read the rcsd.cfg file.

Cause: either rcsd.cfg file does not exist or the syntax in rcsd.log is not correct.

Action: Create rcsd.cfg file or fix the syntax.

(SMAWsf, 10, 11) : %s in file %s around line %d

Cause: The syntax is not correct in rcsd.log
Action: fix the syntax.

(SMAWsf, 10, 12) : A request to exit rcsd came in during a shutdown cycle, this request was ignored

Cause: When rcsd is eliminating a node, bringing the rcsd daemon (sdtool -e) is not allowed.

Action: Try again after the killing node elimination is done.

(SMAWsf, 10, 15) : SA %s to %s host %s failed

Cause: The shutdown agent failed to do initialization | testing | shutdown | un-initialization the node

Action: Check the shutdown agent log and call field engineers.

(SMAWsf, 10, 17) : Failed to open lock file

Cause: internal problem

Action: Call field engineers.

(SMAWsf, 10, 19) : Failed to unlink/create/open CLI Pipe

Cause: internal problem

Action: Call field engineers.

(SMAWsf, 10, 20) : Illegal catalog open parameter

Cause: internal problem.

Action: Call field engineers.

(SMAWsf 10, 30) : Pthread failed: %s : errcode %d %s

Cause: Internal problem. POSIX thread failed.

Action: Call field engineers.

(SMAWsf, 10, 31) : Pthread failed: %s : errcode %d %s

Cause: Internal problem. rcsd was restarted.

Action: Call field engineers.

(SMAWsf, 10, 34) : Host %S MA exec: %s failed, errno %d

Cause: Failed to execute monitor agent's API.

Action: Call field engineers.

(SMAWsf, 10, 36) : Failed to cancel %s, thread f %s %s of host %s

Cause: POSIX thread was not cancellable.

Action: Call field engineers.

(SMAWsf, 10, 38) : Host %s, MA %s, MAH//get/state() failed

Cause: failed to call monitor agent's API MAHostGetState ().

Action: Call field engineers.

(SMAWsf, 10, 101) : Malloc failed during %s

Cause: Not enough memory.

Action: Increase virtual memory size (ulimit -v) or increase system memory. Call field engineers if the problem still exists.

(SMAWsf, 30, 2) : Usage: sdtool {-d[on | off] | -s | -S | -r | -b | -c | -e | -k node-name | -w weight-factor | -n node-factor }

Cause: Illegal argument/command line usage.

Action: Use the correct argument

(SMAWsf, 30, 3) : unlink failed on RCSD response pipe %s, errno %d

Cause: Can not remove the old pipe file.

Action: Call field engineers.

(SMAWsf, 30, 4) : mkfifo failed on RCSD response pipe %s, errno %d

Cause: Could not create the pipe for rcsd.

Action: Call field engineers.

(SMAWsf, 30, 5) : open failed on RCSD response pipe %s, errno %d

Cause: Could not open the pipe for rcsd.

Action: Call field engineers.

(SMAWsf, 30, 6) : open failed on rcsdin pipe %s, errno %d

Cause: Could not open communication pipe from sdtool to rcsd.

Action: Call field engineers.

(SMAWsf, 30, 7) : write failed on rcsdin pipe %s, errno %d

Cause: Could not pass command from sdtool to rcsd.

Action: Call field engineers.

(SMAWsf, 30, 8) : select failed, errno %d

Cause: sdtool could not get information from rcsd.

Action: Call field engineers.

(SMAWsf, 30, 9) : read failed, errno %d

Cause: sdtool failed to read data from rcsd daemon.

Action: Call field engineers.

(SMAWsf, 30, 10) : RCSD returned an error for this command, error is %d

Cause: rcsd failed to execute the command from sdtool.

Action: Check if there are related error messages following.
If yes, take action from there. Otherwise, call field engineers.

(SMAWsf, 30, 12) : A shutdown is in progress for the machine %s, try again later

Cause: rcsd daemon is currently eliminating the machine. The current request is ignored.

Action: Try again later.

(SMAWsf, 30, 13) : The RCSD is not running

Cause: The command failed because rcsd daemon is not running.

Action: Start up rcsd daemon (sdtool -b) then try the command again.

(SMAWsf, 30, 14) : RCSD is exiting. Command is not allowed

Cause: rcsd daemon is in the stage of shutting down. The command is not allowed.

Action: Try the command after rcsd daemon is started up.

(SMAWsf, 30, 15) : Failed to get %s product information

Cause: Most likely the product is not installed properly.

Action: Reinstall the product.

(SMAWsf, 30, 16) : Illegal catlog open parameter

Cause: Failed to open log file.

Action: Call field engineers.

(SMAWsf, 30, 17) : Could not execlp (RCSD). Errno = %d

Cause: Most likely the rcsd binary does not exist.

Action: Reinstall the package.

(SMAWsf, 50, 3) : The SF-CF initialization failed, status %d

Cause: Most likely, CF is not configured and/or is not loaded.

Action: Configure and load CF.

(SMAWsf, 50, 4) : The SF-CF event processing failed, status %d

Cause: Internal problem.

Action: Call field engineers.

(SMAWsf, 50, 6) : The SF-CF has failed to locate host %s

Cause: The nodename in the rcsd.cfg is not a known CF name.

Action: Use the CF name (cftool -n) in rcsd.cfg.

(SMAWsf, 50, 9) : The SF-CF failed to declare %s down, status %d

Cause: Internal problem.

Action: Call field engineers.

(SMAWsf, 50, 11) : Failed to open CFSF device, reason (%d%s

Cause: Could not open CFSF device

Action: Call field engineers.

(SMAWsf, 50, 12) : h_cfsf_get_leftcluster() failed. reason (%d) %s

Cause: Failed to call cfsf_get_leftcluster. Action: Call field engineers.

(SMAWsf, 50, 13) : Node id %d ICF communication failure detected

Cause: CF layer has detected lost heartbeat

Action: rcsd will take action.

(SMAWsf, 50, 14) : Host %s ICF communications failure detected

Cause: rcsd was notified the node has lost heartbeat

Action: rcsd take action to eliminate the node.

(SMAWsf, 50, 20) : Failed to cancel thread of the %s-monitor

Cause: Failed to cancel thread

Action: Call field engineers.

(SMAWsf, 50, 21) : Failed to do %s, reason (%d)%s

Cause: Failed to call some internal functions.

Action: Call field engineers.

(SMAWsf, 50, 22) : Failed to get nodeid for host %s. reason (%d)%s

Cause: Not able to get the cluster node id for the node.

Action: Call field engineers.

11.12 CCBR messages

The CCBR Framework commands, cfbackup(1M) and cfrestore(1M), will generate error messages on stderr and warning messages in an error log file if one or more error conditions are detected. All Framework messages have a date and time prefix, optionally followed by the text WARNING: and the command name, and then followed by the error text. Layered-product plugin modules can also generate warning messages, error messages, or both.

11.12.1 cfbackup warning/error messages

11.12.1.1To stderr

- *date time* cfbackup: invalid option specified

  One or more invalid arguments were used with the cfbackup command. The command syntax is as follows:

cfbackup	[-test]	[-f] [*n*]
where	-test	can be used by plug-in developers. It will cause the $CCBROOT tree to remain after a successful run (it is usually deleted). Also, the backup/restore generation number will not be incremented.
	-f	specifies FORCE, which will always cause a compressed archive file to be created, even when 'fatal' errors have been detected.
	n	specifies the backup/restore generation to use.

- *date time* cfbackup: cmd must be run as root

  The cfbackup command must be executed by root (uid=0).

- *date time* cfbackup: ccbr files & directories must be accessible

  The cfbackup command must be able to access /opt/SMAW/ccbr, /opt/SMAW/ccbr/plugins, and /opt/SMAW/ccbr/ccbr.conf.

11.12.1.2To log file

- *date time* WARNING: cfbackup: specified generation *n* too small - using *p*

  The generation number specified on the cfbackup command is less than the value in /opt/SMAW/ccbr/ccbr.gen. The larger value will be used.

- *date time* cfbackup [FORCE] *n* [(TEST)] log started

  This message indicates that cfbackup is beginning processing.

- *date time nodename* not an active cluster node

  This informational message indicates that the node is not an active PRIME-CLUSTER node.

- *date time* no runnable plug-ins! cmd aborted.

  The cfbackup command cannot find executable scripts in the /opt/SMAW/ccbr/plugins directory.

- *date time* cfbackup n ended unsuccessfully

  This message indicates that the cfbackup command is ending with an error code of 2 or 3.

- *date time* validation failed in pluginname

  This error message indicates that the validation routine in one or more plugin modules has returned an error code of 2 or 3 to the cfbackup command.

- *date time* backup failed in pluginname

  This error message indicates that the backup routine in one or more plugin modules has returned an error code of 2 or 3 to the cfbackup command.

- *date time* archive file creation failed

  This error message indicates the cfbackup command cannot successfully create a tar archive file from the backup tree.

- *date time* archive file compression failed

  This error message indicates that the cfbackup command cannot create a compressed archive file (with compress).

- *date time* cfbackup n ended

  This message indicates that the cfbackup command has completed all processing. The highest return code value detected while processing will be used as the return/error code value.

11.12.2 cfrestore warning/error messages

11.12.2.1To stderr

- *date time* cfrestore: invalid option specified

  One or more invalid arguments were used with the cfrestore command. The command syntax is as follows:

cfrestore	[-test]	[-f] [p] [-y] [*n*]
where	-test	can be used by plug-in developers. It will cause the $CCBROOT tree to remain after a successful run (it is usually deleted). Also, the cpio step will restore all saved files to /tmp/ccbr instead of / --- this will give plug-in developers a chance to check results, before "doing it" for real...
	-f	specifies FORCE, which will always cause a archive file to be restored, even when 'fatal' errors have been detected.
	-p	specifies PASS, which allows cfrestore to use a cfrestore file-tree that has already been 'extracted' from a compressed archive.
	-y	specifies an automatic YES answer, whenever the cfrestore command requests a confirmation response.
	-M	force restore even if we are in multi-user mode
	n	specifies the backup/restore generation to use.

- *date time* cfrestore: cmd must be run as root

  The cfrestore command must be executed by root (uid=0).

- *date time* cfrestore: cmd must be run in single-user mode

  The cfrestore command must be executed while at runlevel 1 or S (single-user mode).

- *date time* cfrestore: ccbr files & directories must be accessible

  The cfrestore command must be able to access /opt/SMAW/ccbr, /opt/SMAW/ccbr/plugins, and /opt/SMAW/ccbr/ccbr.conf.

11.12.2.2To log file

- *date time* cfrestore [FORCE] [TREE] [YES] n [(TEST)] log started

  This message indicates that cfrestore is beginning processing.

- *date time* ERROR: nodename IS an active cluster node

  This cfrestore error message indicates that the node is an active PRIME-CLUSTER node, and that restoring cluster configuration information at this time may lead to severe errors, and is not recommended.

- *date time* cfrestore *n* ended unsuccessfully

  This message indicates that the cfrestore command is ending with an error code of 2 or 3.

- *date time* no runnable plug-ins! cmd aborted.

  The cfrestore command cannot find executable scripts in the /opt/SMAW/ccbr/plugins directory.

- *date time* unable to find selected archive file: *archivefile*

  This message indicates that the cfrestore command cannot locate the archive file at $CCBROOT.tar.Z (Solaris). The CCBROOT value is set using *nodename* and *generation number*.

- *date time* archive file uncompression failed

  This error message indicates that the cfrestore command cannot expand the compressed archive file (with uncompress).

- *date time* archive file extraction failed

  This error message indicates the cfrestore command cannot successfully recreate a backup tree from the tar archive file.

- *date time* archive file recompression failed

  This error message indicates that the cfrestore command cannot recreate the compressed archive file (with compress).

- *date time* warning: backup created with FORCE option

  This warning message indicates that cfbackup created this archive file with the FORCE option specified (usually used to force past an error condition). It is highly recommended that the error logfile in the backup archive be examined to make sure a restore of this data will be valid.

- *date time* plugin present at backup is missing for restore: *pluginname*

  This error message indicates that the named plugin module is missing from the /opt/SMAW/ccbr/plugins directory. This usually indicates that a PRIME-CLUSTER package has been uninstalled and not reinstalled, or that a newer or older package does not have the same named plugin(s).

- *date time* negative reply terminates processing

  This error message indicates that the reply to the question (asked by cfrestore), "Are you sure you want to continue (y/n) ?", was not answered with YES. Processing terminates unless the FORCE option has been specified.

- *date time* plugin validation failed

  This error message indicates that the validation routine of the identified plugin module has returned an error code of 2 or 3 to the cfrestore command. Validation will continue so that all plugin modules have a chance to identify problems.

- *date time* cpio copy for cfrestore failed

  This error message indicates that the automatic cpio restore of all file trees rooted in the "root" subdirectory of the backup tree failed in execution. The cpio command is executed in verbose mode, so that there will be some history of which files were restored. This error usually indicates a partial restore has occurred. This can be a significant problem, and may require manual intervention to repair/restore the modified files.

- *date time* NOTE: no root subdirectory for cpio copy step

  This warning message indicates that cfrestore did not find any files to automatically restore from the backup tree. This is usually an error, probably indicating a damaged backup archive.

- *date time* plugin restore failed

  This error message indicates that the restore routine of the identified plugin module has returned an error code of 2 or 3 to the cfrestore command. Only a small number of plugins will need to provide an active restore routine. Restore will continue so that all plugins have a chance to identify problems. Any problems at this time, after the automatic cpio restore, will need to be examined individually and fixed manually.

- *date time* cfrestore n ended

  This message indicates that the cfrestore command has completed all processing. The highest return code value detected while processing will be used as the return/error code value.

11.13 Monitoring Agent messages

11.13.1 INFORMATION message

3080 The MMB monitoring agent has been started.

3081 The MMB monitoring agent has been stopped.

3082 MMB has been recovered from the failure. (node:*nodename* mmb_ipaddress1:*mmb_ipaddress1* mmb_ipaddress2:*mmb_ipaddress2* node_ipaddress1:*node_ipaddress1* node_ipaddress2:*node_ipaddress2*)

3083 Monitoring another node has been started.

3084 Monitoring another node has been stopped.

3085 The MMB IP address or the Node IP address has been changed. (mmb_ipaddress1:*mmb_ipaddress1* mmb_ipaddress2:*mmb_ipaddress2* node_ipaddress1:*node_ipaddress1* node_ipaddress2:*node_ipaddress2*)

11.13.2 WARNING message

5021 An error has been detected in part of the transmission route to MMB. (node:*nodename* mmb_ipaddress1:*mmb_ipaddress1* mmb_ipaddress2:*mmb_ipaddress2* node_ipaddress1:*node_ipaddress1* node_ipaddress2:*node_ipaddress2*)

Corrective action

An error was detected in part of the transmission route to MMB. Check the following points:

- Whether the normal lamp of the port to which the HUB and the LAN cable are connected is on.
- Whether the MMB port connecter or the LAN cable from the HUB-side connector is disconnected.
- Whether the load to MMB is high.

If one of the above items is found to be the cause, MMB monitoring agent recovers automatically after the corrective action is taken. Automatic recovery takes up to 10 minutes. If the message 3082 appeared once but didn't appear 10 minutes later in the next check cycle, the first message can be ignored because the cause of the message appearance can be considered as temporal high load on the MMB. If the connection fails even after the above items are checked, contact field engineers because there may be a network failure or a hardware failure, such as in the MMB of the HUB.
If this action does not correct the problem, copy down this message, collect investigation information, and contact field engineers. For instructions on collecting investigation information, see Section "Collecting Troubleshooting Information".

11.13.3 ERROR message

6000 An internal error occurred. (function:*function* detail:*code1-code2-code3-code4*)

Corrective action
Copy down this message, collect investigation information, and contact field engineers. For instructions on collecting investigation information, see Section "Collecting Troubleshooting Information".

6003 Error in option specification. (option:*option*)

Corrective action
Specify the correct option, and then re-execute the process.

6004 No system administrator authority.

Corrective action
The user does not have the system administrator authority. Execute the operation with the system administrator authority.

6007 One of the required options (*options*) must be specified.

Corrective action
Specify one of the required options.

7030 CF is not running.

Corrective action
The CF is not operating. If the CF has not been configured, see Section "CF, CIP, and CIM configuration", and configure the CF. If the CF has already been configured, start the CF. For instructions on starting the CF, see Section "Starting and stopping CF".

7033 Cannot find the specified CF node name. (nodename:*nodename*)

Corrective action

The specified CF node name does not exist. Check the following point, and then re-execute the process:

1. Whether the specified CF node name is correct.
   Use cftool(8) and check whether the specified CF node name is correct. If an incorrect CF node name was specified, specify the correct CF node name.
2. Whether the CF of the specified node is operating.
   Use cftool(8), and check whether the CF is operating.
   If the CF is not operating, start the CF. For instructions on starting the CF, see Section "Starting and stopping CF".
   If this action does not correct the problem, copy down this message, collect investigation information, and contact field engineers. For instructions on collecting investigation information, see Section "Collecting Troubleshooting Information"

7210 An error was detected in MMB. (node:*nodename* mmb_ipaddress1:*mmb_ipaddress1* mmb_ipaddress2:*mmb_ipaddress2* node_ipaddress1:*node_ipaddress1* node_ipaddress2:*node_ipaddress2* status:*status* detail:*detail*)

Corrective action

A communication error occurred due to an error in the MMB of the node on which the message was displayed.
A hardware failure may have occurred. Check the following point:

- Whether the MMB is not operating properly.

Copy down this message, collect investigation information, and contact field engineers. For instructions on collecting investigation information, see Section "Collecting Troubleshooting Information".
After field engineers carry out the hardware recovery operation, the MMB monitoring agent function recovers automatically.

7211 The MMB monitoring agent has already been started.

Corrective action

The MMB monitoring agent function has already been started. If there is no need to restart the MMB monitoring agent function, no action is necessary. If the MMB monitoring agent function must be restarted, take the action described below.
To restart the MMB monitoring agent function, execute the following command on the node on which the message was output:

```
# /etc/opt/FJSVcluster/bin/clmmbmonctl restart
```

Then restart the shutdown facility (SF) of the node on which this message was output.

```
# /opt/SMAW/bin/sdtool -r
```

If this action does not correct the problem, copy down this message, collect investigation information, and contact field engineers. For instructions on collecting investigation information, see Section "Collecting Troubleshooting Information".

7212 The MMB information is not set. (nodename:*nodename*)

Corrective action

MMB information for the specified CF node name has not been registered. Execute the clmmbsetup -l command, and check the currently registered MMB information. Then, if necessary, use the shutdown configuration wizard or the clmmbsetup command to register the MMB information.
For details about the shutdown configuration wizard, see Section "Configuring the Shutdown Facility". For details about the clmmbsetup command, see clmmbsetup(8).
If this action does not correct the problem, copy down this message, collect investigation information, and contact field engineers. For instructions on collecting investigation information, see Section "Collecting Troubleshooting Information".

7213 An error has been detected in the transmission route to MMB. (node:*nodename* mmb_ipaddress1:*mmb_ipaddress1* mmb_ipaddress2:*mmb_ipaddress2* node_ipaddress1:*node_ipaddress1* node_ipaddress2:*node_ipaddress2*)

Corrective action

An error was detected in the transmission route to the MMB. Check the following points:

- Whether the normal lamp of the port to which the HUB and the LAN cable are connected is on.
- Whether the MMB port connecter or the LAN cable from the HUB-side connector is disconnected.
- Whether an incorrect IP address was specified for the MMB IP address or the local administration IP address.

If one of the above items is found to be the cause, MMB monitoring agent recovers automatically after the corrective action is taken. Automatic recovery takes up to 10 minutes.
If the connection fails even after the above items are checked, contact field engineers because there may be a network failure or a hardware failure, such as in the MMB of the HUB. If this action does not correct the problem, copy down this message, collect investigation information, and contact field engineers. For instructions on collecting investigation information, see Section "Collecting Troubleshooting Information".
After field engineers carry out the hardware recovery operation, the MMB monitoring agent function recovers automatically.

7214 The username or password to login to the MMB is incorrect.

Corrective action

You cannot login to the MMB.
Either the user name or password for logging in to the MMB is different from the value that was set to the MMB. Specify the settings for MMB monitoring agent and the shutdown facility again. For instructions on setting the MMB, see "Checking the Shutdown Agent Information" in the "PRIMECLUSTER *Installation and Administration Guide (for Linux)*".
If this action does not correct the problem, copy down this message, collect investigation information, and contact field engineers. For instructions on collecting investigation information, see Section "Collecting Troubleshooting Information".

7215 An error was detected in the MMB IP address or the Node IP address. (mmb_ipaddress1:*mmb_ipaddress1* mmb_ipaddress2:*mmb_ipaddress2* node_ipaddress1:*node_ipaddress1* node_ipaddress2:*node_ipaddress2*)

Corrective action

The MMB IP address of the node on which this message was output or the local administration IP address was changed. Immediately after this message is output, if the Test State of the shutdown facility (SF) is Test Failed, check the following point:

- Whether the change in the MMB IP address or the local administration IP address is correct.

If the above item is found to be the cause, take corrective action. Then execute the following command on all nodes, and restart the shutdown facility (SF):

```
# /opt/SMAW/bin/sdtool -r
```

If this action does not correct the problem, copy down this message, collect investigation information, and contact field engineers. For instructions on collecting investigation information, see Section "Collecting Troubleshooting Information".

7230 The Host OS information is not set. (nodename:*nodename*)

Corrective action

Run the clvmgsetup -l command to check the information on the currently registered Host OS. If necessary, register the Host OS information using the clvmgsetup command.
If this action does not correct the problem, copy down this message, collect investigation information, and contact field engineers. For instructions on collecting investigation information, see Section "Collecting Troubleshooting Information".

7231 Cannot find the guest domain name.

Corrective action

Make sure that the node running the clvmgsetup command is a guest domain.
Set the guest domain referring to "3.2.1 Host OS setup" of "*PRIMECLUSTER Installation and Administration Guide*" when executing it by the guest domain.

7232 Cannot find the specified guest domain name. (domainname:*domainname*)

Corrective action

The specified guest domain name does not exist. Confirm the following points, and retry.

1. Is the specified guest domain name correct?
   Confirm whether the specified guest domain name is wrong referring to "*PRIMEQUEST Virtual Machine Function User's Manual*". If the guest domain name is wrong, specify the correct one.
2. Is the specified guest domain running?
   Check whether the guest domain is running. If it is not, start it. For details on how to check and start operation of the guest domain, refer to "*PRIMEQUEST Virtual Machine Function User's Manual*".

If this action does not correct the problem, copy down this message, collect investigation information, and contact field engineers. For instructions on collecting investigation information, see Section "Collecting Troubleshooting Information".

7233 The username or password to login to the Host OS is incorrect.

Corrective action

Register Host OS information by using the clvmgsetup command. Refer to "5.1.2.4 *VMGuest(Virtual Machine)*" of "*PRIMECLUSTER Installation and Administration Guide*" for the registration of Host OS information.
If this action does not correct the problem, copy down this message, collect investigation information, and contact field engineers. For instructions on collecting investigation information, see Section "Collecting Troubleshooting Information".

7234 Connection to the Host OS is refused. (node:*nodename* detail:*code*)

Corrective action
Check the following points.

- Is the IP address specified for the Host OS set correctly?
- Is the IP address allocated to the guest OS correct?
- Is a passphrase set for connection to the Host OS?

If any of the above points turns out to be the cause of the error, take the appropriate corrective action and execute the following command on the node on which the prompt was output:

```
# /opt/SMAW/bin/sdtool -r
```

After that, restart the shutdown facility (SF).
If the connection can still not be established after checking above items, the possible cause may be a hardware damage (e.g. network or hub failure).In that case, please contact field engineers.
If this action does not correct the problem, copy down this message, collect investigation information, and contact field engineers. For instructions on collecting investigation information, see Section "Collecting Troubleshooting Information".

7235 First SSH connection to the Host OS has not been done yet. (node:*nodename* detail:*code*)

Corrective action
Reconnect with the Host OS beforehand by SSH, using the FJSVvmSP account created for forcing stop of the guest OS.

7236 Connection to the Host OS was disconnected. (node:*nodename* detail:*code*)

Corrective action
Check the state of other nodes and path of a private LAN.

7237 clvmgsetup has been executed.

Corrective action
The clvmgsetup command has already been executed. Execute it again after the command under execution has ended.

12 Manual pages

This chapter lists the online manual pages for CCBR, CF, CIP, PAS, PCS, Resource Database, RMS, RMS Wizards, SF, and Web-Based Admin View.

To display a manual page, type the following command:

$ **man** *man_page_name*

12.1 CCBR

System administration

cfbackup
: save the cluster configuration information for a PRIMECLUSTER node

cfrestore
: restore saved cluster configuration formation on a PRIMECLUSTER node

12.2 CF

System administration

cfconfig
: configure or unconfigure a node for a PRIMECLUSTER cluster

cfregd
: CF registry synchronization daemon

cfset
: apply or modify /etc/default/cluster.config entries into the CF module

cftool
: print node communications status for a node or the cluster

rcqconfig
: configure or start quorum

rcqquery
: get quorum state of the cluster

12.3 CIP

System administration

cipconfig
: start or stop CIP 2.0

ciptool
: retrieve CIP information about local and remote nodes in the cluster

File format

cip.cf
: CIP configuration file format

12.4 Monitoring Agent

System administration

cldevparam
: change and display the tunable operation environment for asynchronous monitoring

clmmbmonctl
: start, stop, restart, and display the operating system of the MMB asynchronous monitoring daemon

clmmbsetup
: register, change, delete, and display MMB information

12.5 PAS

System administration

mipcstat
: MIPC statistics

clmstat
: CLM statistics

12.6 PCS

System administration

pcstool
: modify PCS configurations from the command line

pcscui
: character-based interface for PCS

pcs_reinstall
: utility for re-integrating PCS with dependent products

12.7 Resource Database

To display a Resource Database manual page, add /etc/opt/FJSVcluster/man to the environment variable MANPATH.

System administration

clautoconfig
: execute of the automatic resource registration

clbackuprdb
: save the resource database

clexec
: execute the remote command

cldeldevice
: delete resource registered by automatic resource registration

clinitreset
: reset the resource database

clrestorerdb
: restore the resource database

clsetparam
: display and change the resource database operational environment

clsetup
: set up the resource database

clstartrsc
: activate resource (only GDS)

clstoprsc
: deactivate resource (only GDS)

clsyncfile
: distribute a file between cluster nodes

User command

clgettree
: display the tree information of the resource database

12.8 RMS

System administration

hvassert
: assert (test for) an RMS resource state

hvcm
: start the RMS configuration monitor

hvconfig
: display or save the RMS configuration file

hvdisp
: display RMS resource information

hvdump
: collect debugging information about RMS

hvlogclean
: clean RMS log files

hvsetenv
: manipulate RMS rc start or AutoStartUp

hvshut
: shut down RMS

hvswitch
: switch control of an RMS user application resource to another node

thvutil
: manipulate availability of an RMS resource

File formats

config.us
: RMS configuration file

hvenv.local
: RMS local environment variables file

hvgdstartup
: RMS generic detector startup file

12.9 RMS Wizards

RMS Wizards and RMS Application Wizards
: RMS Wizards are documented as html pages in the SMAWRhv-do package on the CD-ROM. After installing this package, the documentation is available in the following directory: /usr/doc/packages/SMAWRhv-do/wizards.en

12.10 SF

System administration

rcsd
: Shutdown Daemon of the Shutdown Facility

sdtool
: interface tool for the Shutdown Daemon

sfcipher
: password encryption

File formats

rcsd.cfg
: configuration file for the Shutdown Daemon

SA_blade.cfg
: configuration file for Blade Shutdown Agent

SA_ipmi.cfg
: configuration file for a Intelligent Platform Management Interface Shutdown Agent

SA_icmp.cfg
: configuration file for ICMP Shutdown Agent

SA_wtinps.cfg
: configuration file for WTI IPS Shutdown Agent

SA_vmSPgp.cfg
: configuration file for Shutdown Agent (for panic) of Virtual Machine Service Provider (vmSP)

SA_vmSPgr.cfg
: configuration file for Shutdown Agent (for reset) of Virtual Machine Service Provider (vmSP)

SA_vmchkhost.cfg
: configuration file for shutdown agent of vmchkhost (Host OS check)

SA_libvirtgp.cfg
: configuration file for libvirt Shutdown Agent (for panic)

SA_libvirtgr.cfg
: configuration file for libvirt Shutdown Agent (for reset)

12.11 Web-Based Admin View

System administration

fjsvwvbs
: stop Web-Based Admin View

fjsvwvcnf
: start, stop, or restart the web server for Web-Based Admin View

wvCntl
: start, stop, or get debugging information for Web-Based Admin View

wvGetparam
: display Web-Based Admin View's environment variable

wvSetparam
: set Web-Based Admin View environment variable

wvstat
: display the operating status of Web-Based Admin View

13 Release information

No	VL	Edition	Section	Description
1	4.3A10	June 2011	"Overview" "Available SAs" "Shutdown Agents"	Added "VMCHKHOST" and "libvirt" to the Shutdown Agent.
2	4.3A10	June 2011	"Overview" "Available SAs" "Shutdown Agents" "SF"	Deleted descriptions for "RSB" and "RPS".
3	4.3A10	June 2011	"Overview" "vmSP"	Added descriptions that "SA_vmSPgp" and "SA_vmSPgr" are for the Xen virtual machine function.
4	4.3A10	June 2011	"Overview" "VMCHKHOST"	Added descriptions that "SA_vmchkhost" is for the Xen/KVM virtual machine function.
5	4.3A10	June 2011	"Available SAs" "Crash Dump"	Deleted descriptions for "diskdump".
6	4.3A10	June 2011	"MMB"	Changed "PSA" to "PSA/SVmco".
7	4.3A10	June 2011	"MMB"	Added reference manuals.
8	4.3A10	June 2011	"SF"	Added "SA_vmchkhost.cfg", "SA_libvirtgp.cfg", and "SA_libvirtgr.cfg" to the File formats.

No	VL	Edition	Section	Description
9	4.3A20	December 2012	"CF, CIP, and CIM configuration" "Differences between CIP and CF over IP" "Example of creating a cluster" "CIP configuration file" "Resource Database configuration" "Configuring the Resource Database on the new node" "Shutdown Agents" "Overview"	Added descriptions of IPv6.
10	4.3A20	December 2012	"CF, CIP, and CIM configuration"	Changed the important note when searching for nodes in the CIP configuration.
11	4.3A20	December 2012	"CF, CIP, and CIM configuration"	Described the configuration which shares the administrative LAN and cluster interconnects on the same NIC.
12	4.3A20	December 2012	"cfcp/cfsh"	Deleted descriptions regarding the security of PRIMECLUSTER.
13	4.3A20	December 2012	"Example of CF configuration by CLI"	Added "Example of CF configuration by CLI."
14	4.3A20	December 2012	"CIP configuration file"	Added conditions to stop CIP.
15	4.3A20	December 2012	"Shutdown Agents"	Changed descriptions of "user" and "password" in the configuration file for libvirt.
16	4.3A20	December 2012	All	Changed the description of SIS.

Glossary

Items in this glossary that apply to specific Cluster Foundation components are indicated with the following notation:

- (CF)-Cluster Foundation
- (PCS)-PRIMECLUSTER Configuration Services
- (RMS)-Reliant Monitor Services
- (RCVM)-Volume Manager (not available in all markets)
- (SIS)-Scalable Internet Services

Some of these products may not be installed on your cluster. See field engineers for more information.

AC

See *Access Client*.

Access Client

GFS kernel module on each node that communicates with the Meta Data Server and provides simultaneous access to a shared file system.

administrative LAN

An optional private local area network (LAN) used for administrative commands to the nodes in the cluster. To provide an extra level of security, normal users do not have access to the administrative LAN. In Cluster Foundation configurations, the System Console and Cluster Console reside on the administrative LAN if one is present.

See also *public LAN*.

API

See *Application Program Interface*.

application (RMS)

In the RMS context, an application object is a special resource used to group other resources into a logical collection. Typically, it is used to represent a real-world application or application suite in a high-availability configuration.

Application Program Interface

A shared boundary between a service provider and the application that uses that service.

application template (RMS)

A predefined group of object definition value choices used by PCS, the Wizard Tools, or the PCS Wizard Kit to create object definitions for a specific type of application.

attribute (RMS)

The part of an object definition that specifies how the base monitor acts and reacts for a particular object type during normal operations.

automatic switchover (RMS)

The procedure by which RMS automatically switches control of a userApplication over to another node after specified conditions are detected.

See also *directed switchover (RMS)*, *failover (RMS, SIS)*, *switchover (RMS)*, *symmetrical switchover (RMS)*.

availability

Availability describes the need of most enterprises to operate applications via the Internet 24 hours a day, 7 days a week. The relationship of the actual to the planned usage time determines the availability of a system.

base cluster foundation (CF)

This Cluster Foundation module resides on top of the basic OS and provides internal interfaces for the CF (Cluster Foundation) functions that the Cluster Foundation services use in the layer above.

See also *Cluster Join Services (CF).*

base monitor (RMS)

The RMS module that maintains the availability of resources. The base monitor is supported by daemons and detectors. Each node being monitored has its own copy of the base monitor.

Cache Fusion

The improved interprocess communication interface in Oracle 9i that allows logical disk blocks (buffers) to be cached in the local memory of each node. Thus, instead of having to flush a block to disk when an

update is required, the block can be copied to another node by passing a message on the interconnect, thereby removing the physical I/O overhead.

CCBR

See *Cluster Configuration Backup and Restore*.

CF

See *Cluster Join Services (CF)*.

CF node name (CF)

The CF cluster node name, which is configured when a CF cluster is created.

child (RMS)

A resource defined in the configuration file that has at least one parent. A child can have multiple parents, and can either have children itself (making it also a parent) or no children (making it a leaf object).

See also *resource (RMS)*, *object (RMS)*, *parent (RMS)*.

cluster

A set of computers that work together as a single computing source. Specifically, a cluster performs a distributed form of parallel computing.

See also *RMS configuration (RMS)*.

Cluster Admin

A Java-based, OS-independent management tool for Cluster Foundation products such as CF, SIS, RMS and PCS. Cluster Admin is available from the Web-Based Admin View interface.

See also *Cluster Foundation (CF)*, *Scalable Internet Services (SIS)*, *Reliant Monitor Services (RMS)*, *PRIMECLUSTER Configuration Services (PCS)*, *Web-Based Admin View*.

Cluster Configuration Backup and Restore

CCBR provides a simple method to save the current Cluster Foundation configuration information of a cluster node. It also provides a method to restore the configuration information.

Cluster Foundation (CF)

The set of Cluster Foundation modules that provides basic clustering communication services.

See also *base cluster foundation (CF)*.

cluster interconnect

The set of private network connections used exclusively for Cluster Foundation communications.

Cluster Interconnect Protocol

CIP is an interface such as eth0 except the physical layer is built on top of the cluster interconnect.

Cluster Join Services (CF)

This Cluster Foundation module handles the forming of a new cluster and the addition of nodes.

concatenated virtual disk (RCVM)

Concatenated virtual disks consist of two or more pieces on one or more disk drives. They correspond to the sum of their parts. Unlike simple virtual disks where the disk is subdivided into small pieces, the individual disks or partitions are combined to form a single large logical disk.

See also, *mirror virtual disk (RCVM)*, *simple virtual disk (RCVM)*, *striped virtual disk (RCVM)*, *virtual disk*.

Configuration Definition Language (PCS)

The syntax for PCS configuration templates.

See also *PRIMECLUSTER Configuration Services (PCS)*.

configuration file (RMS)

In the RMS context, the single file that defines the monitored resources and establishes the interdependencies between them. The default name of this file is config.us.

console

See *single console*.

custom detector (RMS)

See *detector (RMS)*.

custom type (RMS)

See *graph (RMS)*.

daemon

A continuous process that performs a specific function repeatedly.

database node (SIS)

Nodes that maintain the configuration, dynamic data, and statistics in a SIS configuration.

See also *gateway node (SIS)*, *service node (SIS)*, *Scalable Internet Services (SIS)*.

detector (RMS)

A process that monitors the state of a specific object type and reports a change in the resource state to the base monitor.

DHCP

Dynamic Host Control Protocol. A standard method of delivering information to a host at boot time. This is most often used to dynamically assign the host's IP address and netmask, but many other parameters are possible, including domain names, DNS servers, and time servers.

directed switchover (RMS)

The RMS procedure by which an administrator switches control of a userApplication over to another node.

See also *automatic switchover (RMS)*, *failover (RMS, SIS)*, *switchover (RMS)*, *symmetrical switchover (RMS)*.

DOWN (CF)

A node state that indicates that the node is unavailable (marked as down). A LEFTCLUSTER node must be marked as DOWN before it can rejoin a cluster.

See also *UP (CF)*, *LEFTCLUSTER (CF)*, *node state (CF)*.

Enhanced Lock Manager (ELM) (CF)

A light weight, high performance, highly responsive lock manger, specifically designed for providing a high reliability heartbeat messaging mechanism for Cluster Foundation modules.

ENS (CF)

See *Event Notification Services (CF)*.

environment variables

Variables or parameters that are defined globally.

error detection (RMS)

The process of detecting an error. For RMS, this includes initiating a log entry, sending a message to a log file, or making an appropriate recovery response.

Event Notification Services (CF)

This Cluster Foundation module provides an atomic-broadcast facility for events.

failover (RMS, SIS)

With SIS, this process switches a failed node to a backup node. With RMS, this process is known as switchover.

See also *automatic switchover (RMS)*, *directed switchover (RMS)*, *switchover (RMS)*, *symmetrical switchover (RMS)*.

gateway node (SIS)

Gateway nodes have an external network interface. All incoming packets are received by this node and forwarded to the selected service node, depending on the scheduling algorithm for the service.

See also *service node (SIS)*, *database node (SIS)*, *Scalable Internet Services (SIS)*.

GDS

See *Global Disk Services*.

generic type (RMS)

An object type which has generic properties. A generic type is used to customize RMS for monitoring resources that cannot be assigned to one of the supplied object types.

See also *object type (RMS)*.

GFS

See *Global File Services*.

Global Disk Services

This optional product provides volume management that improves the availability and manageability of information stored on the disk unit of the Storage Area Network (SAN).

Global File Services

This optional product provides direct, simultaneous accessing of the file system on the shared storage unit from two or more nodes within a cluster.

Global Link Services

This optional products provides network high availability solutions by multiplying a network route.

GLS

See *Global Link Services*.

graph (RMS)

See *system graph (RMS)*.

graphical user interface

A computer interface with windows, icons, toolbars, and pull-down menus that is designed to be simpler to use than the command-line interface.

GUI

See *graphical user interface*.

HBA blockage function

Function that stops the HBA in the switchover source when node switching takes place. This function enables PRIMECLUSTER node switching to be executed at high speed. This function shortens the node switchover time by forcibly stopping all ongoing input/output processes in the HBA of the switchover source.

high availability

A system design philosophy in which redundant resources are employed to avoid single points of failure.

See also *Reliant Monitor Services (RMS)*.

interconnect (CF)

See *cluster interconnect*.

Internet Protocol address

A numeric address that can be assigned to computers or applications.

See also *IP address*.

Internode Communications facility

This module is the network transport layer for all Cluster Foundation internode communications. It interfaces by means of OS-dependent code to the network I/O subsystem and guarantees delivery of messages queued for transmission to the destination node in the same sequential order unless the destination node fails.

IP address

See *Internet Protocol address*.

IP aliasing

This enables several IP addresses (aliases) to be allocated to one physical network interface. With IP aliasing, the user can continue communicating with the same IP address, even though the application is now running on another node.

See also *Internet Protocol address*.

JOIN (CF)

See *Cluster Join Services (CF)*.

keyword

A word that has special meaning in a programming language. For example, in an RMS configuration file, the keyword object identifies the kind of definition that follows.

leaf object (RMS)

A bottom object in a system graph. In the configuration file, this object definition is at the beginning of the file. A leaf object does not have children.

LEFTCLUSTER (CF)

A node state that indicates that the node cannot communicate with other nodes in the cluster. That is, the node has left the cluster. The reason for the intermediate LEFTCLUSTER state is to avoid the network partition problem.

See also *UP (CF)*, *DOWN (CF)*, *network partition (CF)*, *node state (CF)*.

link (RMS)

Designates a child or parent relationship between specific resources.

local area network

See *Reliant Monitor Services (RMS)*.

local node

The node from which a command or process is initiated.

See also *remote node*, *mirror virtual disk (RCVM)*.

log file

The file that contains a record of significant system events or messages. The ASCC control and satellite daemons maintain log files on every node on which they run. The Wizard Tools, PCS, the RMS base monitor, and RMS detectors each maintain their own log files as well.

Management Board

One of the hardware units installed in PRIMEQUEST.

Management Information Base

A hierarchical database of information about the local network device. The database is maintained by network management software such as an SNMP agent.

See also *Simple Network Management Protocol*.

MDS

See *Meta Data Server*.

message

A set of data transmitted from one software process to another process, device, or file.

message queue

A designated memory area which acts as a holding place for messages so they can be processed in the same order they were received.

Meta Data Server

GFS daemon that centrally manages the control information, or meta-data, of a file system.

MIB

See *Management Information Base*.

MIPC

Mesh Interprocessor Communication

mirror virtual disk (RCVM)

Mirror virtual disks consist of two or more physical devices, and all output operations are performed simultaneously on all of the devices.

See also *concatenated virtual disk (RCVM)*, *simple virtual disk (RCVM)*, *striped virtual disk (RCVM)*, *virtual disk*.

mirrored disks (RCVM)

A set of disks that contain the same data. If one disk fails, the remaining disks of the set are still available, preventing an interruption in data availability.

See also *mirrored pieces (RCVM)*.

mirrored pieces (RCVM)

Physical pieces that together comprise a mirrored virtual disk. These pieces include mirrored disks and data disks.

See also *mirrored disks (RCVM)*.

MMB

Abbreviation for Management Board, which is one of the hardware units installed in PRIMEQUEST.

mount point

The point in the directory tree where a file system is attached.

multihosting

Multiple controllers simultaneously accessing a set of disk drives.

native operating system

The part of an operating system that is always active and translates system calls into activities.

network partition (CF)

This condition exists when two or more nodes in a cluster cannot communicate over the interconnect; however, with applications still running, the nodes can continue to read and write to a shared device, compromising data integrity.

node

A host that is a member of a cluster.

node state (CF)

Every node in a cluster maintains a local state for every other node in that cluster. The node state of every node in the cluster must be either UP, DOWN, or LEFTCLUSTER.

See also *UP (CF)*, *DOWN (CF)*, *LEFTCLUSTER (CF)*.

object (RMS)

A representation of a physical or virtual resource in the RMS configuration file or in a system graph.

See also *leaf object (RMS)*, *object definition (RMS)*, *object type (RMS)*.

object definition (RMS)

An entry in the configuration file that identifies a resource to be monitored by RMS. Attributes included in the definition specify properties of the corresponding resource.

See also *attribute (RMS)*, *object (RMS)*.

object type (RMS)

A category of similar resources monitored as a group, such as disk drives. Each object type has specific properties, or attributes, which limit or define what monitoring or action can occur. When a resource is associated with a particular object type, attributes associated with that object type are applied to the resource.

See also *graph (RMS)*.

online maintenance

The capability of adding, removing, replacing, or recovering devices without shutting or powering off the node.

operating system dependent (CF)

This module provides an interface between the native operating system and the abstract, OS-independent interface that all Cluster Foundation modules depend upon.

Oracle Real Application Clusters (RAC)

Oracle RAC allows access to all data in a database to users and applications in a clustered or MPP (massively parallel processing) platform. Formerly known as Oracle Parallel Server (OPS).

OSD (CF)

See *operating system dependent (CF)*.

parent (RMS)

An object in the RMS configuration file or system graph that has at least one child.

See also *child (RMS)*, *configuration file (RMS)*, *leaf object (RMS)*, *system graph (RMS)*.

PCS

See *PRIMECLUSTER Configuration Services (PCS)*.

PCS Wizard Kit (PCS)

RMS configuration products that have been designed for specific applications. Each component of the PCS Wizard Kit includes customized default settings, subapplications, detectors, and scripts. These application wizards also tailor the PCS interface to provide controls for the additional features.

See also *PRIMECLUSTER Configuration Services (PCS)*, *Reliant Monitor Services (RMS)*.

primary node (RMS)

The default node on which a user application comes online when RMS is started. This is always the node name of the first child listed in the userApplication object definition.

PRIMECLUSTER Configuration Services (PCS)

The graphical configuration interface for Cluster Foundation products. PCS uses standard templates written in Configuration Definition Language (CDL) to provide a user-friendly configuration environment for products such as RMS. The standard templates can be modified or replaced to provide a customized interface for specific applications or installations.

PRIMECLUSTER services (CF)

Service modules that provide services and internal interfaces for clustered applications.

private network addresses

Private network addresses are a reserved range of IP addresses specified by the Internet Corporation for Assigned Names and Numbers (ICANN). Modern switches and routers prevent these addresses from being routed to the Internet, allowing two or more organizations to assign the same private addresses for internal use without causing conflicts or security risks.

private resource (RMS)

A resource accessible only by a single node and not accessible to other RMS nodes.

See also *resource (RMS)*, *shared resource*.

public LAN

The local area network (LAN) by which normal users access a machine.

See also *administrative LAN*.

queue

See *message queue*.

redundancy

The capability of one component to assume the resource load of another physically similar component in case the original component fails or is shut down. Common examples include RAID hardware and/or RAID software to replicate data stored on secondary storage devices, multiple network connections to provide alternate data paths, and multiple nodes that can be dynamically reprovisioned to maintain critical services in a cluster.

Reliant Monitor Services (RMS)

The package that maintains high availability of user-specified resources by providing monitoring and switchover capabilities.

remote node

A node that is accessed through a LAN or telecommunications line.

See also *local node*, *node*.

reporting message (RMS)

A message that a detector uses to report the state of a particular resource to the base monitor.

resource (RMS)

A hardware or software element (private or shared) that provides a function, such as a mirrored disk, mirrored disk pieces, or a database server. A local resource is monitored only by the local node.

See also *private resource (RMS)*, *shared resource*.

resource definition (RMS)

See *object definition (RMS)*.

resource label (RMS)

The name of the resource as displayed in a system graph.

resource state (RMS)

Current state of a resource.

RMS

See *queue*.

RMS commands (RMS)

Commands that enable RMS resources to be administered from the command line.

RMS configuration (RMS)

A configuration made up of two or more nodes connected to shared resources. Each node has its own copy of operating system and RMS software, as well as its own applications.

RMS Wizard Kit (RMS)

RMS configuration products that have been designed for specific applications. Each component of the Wizard Kit includes customized default settings, subapplications, detectors, and scripts. These application wizards also tailor the RMS Wizard Tools interface to provide controls for the additional features.

See also *RMS Wizard Tools (RMS)*, *Reliant Monitor Services (RMS)*.

RMS Wizard Tools (RMS)

A software package composed of various configuration and administration tools used to create and manage applications in an RMS configuration.

See also, *RMS Wizard Kit (RMS)*, *Reliant Monitor Services (RMS)*.

SAN

See *Storage Area Network*.

scalability

The ability of a computing system to efficiently handle any dynamic change in work load. Scalability is especially important for Internet-based applications where growth caused by Internet usage presents a scalable challenge.

Scalable Internet Services (SIS)

The package that dynamically balances network traffic loads across cluster nodes while maintaining normal client/server sessions for each connection.

SCON

See *single console*.

script (RMS)

A shell program executed by the base monitor in response to a state transition in a resource. The script may cause the state of a resource to change.

service node (SIS)

Service nodes provide one or more TCP services (such as FTP, Telnet, and HTTP) and receive client requests forwarded by the gateway nodes.

See also *database node (SIS)*, *gateway node (SIS)*, *Scalable Internet Services (SIS)*.

SF

See *Shutdown Facility*.

shared resource

A resource, such as a disk drive, that is accessible to more than one node.

See also *private resource (RMS)*, *resource (RMS)*.

Shutdown Facility

The Cluster Foundation interface that manages the shutdown and startup of cluster nodes. The SF is automatically invoked during failover operations. It also notifies other Cluster Foundation products of the successful completion of node shutdown so that recovery operations can begin.

Simple Network Management Protocol

A set of protocols that facilitates the exchange of information between managed network devices. The protocols are implemented by software agents residing in the devices. Each agent can read and write data in the local Management Information Base (MIB) in response to SNMP requests from other devices on the network.

See also *Management Information Base*.

simple virtual disk (RCVM)

Simple virtual disks define either an area within a physical disk partition or an entire partition.

See also *concatenated virtual disk (RCVM)*, *mirror virtual disk (RCVM)*, *striped virtual disk (RCVM)*, *virtual disk*.

single console

The workstation that acts as the single point of administration for nodes being monitored by RMS. The single console software, SCON, is run from the single console.

SIS

See *Scalable Internet Services (SIS)*.

SNMP

See *Simple Network Management Protocol*.

state

See *resource state (RMS)*.

Storage Area Network

The high-speed network that connects multiple, external storage units and storage units with multiple computers. The connections are generally fiber channels.

striped virtual disk (RCVM)

Striped virtual disks consist of two or more pieces. These can be physical partitions or further virtual disks (typically a mirror disk). Sequential I/O operations on the virtual disk can be converted to I/O operations on two or more physical disks. This corresponds to RAID Level 0 (RAID0).

See also *concatenated virtual disk (RCVM)*, *mirror virtual disk (RCVM)*, *simple virtual disk (RCVM)*, *virtual disk*.

switchover (RMS)

The process by which RMS switches control of a `userApplication` over from one monitored node to another.

See also *automatic switchover (RMS)*, *directed switchover (RMS)*, *failover (RMS, SIS)*, *symmetrical switchover (RMS)*.

symmetrical switchover (RMS)

This means that every RMS node is able to take on resources from any other RMS node.

See also *automatic switchover (RMS)*, *directed switchover (RMS)*, *failover (RMS, SIS)*, *switchover (RMS)*.

system disk (GDS)

Disk on which the active operating system is installed. System disk refers to the entire disk that contains the slices that are currently operating as one of the following file systems (or the swap area):

For Solaris: /, /usr, /var , or swap area
For Linux: /, /usr, /var, /boot, /boot/efi, or swap area

system graph (RMS)

A visual representation (a map) of monitored resources used to develop or interpret the RMS configuration file.

See also *configuration file (RMS)*.

template

See *application template (RMS)*.

type

See *object type (RMS)*.

UP (CF)

A node state that indicates that the node can communicate with other nodes in the cluster.

See also *DOWN (CF)*, *LEFTCLUSTER (CF)*, *node state (CF)*.

virtual disk

A pseudo-device that allows a portion or a combination of physical disks to be treated as a single logical disk. The virtual disk driver is inserted between the highest level of the OS logical input/output (I/O) system and the physical device driver(s), allowing all logical I/O requests to be mapped to the appropriate area on the physical disk(s).

See also *concatenated virtual disk (RCVM)*, *mirror virtual disk (RCVM)*, *simple virtual disk (RCVM)*, *striped virtual disk (RCVM)*.

Web-Based Admin View

A Java-based, OS-independent interface to Cluster Foundation management components.

See also *Cluster Admin*.

wizard (RMS)

An interactive software tool that creates a specific type of application using pretested object definitions.

Wizard Kit (RMS)

See *PCS Wizard Kit (PCS)*, *RMS Wizard Kit (RMS)*.

Wizard Tools (RMS)

See *RMS Wizard Kit (RMS)*.

Abbreviations

AC
Access Client

API
application program interface

bm
base monitor

CCBR
Cluster Configuration Backup/Restore

CF
Cluster Foundation or Cluster Framework

CIM
Cluster Integrity Monitor

CIP
Cluster Interconnect Protocol

CLI
command-line interface

CRM
Cluster Resource Management

DLPI
Data Link Provider Interface

ENS
Event Notification Services

GDS
Global Disk Services

GFS
Global File Services

GLS
Global Link Services

GUI
graphical user interface

HA
high availability

HBA
Host Bus Adapter (another name for the Fibre Channel card)

ICF
Internode Communication Facility

I/O
input/output

JOIN
cluster join services module

LAN
local area network

MDS
Meta Data Server

MIB
Management Information Base

MIPC
Mesh Interprocessor Communication

MMB
Management Board

NIC
network interface card

NSM
Node State Monitor

OPS
Oracle Parallel Server

OSD
operating system dependant

PAS
Parallel Application Services

RCCU
Remote Console Control Unit

RCI
Remote Cabinet Interface

RMS
Reliant Monitor Services

RTP
Reliant Telco Product

SA
Shutdown Agent

SAN
Storage Area Network

SCON
single console software

SD
Shutdown Daemon

SF
Shutdown Facility

SIS
Scalable Internet Services

VIP
Virtual Interface Provider

Figures

Tables

Index

D

W