

# **FUJITSU Software**

## **PRIMECLUSTER**

A horizontal band featuring a red abstract graphic with flowing, curved lines and bright light spots, creating a sense of motion and energy.

### **Cluster Foundation (CF) Configuration and Administration Guide 4.3**

Linux

J2UZ-5284-05ENZ0(03)  
August 2016

## 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.

All Rights Reserved, Copyright (C) FUJITSU LIMITED 2012-2016.

---

## **1 Preface 1**

- 1.1 Contents of this manual 1
- 1.2 PRIMECLUSTER documentation 2
- 1.3 Conventions 3
  - 1.3.1 Notation 3
    - 1.3.1.1 Prompts 3
    - 1.3.1.2 The keyboard 3
    - 1.3.1.3 Typefaces 4
    - 1.3.1.4 Example 1 4
    - 1.3.1.5 Example 2 4
  - 1.3.2 Command syntax 4
- 1.4 Important notes and cautions 5
- 1.5 Abbreviations 5
- 1.6 Revision history 6

## **2 Cluster Foundation 7**

- 2.1 CF, CIP, and CIM configuration 7
  - 2.1.1 Differences between CIP and CF over IP 13
  - 2.1.2 cfset 15
  - 2.1.3 CF security 18
    - 2.1.3.1 cfcp/cfsh 18
    - 2.1.3.2 sshconf 19
  - 2.1.4 Signed applets 20
  - 2.1.5 Example of creating a cluster 20
  - 2.1.6 Adding a new node to CF 40
  - 2.1.7 Example of CF configuration by CLI 41
- 2.2 CIP configuration file 43

## **3 CF Registry and Integrity Monitor 45**

- 3.1 CF Registry 45
- 3.2 Cluster Integrity Monitor 46
  - 3.2.1 Configuring CIM 47
  - 3.2.2 Query of the quorum state 48
  - 3.2.3 Reconfiguring quorum 48

## **4 Cluster resource management 51**

- 4.1 Overview 51
- 4.2 Kernel parameters for Resource Database 51
- 4.3 Resource Database configuration 54
- 4.4 Start up synchronization 56
  - 4.4.1 Start up synchronization and the new node 58
- 4.5 Adding a new node 58

- 
- 4.5.1 Backing up the Resource Database 60
  - 4.5.2 Reconfiguring the Resource Database 61
  - 4.5.3 Configuring the Resource Database on the new node 62
  - 4.5.4 Adjusting StartingWaitTime 63
  - 4.5.5 Restoring the Resource Database 63

## **5 GUI administration 65**

- 5.1 Overview 66
- 5.2 Starting Cluster Admin GUI and logging in 66
- 5.3 Main CF table 70
- 5.4 CF route tracking 72
- 5.5 Node details 75
- 5.6 Displaying the topology table 76
- 5.7 Starting and stopping CF 78
  - 5.7.1 Starting CF 81
  - 5.7.2 Stopping CF 83
- 5.8 Marking nodes DOWN 84
- 5.9 Using PRIMECLUSTER log viewer 84
  - 5.9.1 Search based on time filter 86
  - 5.9.2 Search based on keyword 87
  - 5.9.3 Search based on severity levels 88
- 5.10 Displaying statistics 89
- 5.11 Heartbeat monitor 93
- 5.12 Adding and removing a node from CIM 95
- 5.13 Unconfigure CF 97
- 5.14 CIM Override 98

## **6 LEFTCLUSTER state 99**

- 6.1 Description of the LEFTCLUSTER state 100
- 6.2 Recovering from LEFTCLUSTER 102
  - 6.2.1 Caused by a panic/hung node 102
  - 6.2.2 Caused by staying in the kernel debugger too long 104
  - 6.2.3 Caused by a cluster partition 104

## **7 CF topology table 107**

- 7.1 Basic layout 108
- 7.2 Selecting devices 109
- 7.3 Examples 110

## **8 Shutdown Facility 113**

- 8.1 Overview 113

- 
- 8.2 Available SAs 116
    - 8.2.1 IPS 116
    - 8.2.2 SNMP 117
    - 8.2.3 Blade 118
    - 8.2.4 IPMI 118
    - 8.2.5 kdump 119
    - 8.2.6 MMB 119
    - 8.2.7 vmgp 120
    - 8.2.8 vmSP 120
    - 8.2.9 ICMP 120
    - 8.2.10 VMCHKHOST 121
    - 8.2.11 libvirt 121
    - 8.2.12 Sun Fire 121
  - 8.3 SF split-brain handling 121
    - 8.3.1 Administrative LAN 121
    - 8.3.2 SF split-brain handling 122
      - 8.3.2.1 RMS ShutdownPriority attribute 122
      - 8.3.2.2 Shutdown Facility weight assignment 123
      - 8.3.2.3 Disabling split-brain handling 123
    - 8.3.3 Runtime processing 123
    - 8.3.4 Configuration notes 124
  - 8.4 Configuring the Shutdown Facility 127
    - 8.4.1 Invoking the Configuration Wizard 127
    - 8.4.2 Configuration via CLI 145
      - 8.4.2.1 Shutdown Daemon 145
      - 8.4.2.2 Shutdown Agents 147
  - 8.5 SF administration 157
    - 8.5.1 Starting and stopping SF 157
      - 8.5.1.1 Starting and stopping SF manually 158
      - 8.5.1.2 Starting and stopping SF automatically 158
    - 8.5.2 Checking SA status 158
  - 8.6 Logging 161

## **9 Diagnostics and troubleshooting 163**

- 9.1 Beginning the process 163
- 9.2 Symptoms and solutions 167
  - 9.2.1 Join-related problems 167
    - 9.2.1.1 Identifying join-related problems 167
    - 9.2.1.2 Solving join-related problems 168
- 9.3 Collecting Troubleshooting Information 174
  - 9.3.1 Executing the fjsnap or pclsnap Command 175
  - 9.3.2 Crash Dump 176

---

## **10 Manual pages 177**

- 10.1 CCBR 177
- 10.2 CF 177
- 10.3 CIP 178
- 10.4 Monitoring Agent 178
- 10.5 PAS 178
- 10.6 Resource Database 179
- 10.7 RMS 180
- 10.8 RMS Wizards 181
- 10.9 SF 181
- 10.10 Web-Based Admin View 182

## **11 Release information 183**

### **Glossary 187**

### **Abbreviations 205**

### **Figures 209**

### **Tables 213**

### **Index 215**

---

# 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
- RCVN 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 "Diagnostics and troubleshooting" provides help for troubleshooting and problem resolution for PRIMECLUSTER Cluster Foundation.
- 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.
- "PRIMECLUSTER Reliant Monitor Services (RMS) with Wizard Tools Configuration and Administration Guide" —Provides instructions for configuring and administering PRIMECLUSTER Reliant Monitor Services using the Wizard Tools interface. It also describes diagnostic procedures to solve RMS configuration problems, including how to view and interpret RMS log files.
- "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"—Describes the PRIMECLUSTER environment configuration and messages occurred during the operation.
- "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 *fujj3#* would mean that the command was run as user *root* on the node named *fujj3*.

#### 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.
<i>Italic</i>	<p>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, <code>&lt;nodename&gt;RMS</code>; unless directed otherwise, you should not enter the angle brackets.</p> <p>The name of an item in a character-based or graphical user interface.</p> <p>This may refer to a menu item, a radio button, a checkbox, a text input box, a panel, or a window title.</p>
<b>Bold</b>	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.

PRIMEQUEST 2000/1000 Series is abbreviated as PRIMEQUEST.

## 1.6 Revision history

Revision	Location	Manual code
Deleted the explanations of the following commands: - hvdist - hvgdmake - hvrclev - hvreset - hvthrottle	10.7	J2UZ-5284-05ENZ0(01)
Changed the description about the cluster Resource Database.	4.1	J2UZ-5284-05ENZ0(02)
Added the description for the configuration of sharing NIC with administrative LAN and cluster interconnect in the VMware environment.	2.1	J2UZ-5284-05ENZ0(03)
Added the corrective action in case of failure with rcqconfig command.	2.1.7	
Changed the description of Shutdown Daemon (SD).	8.1	
Added "Checking SA status."	8.5.2	

---

## 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.

### 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.

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. 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 "PRIME-CLUSTER *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 KVM environment or VMware environment, the configuration which shares the administrative LAN and cluster interconnects on the NIC is also supported.

### **KVM environment**

In the configuration which shares the administrative LAN and cluster interconnects on the 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.
- Restriction on the starting and stopping of GLS, and the rebooting for network service of system  
When stopping and starting GLS, or rebooting the network service of System, stop CF beforehand. For instructions on stopping CF, refer to the Section "Starting and stopping CF".

**VMware environment**

When sharing NIC with administrative LAN and cluster interconnect in the VMware environment, separate the network allocated to the virtual machine using VMware's function. In this case, CF configuration can be conducted from GUI.

## 2.1.1 Differences between CIP and CF over IP



CF over IP is not available in Linux.

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/UDP 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).

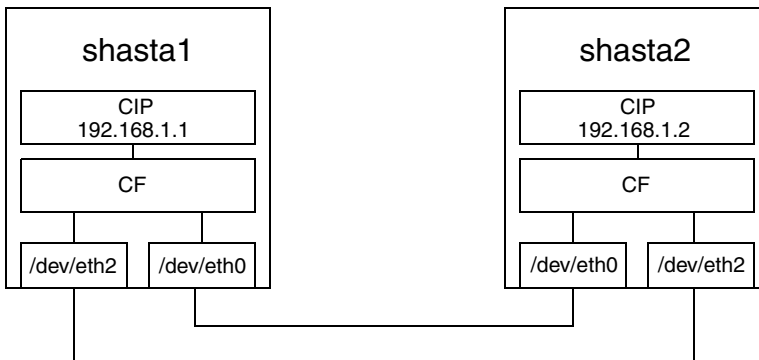


Figure 1: CIP diagram

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).

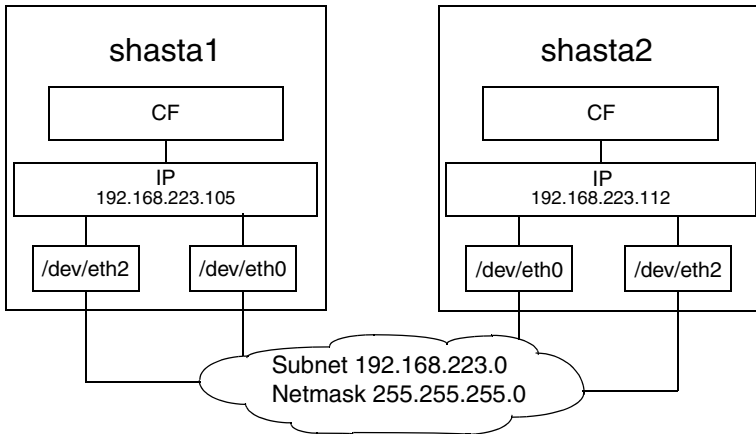


Figure 2: CF over IP diagram

## 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 `cfc` and `cfsh` during initial cluster configuration) or by using a text editor.
- The file consists of the following tuple 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 if it 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).

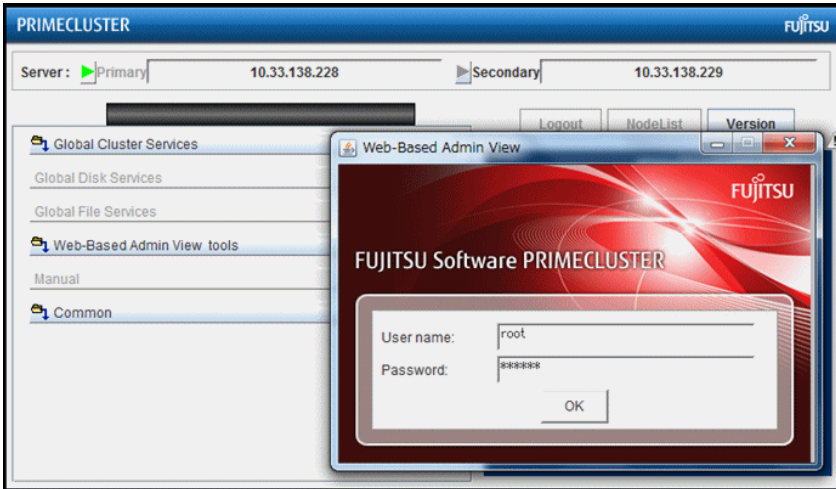


Figure 3: Login pop-up

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 *PRIME-CLUSTER 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).

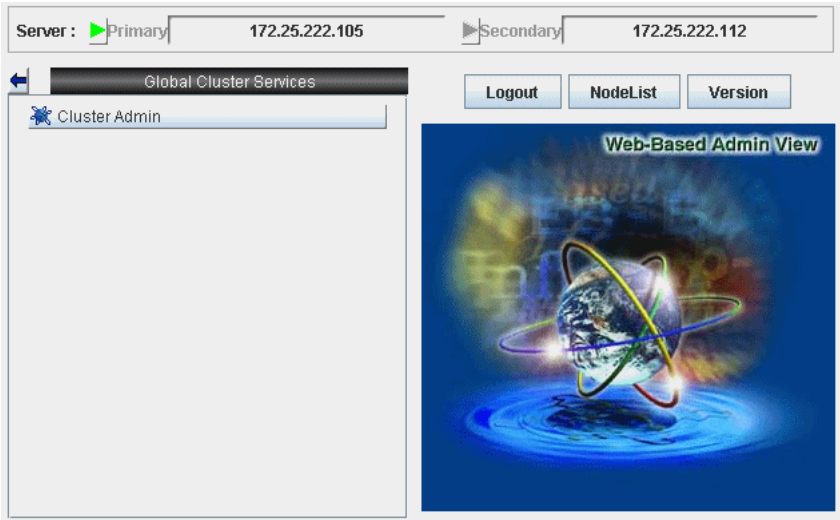


Figure 5: Global Cluster Services window in Web-Based Admin View

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

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

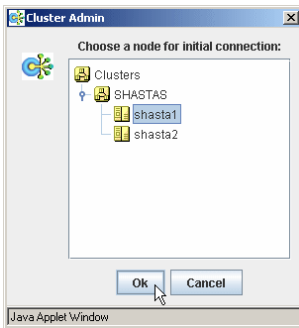


Figure 6: Initial connection pop-up

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.

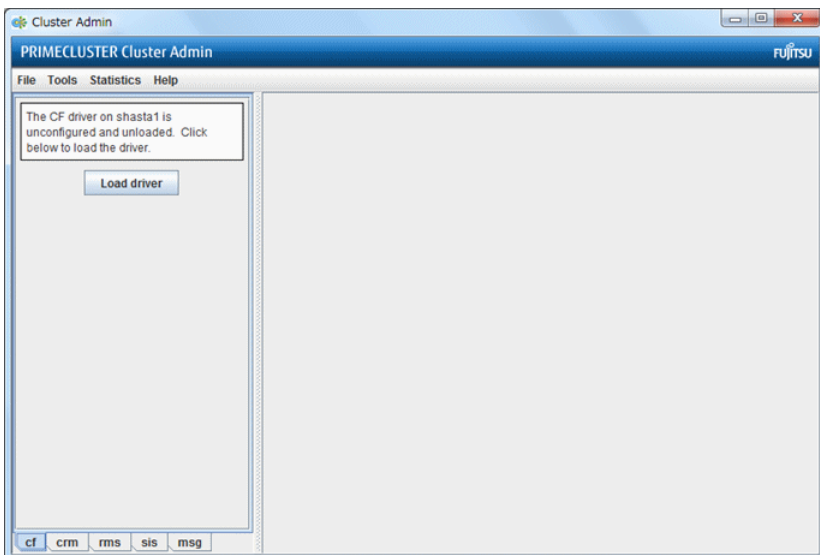


Figure 7: CF is unconfigured and unloaded

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

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

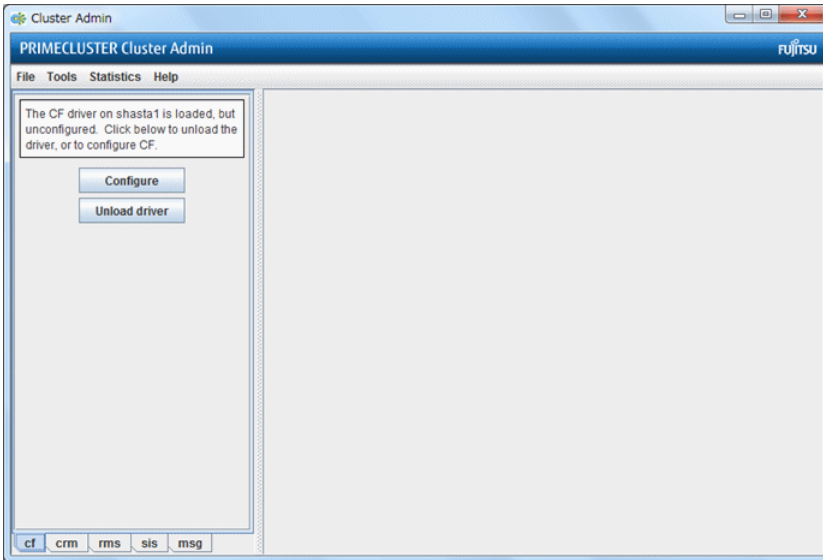


Figure 8: CF loaded but not configured

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.

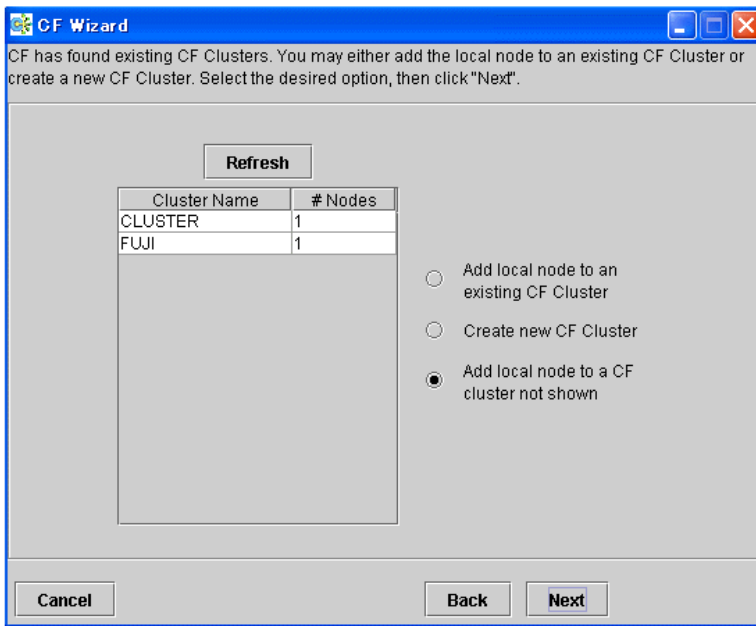


Figure 9: Creating or joining a cluster

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*.

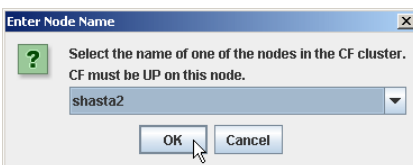


Figure 10: Adding a local node to a CF cluster not shown

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.

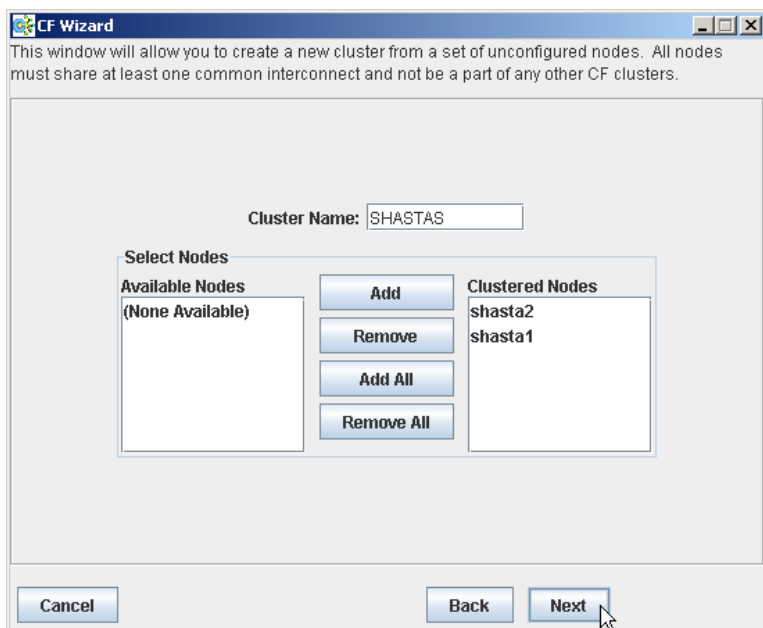


Figure 11: Selecting cluster nodes and the cluster name

This window lets you choose 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.

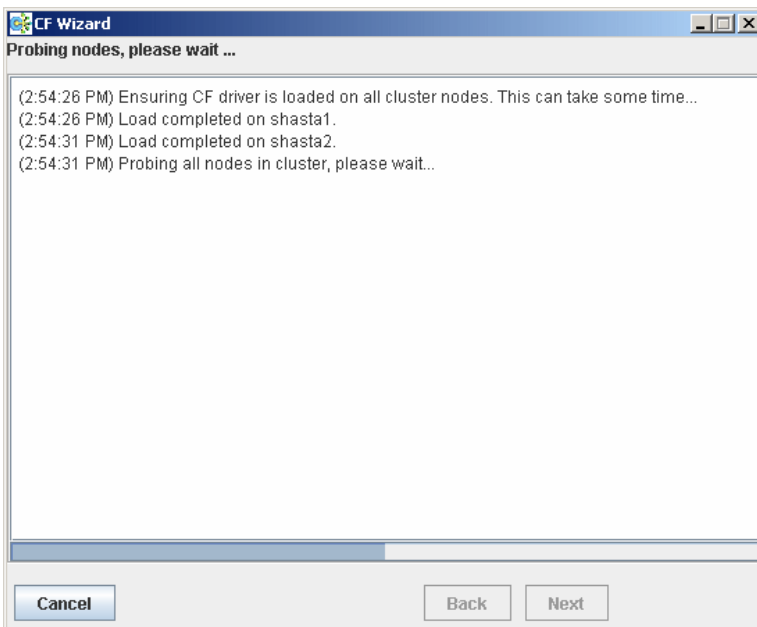
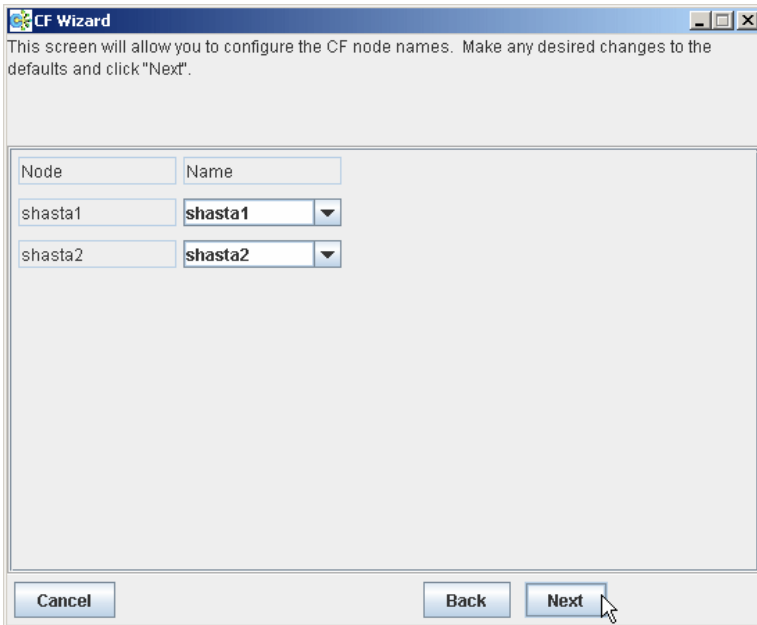


Figure 12: CF loads and pings

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.



The image shows a Windows-style dialog box titled "CF Wizard". The title bar includes standard minimize, maximize, and close buttons. The main text area contains the instruction: "This screen will allow you to configure the CF node names. Make any desired changes to the defaults and click 'Next'." Below this text is a table with two columns: "Node" and "Name". There are two rows of data. The first row shows "shasta1" in the "Node" column and "shasta1" in the "Name" column, with a small downward arrow next to the name. The second row shows "shasta2" in the "Node" column and "shasta2" in the "Name" column, also with a small downward arrow next to the name. At the bottom of the dialog, there are three buttons: "Cancel" on the left, "Back" in the center, and "Next" on the right. A mouse cursor is pointing at the "Next" button.

Node	Name
shasta1	shasta1
shasta2	shasta2

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).

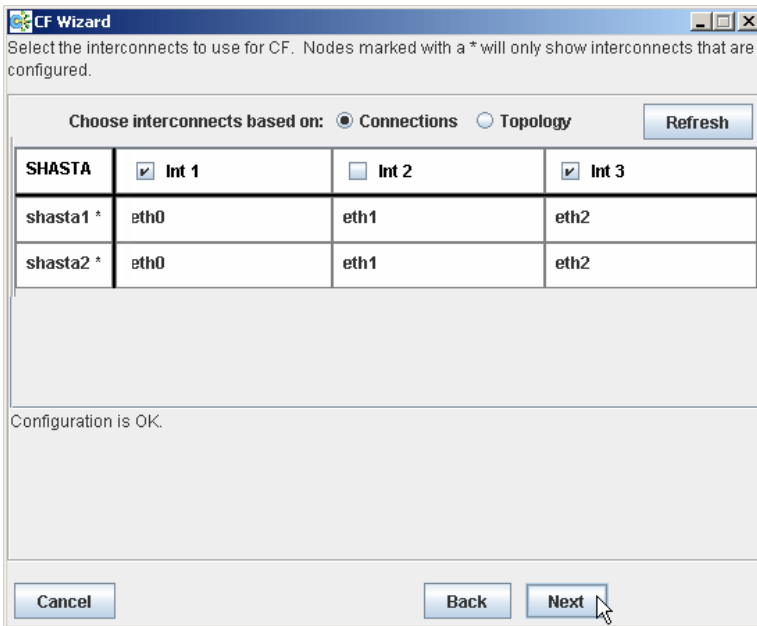


Figure 14: CF topology and connection table

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 reprobe 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).

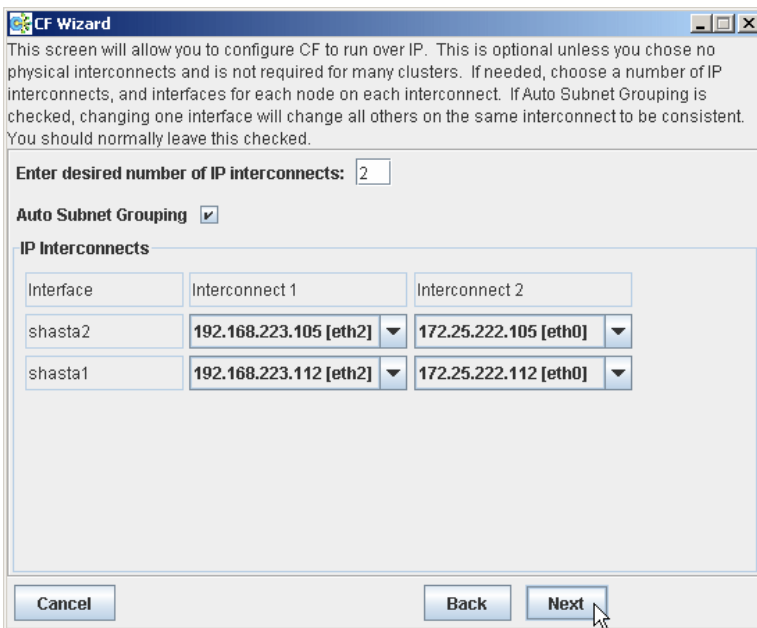


Figure 15: CF over IP window

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.

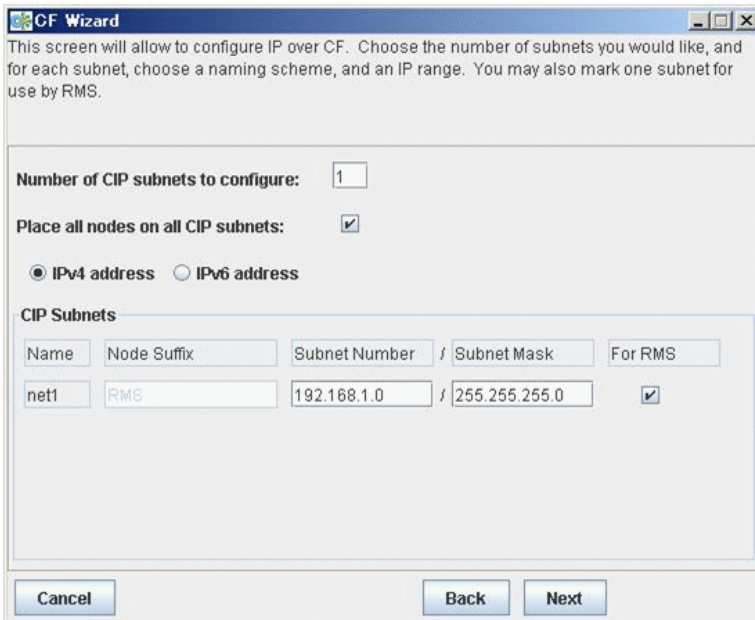
Select the subnetworks that you want to use as your IP interconnects. You should avoid using addresses on your public network. CF allows promiscuous joins without any limits, so it is best to use private subnetworks for your IP interconnects.

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.



The screenshot shows the 'CF Wizard' window with the title bar. The main text reads: 'This screen will allow to configure IP over CF. Choose the number of subnets you would like, and for each subnet, choose a naming scheme, and an IP range. You may also mark one subnet for use by RMS.' Below this, there are several configuration options: 'Number of CIP subnets to configure:' with a text box containing '1'; 'Place all nodes on all CIP subnets:' with a checked checkbox; and radio buttons for 'IPv4 address' (selected) and 'IPv6 address'. A section titled 'CIP Subnets' contains a table with five columns: 'Name', 'Node Suffix', 'Subnet Number', 'Subnet Mask', and 'For RMS'. The first row of the table has the following values: 'net1', 'RMS', '192.168.1.0', '255.255.255.0', and a checked checkbox. At the bottom of the window are three buttons: 'Cancel', 'Back', and 'Next'.

Name	Node Suffix	Subnet Number	Subnet Mask	For RMS
net1	RMS	192.168.1.0	255.255.255.0	<input checked="" type="checkbox"/>

Figure 16: CIP wizard (IPv4) window

The screenshot shows the 'CF Wizard' window. At the top, a text box explains: 'This screen will allow to configure IP over CF. Choose the number of subnets you would like, and for each subnet, choose a naming scheme, and an IP range. You may also mark one subnet for use by RMS.' Below this, there are several configuration options:

- Number of CIP subnets to configure:** A text box containing the value '1'.
- Place all nodes on all CIP subnets:** A checked checkbox.
- IP Address Selection:** Two radio buttons. 'IPv4 address' is unselected, and 'IPv6 address' is selected.
- CIP Subnets Table:** A table with five columns: 'Name', 'Node Suffix', 'Network Prefix', 'Prefix Length', and 'For RMS'.
 

Name	Node Suffix	Network Prefix	Prefix Length	For RMS
net1	RMS	FD00:0:0:1::	64	<input checked="" type="checkbox"/>

At the bottom of the window are three buttons: 'Cancel', 'Back', and 'Next'.

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* field 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:  
CIP 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:  
CIP 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:  
CIP nodename: fuji2RMS  
IPv6 address: FD00:0:0:1::1  
Prefix Length: 64
- On fuji3, a CIP interface will be configured with the following:  
CIP 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).

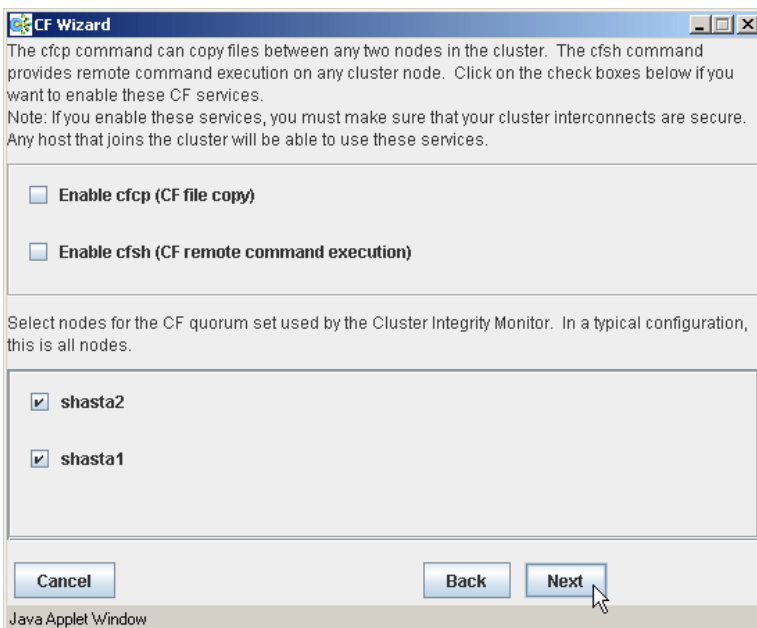


Figure 18: CIM configuration window

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

- The upper portion allows you to enable `cfcf` and `cfsh`.  
`cfcf` 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 `cfcf`, 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).

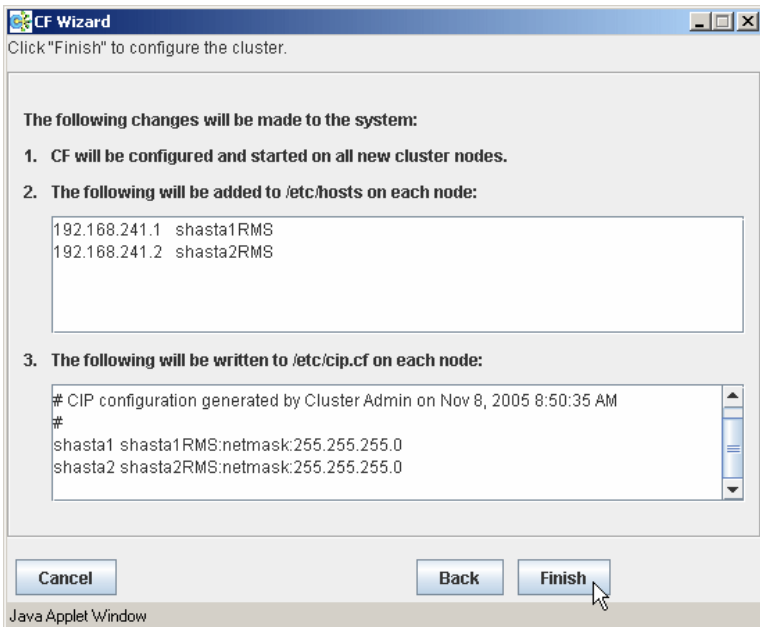


Figure 19: Summary window

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.

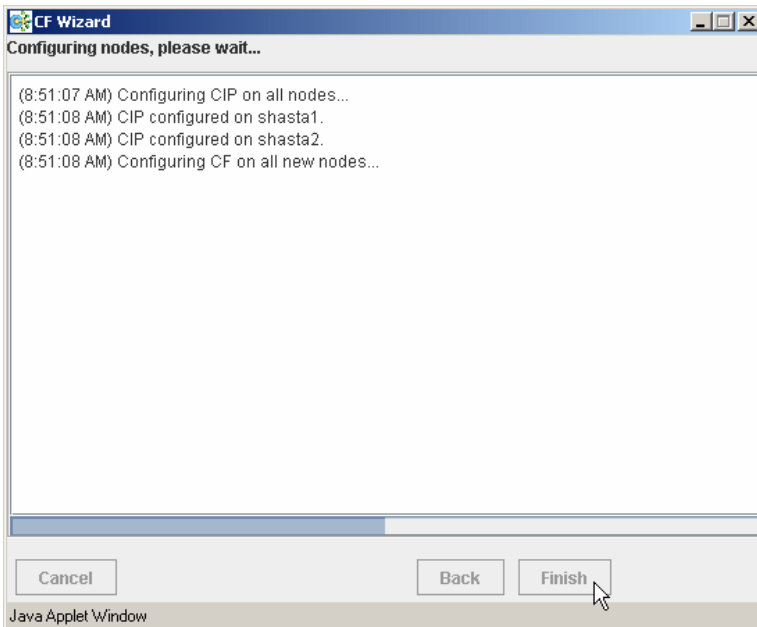


Figure 20: Configuration processing window

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

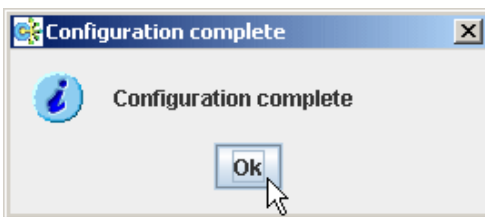


Figure 21: Configuration completion pop-up

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

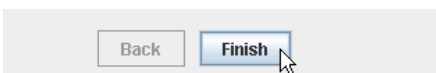


Figure 22: Configuration window after completion

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. If you have already decided the shutdown agent to be used, click on *yes* to run the SF Wizard.

If you have not decided the shutdown agent to be used, click on *no*. Then, see "5.1.2.1 Checking the Shutdown Agent Information" in the PRIMECLUSTER Installation and Administration Guide to decide it.

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).

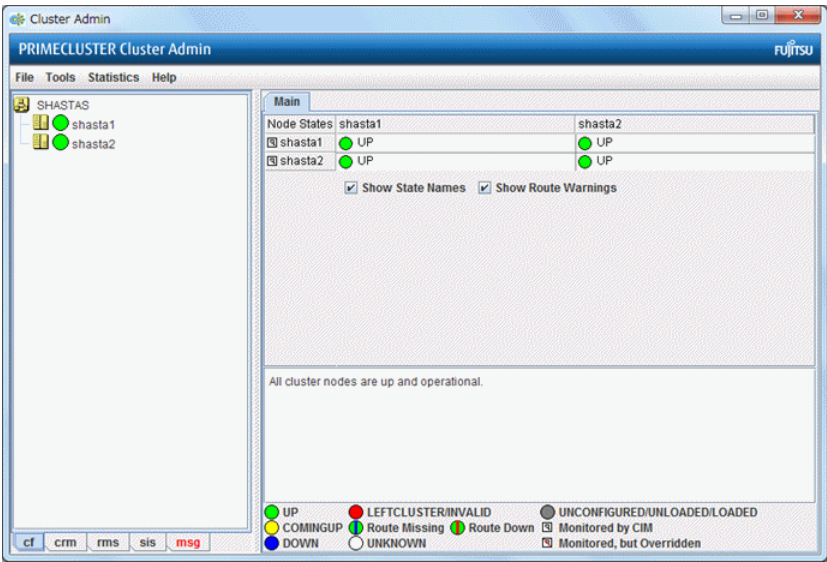


Figure 23: Main CF window

## 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 cfcf/cfsh.  
Specify /etc/default/cluster.config as below on all the nodes which configure the cluster system.

```
CFCF "cfcf"
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/cluster
# 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
```

If this command fails, check again that CF node names and cluster names configured in /etc/default/cluster in step 4 are correct.

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 again that CF node names, RMS node names, and CIP addresses configured in `/etc/cip.cf` and `/etc/hosts` in step 1 and step 2 are correct.

## 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 brackets "[", "]", ":", and "/".

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".

---

## 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 `rcquery(1M)` at any time. Refer to the Chapter "Manual pages" for complete details on the CLI options and arguments.

`rcquery(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.

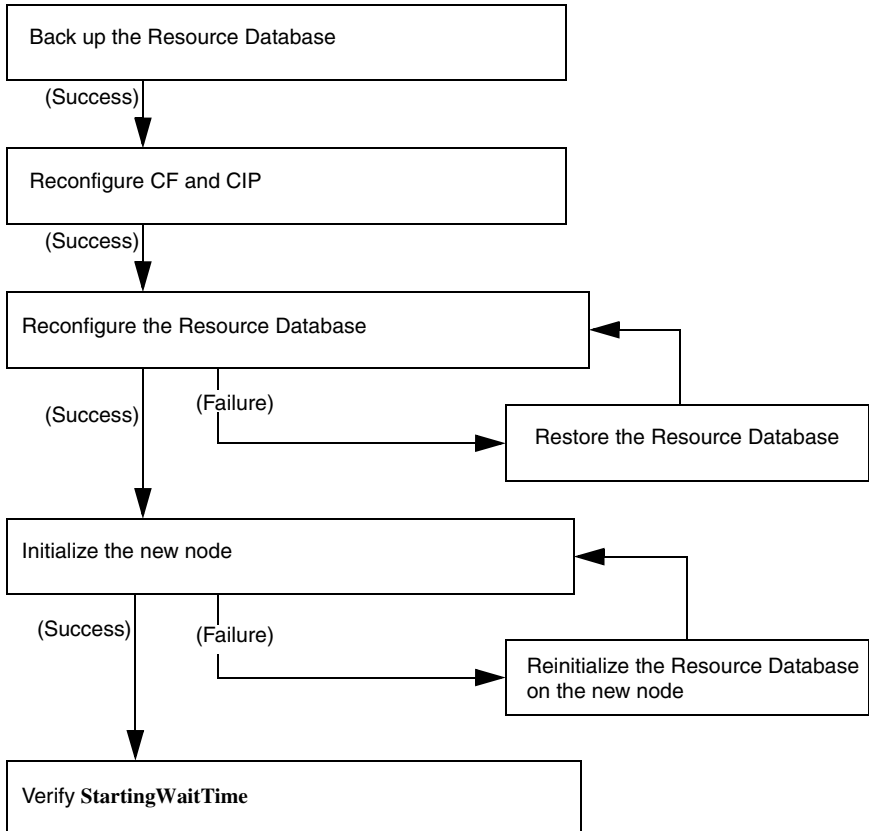


Figure 24: Adding a new node

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 complete, then you should execute the `clinitreset` command. 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).

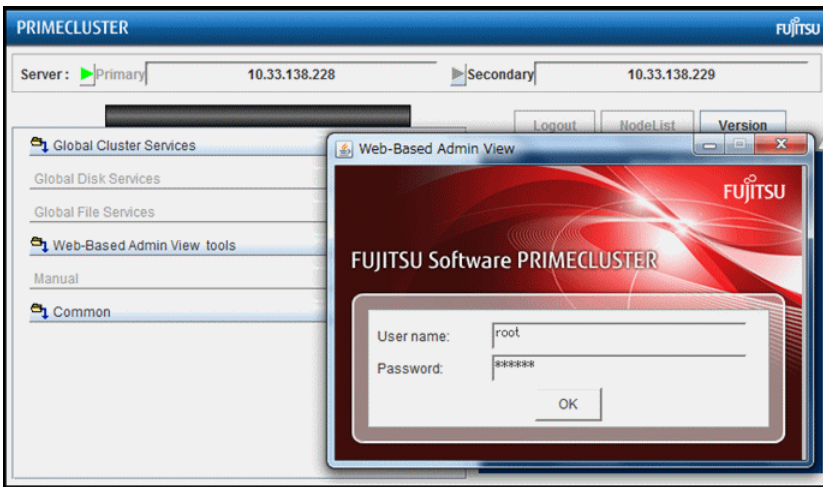


Figure 25: Cluster Admin start-up window

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).

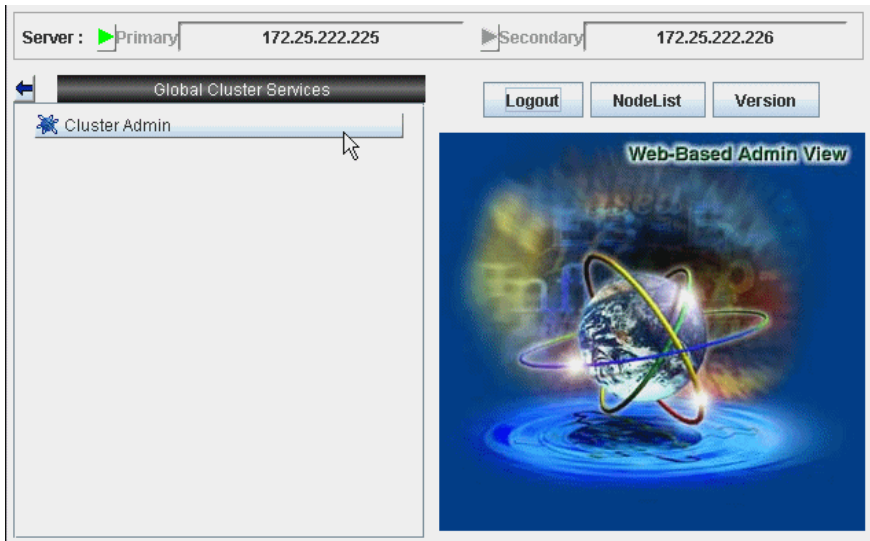


Figure 26: Cluster Admin top window

Click on the *Cluster Admin* button.

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

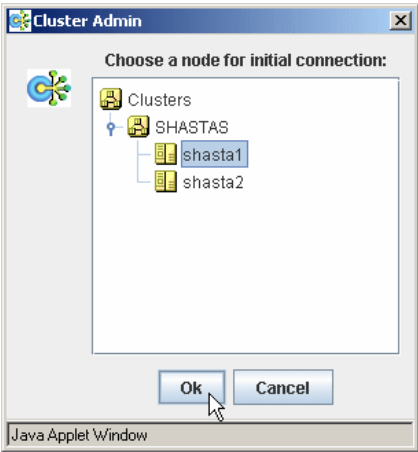


Figure 27: Initial connection choice window

Select a node and click on *Ok*.

The Cluster Admin main window appears (Figure 28).

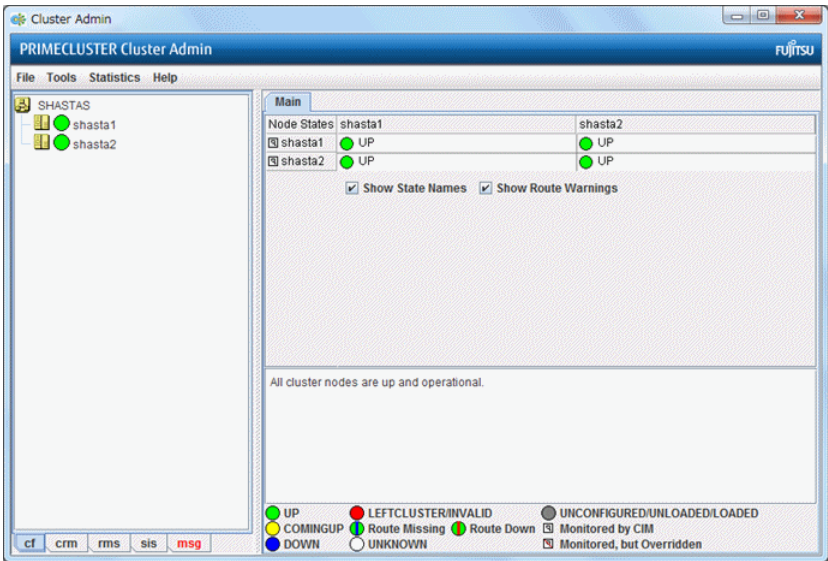


Figure 28: Cluster Admin main window

By default, the *cf* tab is selected and the CF main window is presented. Use the appropriate privilege level while logging in.

## 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).

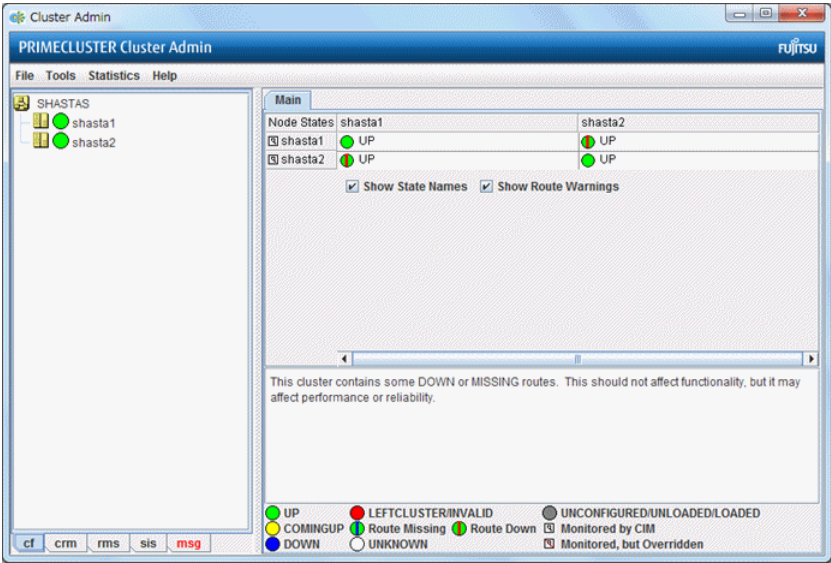


Figure 29: CF route DOWN

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).

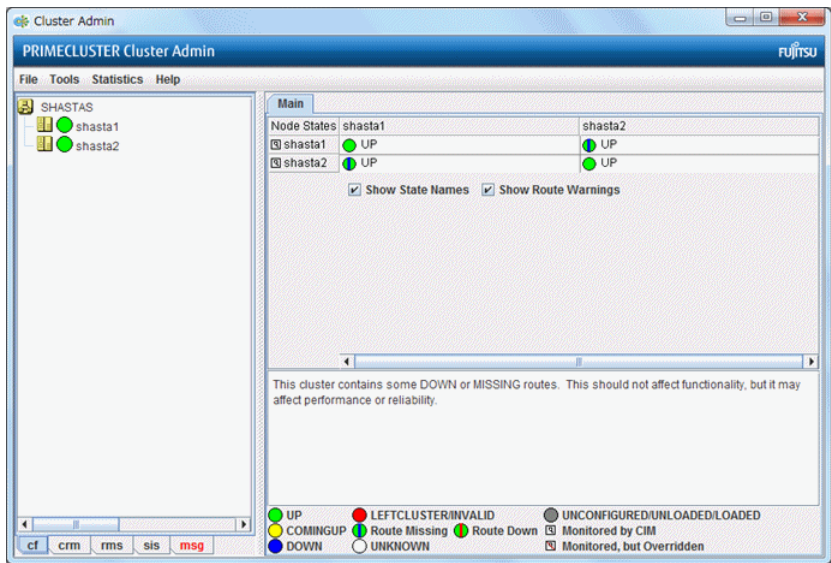


Figure 30: CF interface missing

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).

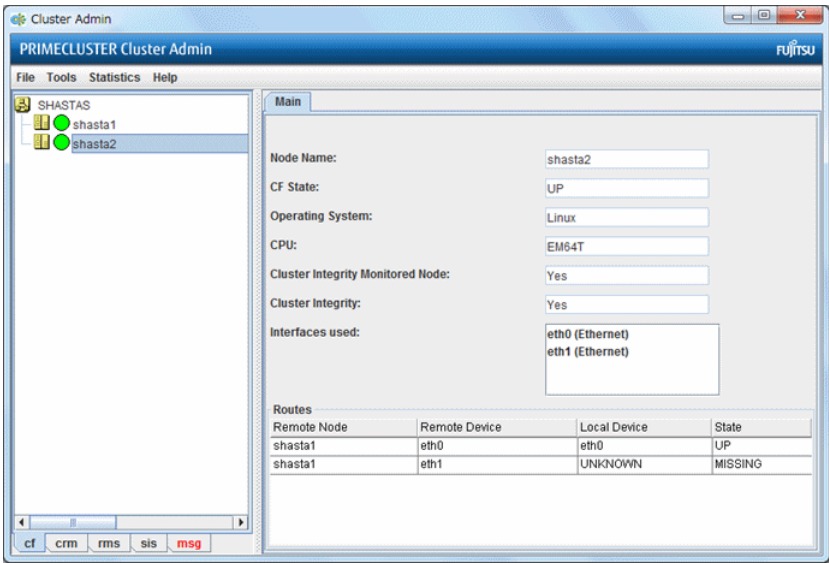


Figure 31: CF route table



## 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.

Main

Node Name:

shasta1

CF State:

UP

Operating System:

Linux

CPU:

Pentium

Cluster Integrity Monitored Node:

Yes

Cluster Integrity:

Yes

Interfaces used:

eth0 (Ethernet)  
eth2 (Ethernet)

Routes

Remote Node	Remote Device	Local Device	State
shasta2	eth0	eth0	UP
shasta2	eth2	eth2	UP

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.

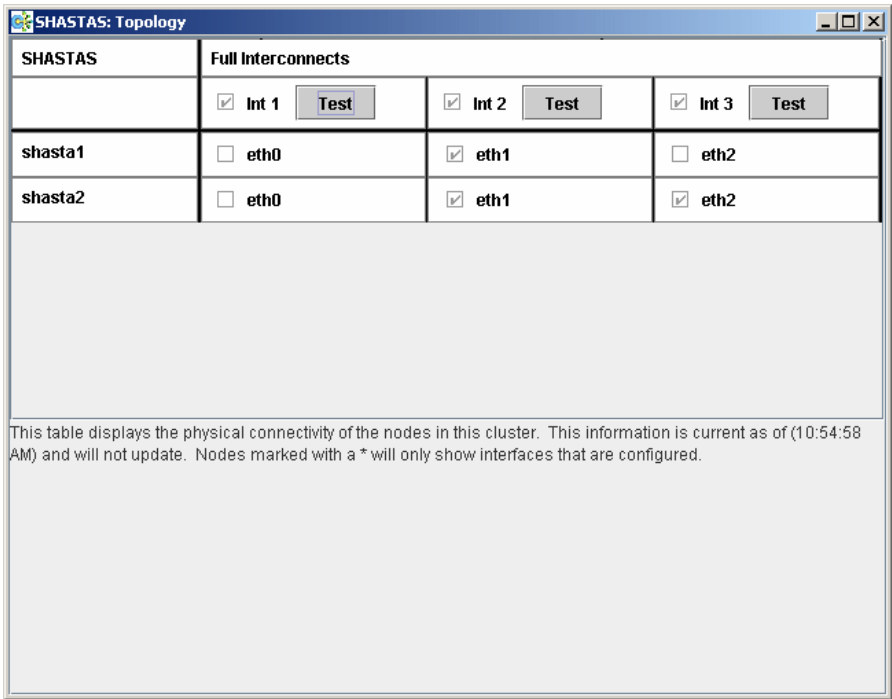


Figure 33: CF topology table

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).

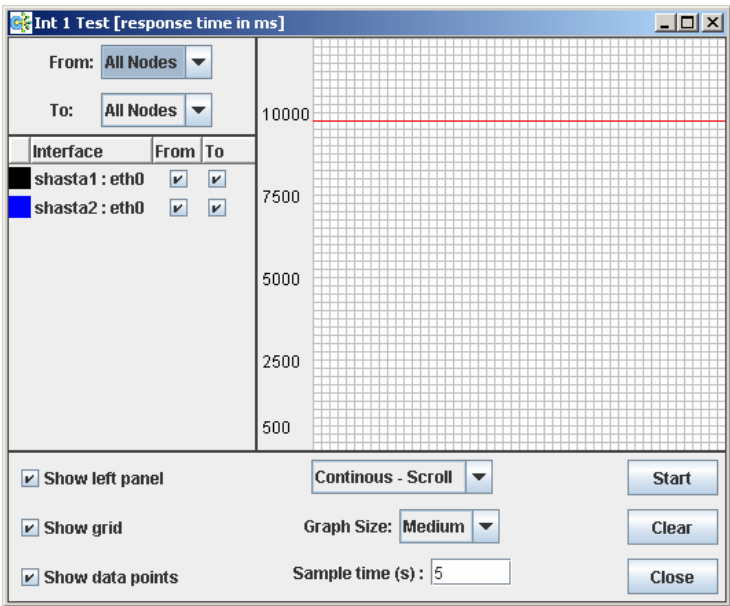


Figure 34: Response Time monitor

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*.

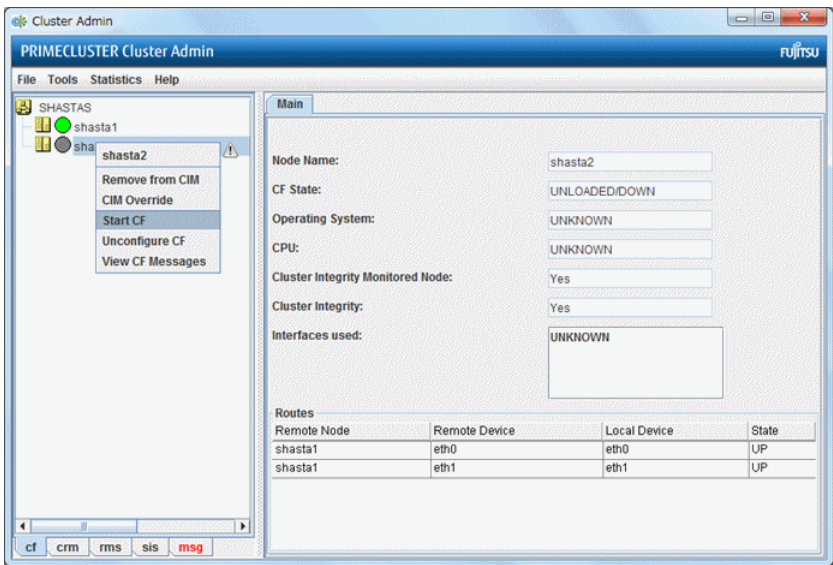


Figure 35: Starting CF

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.

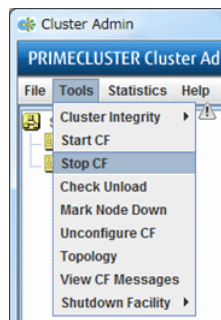


Figure 36: Starting and stopping CF from the Tools menu

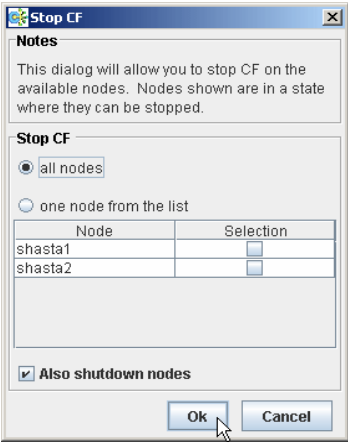


Figure 37: Stopping CF and shutting down all nodes

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.

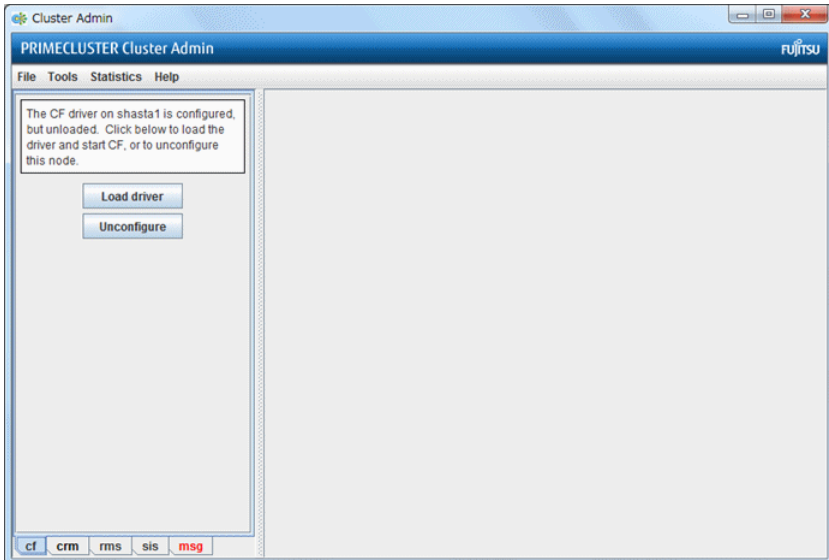


Figure 38: CF configured but not loaded

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.

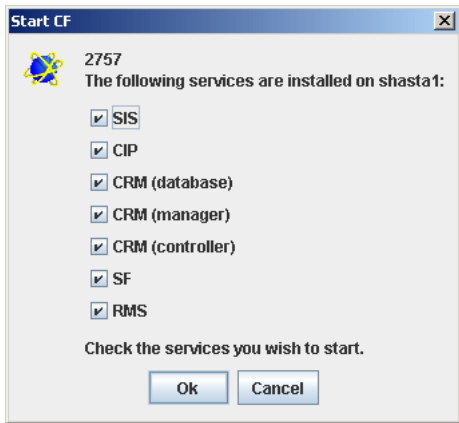


Figure 39: Start CF services pop-up

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).

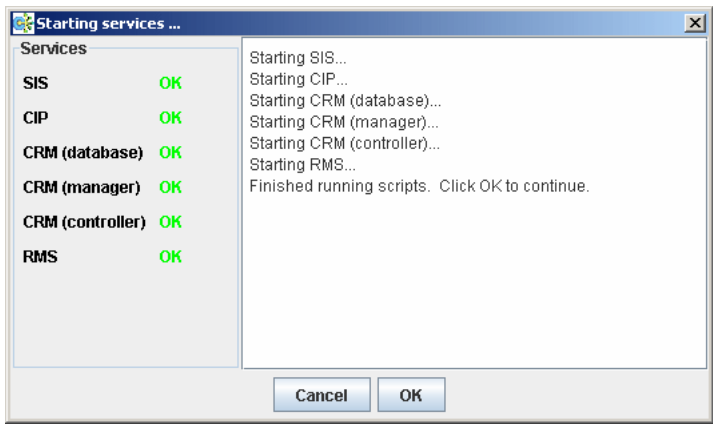


Figure 40: Start CF services status window

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).

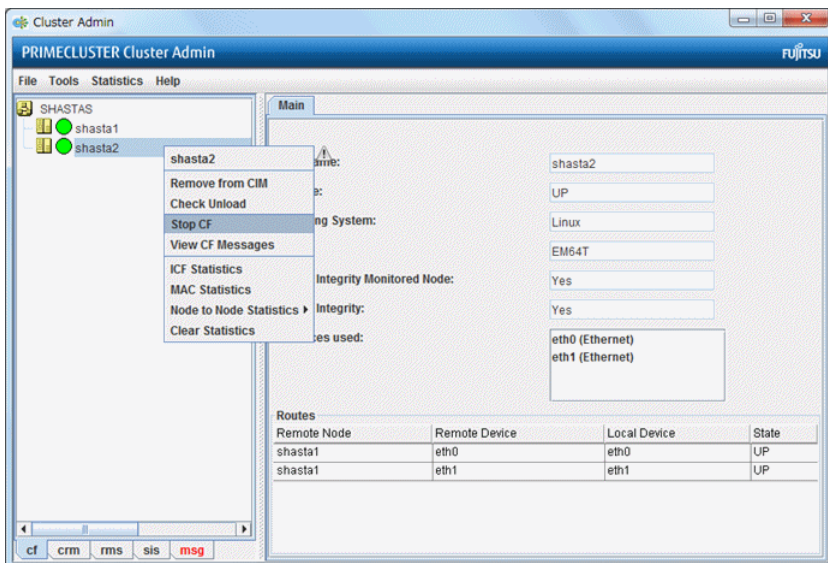


Figure 41: Stop CF

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

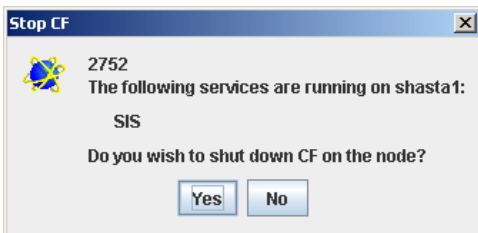


Figure 42: Stopping CF

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.

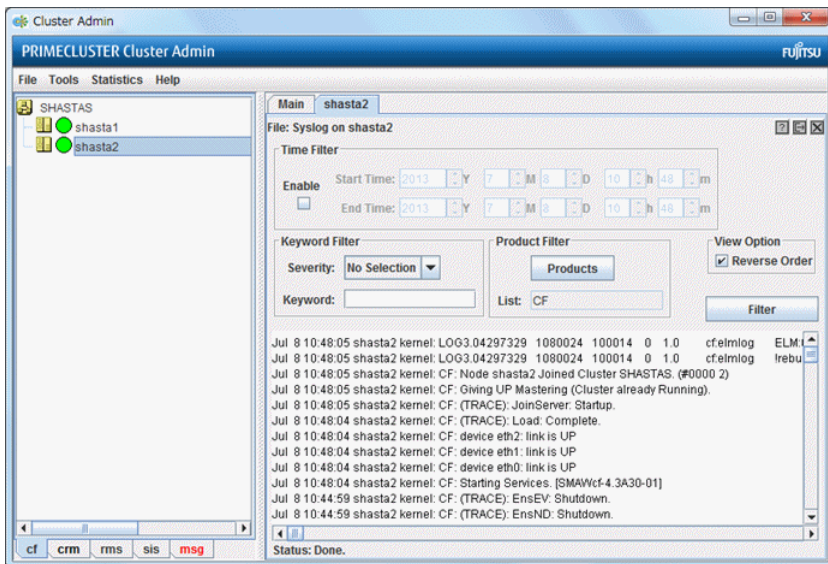


Figure 43: PRIMECLUSTER log viewer

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).

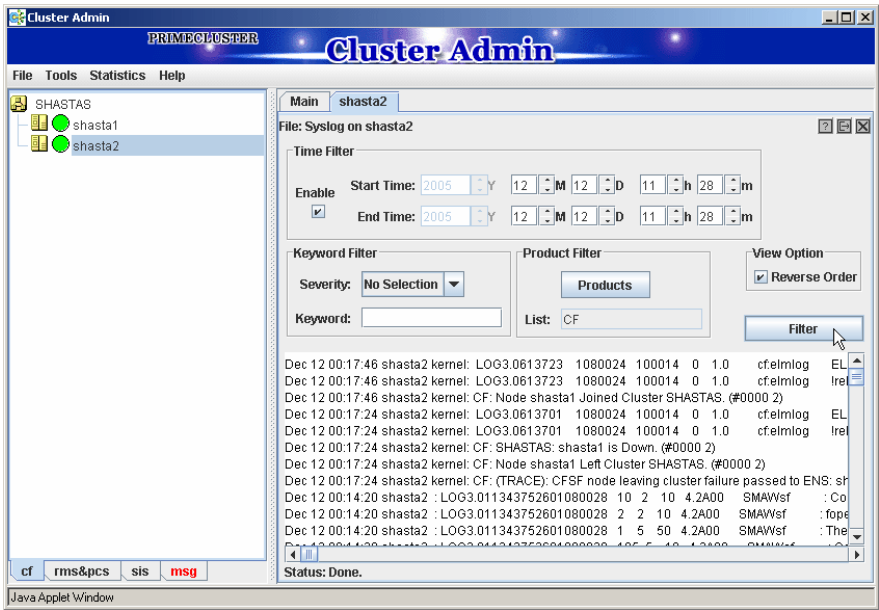


Figure 44: Search based on date/time

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).

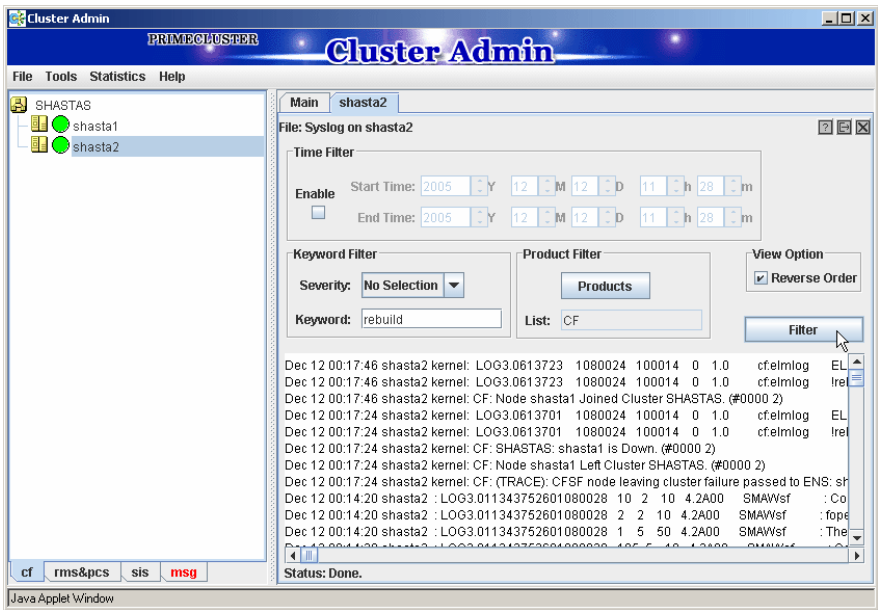


Figure 45: Search based on keyword

### 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.

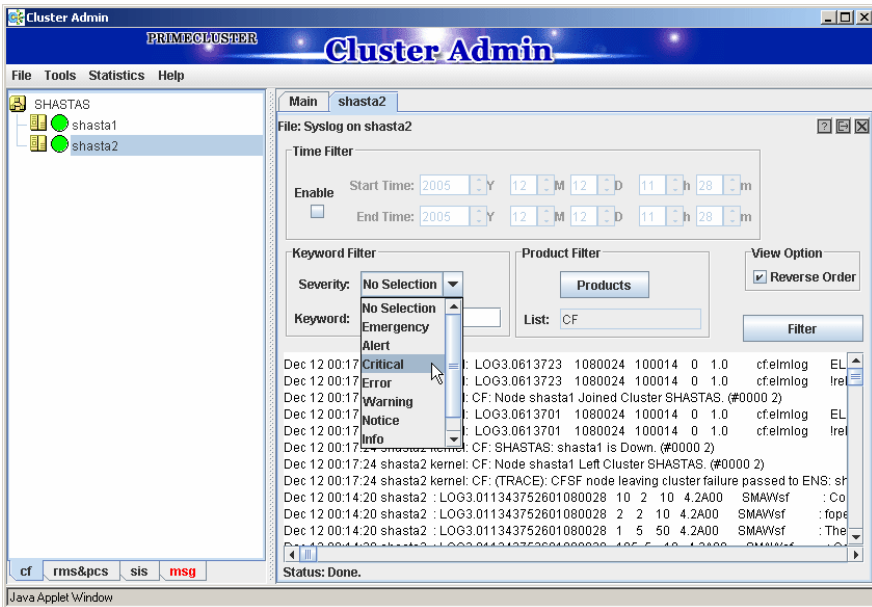


Figure 46: Search based on severity

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

Table 3: PRIMECLUSTER log viewer severity levels

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

Table 3: PRIMECLUSTER log viewer severity levels

## 5.10 Displaying statistics

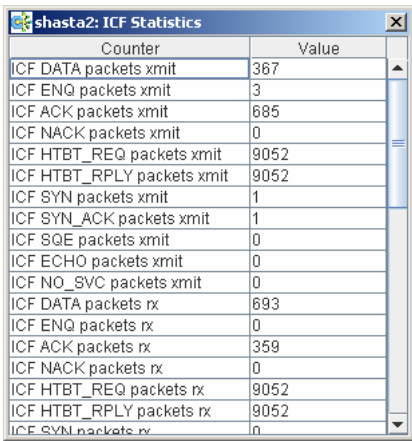
CF can display various statistics about its operation. There are three types of statistics available:

- 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.



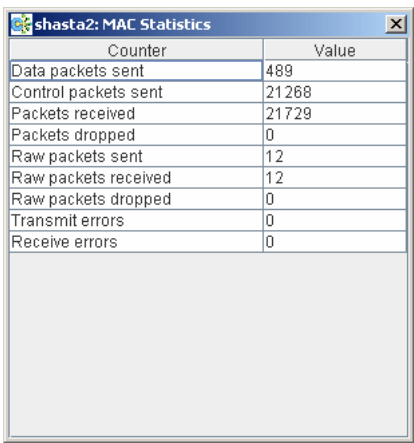
The screenshot shows a window titled "shasta2: ICF Statistics". Inside the window is a table with two columns: "Counter" and "Value". The table lists various network statistics for ICF, including transmit and receive counts for DATA, ENQ, ACK, NACK, HTBT\_REQ, HTBT\_RPLY, SYN, and SQE packets. The values are numerical counts.

Counter	Value
ICF DATA packets xmit	367
ICF ENQ packets xmit	3
ICF ACK packets xmit	685
ICF NACK packets xmit	0
ICF HTBT_REQ packets xmit	9052
ICF HTBT_RPLY packets xmit	9052
ICF SYN packets xmit	1
ICF SYN_ACK packets xmit	1
ICF SQE packets xmit	0
ICF ECHO packets xmit	0
ICF NO_SVC packets xmit	0
ICF DATA packets rx	693
ICF ENQ packets rx	0
ICF ACK packets rx	359
ICF NACK packets rx	0
ICF HTBT_REQ packets rx	9052
ICF HTBT_RPLY packets rx	9052
ICF SYN packets rx	0

Figure 47: ICF statistics



Figure 48 shows the display window for MAC Statistics.



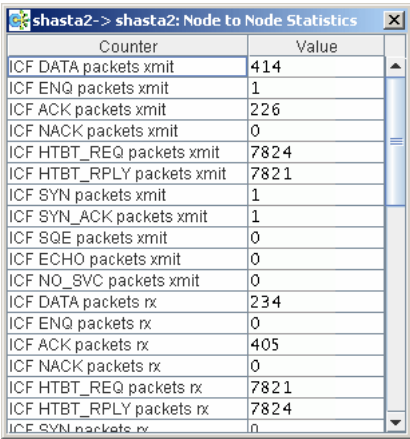
The screenshot shows a window titled "shasta2: MAC Statistics". Inside the window is a table with two columns: "Counter" and "Value". The table contains the following data:

Counter	Value
Data packets sent	489
Control packets sent	21268
Packets received	21729
Packets dropped	0
Raw packets sent	12
Raw packets received	12
Raw packets dropped	0
Transmit errors	0
Receive errors	0

Below the table is a large, empty rectangular area.

Figure 48: MAC statistics

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



The screenshot shows a window titled "shasta2-> shasta2: Node to Node Statistics". It contains a table with two columns: "Counter" and "Value". The table lists various network statistics for ICF (Inter-Cluster Fabric) packets, including DATA, ENQ, ACK, NACK, HTBT\_REQ, HTBT\_RPLY, SYN, and SYN\_ACK, for both transmit (xmit) and receive (rx) directions. The values are displayed in the "Value" column.

Counter	Value
ICF DATA packets xmit	414
ICF ENQ packets xmit	1
ICF ACK packets xmit	226
ICF NACK packets xmit	0
ICF HTBT_REQ packets xmit	7824
ICF HTBT_RPLY packets xmit	7821
ICF SYN packets xmit	1
ICF SYN_ACK packets xmit	1
ICF SQE packets xmit	0
ICF ECHO packets xmit	0
ICF NO_SVC packets xmit	0
ICF DATA packets rx	234
ICF ENQ packets rx	0
ICF ACK packets rx	405
ICF NACK packets rx	0
ICF HTBT_REQ packets rx	7821
ICF HTBT_RPLY packets rx	7824
ICF SYN packets rx	0

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).

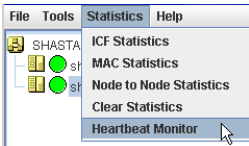


Figure 50: Selecting the Heartbeat monitor

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).

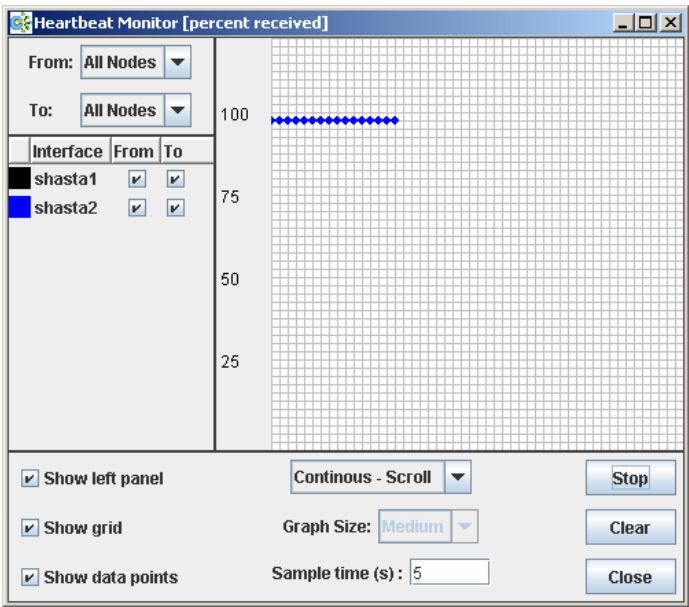


Figure 51: Heartbeat monitor

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).

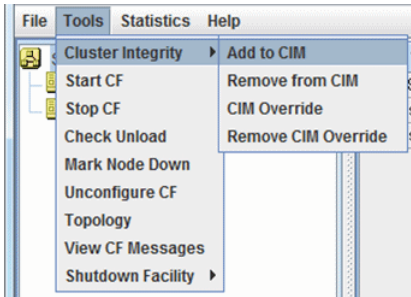


Figure 52: CIM options

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

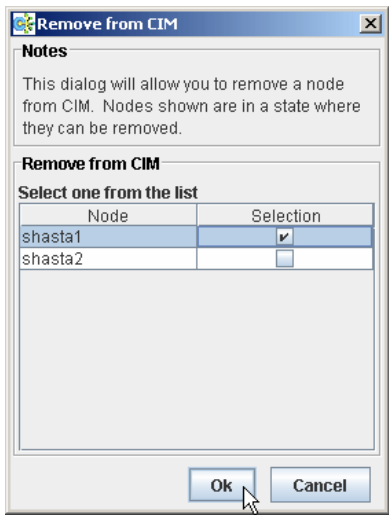


Figure 53: Add to CIM

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).

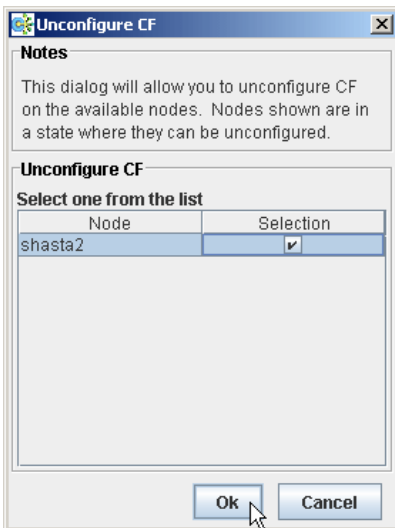


Figure 54: Unconfigure CF

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).

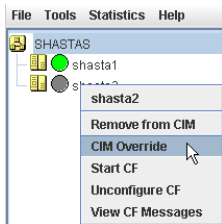


Figure 55: CIM Override

A confirmation pop-up appears (Figure 56).

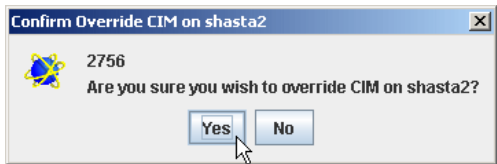


Figure 56: CIM Override confirmation

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).

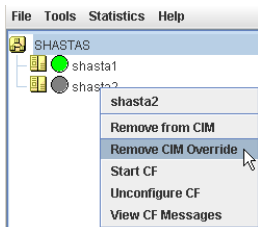


Figure 57: Remove CIM Override

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.

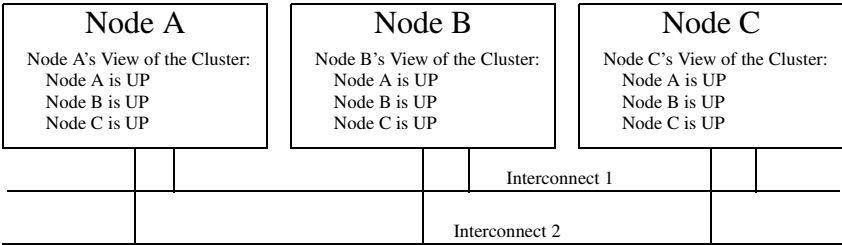


Figure 58: Three-node cluster with working connections

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.

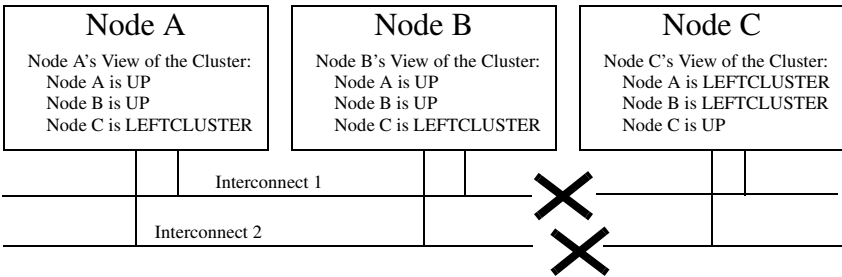


Figure 59: Three-node cluster where connection is lost

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.

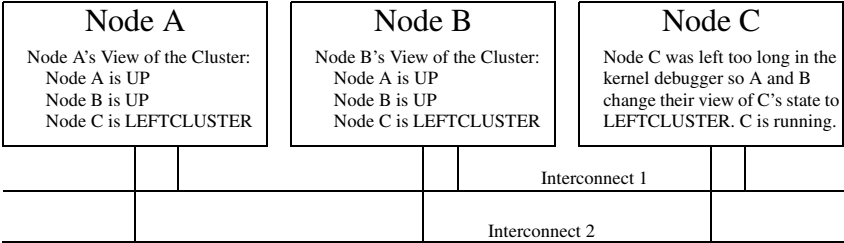


Figure 60: Node C placed in the kernel debugger too long

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.

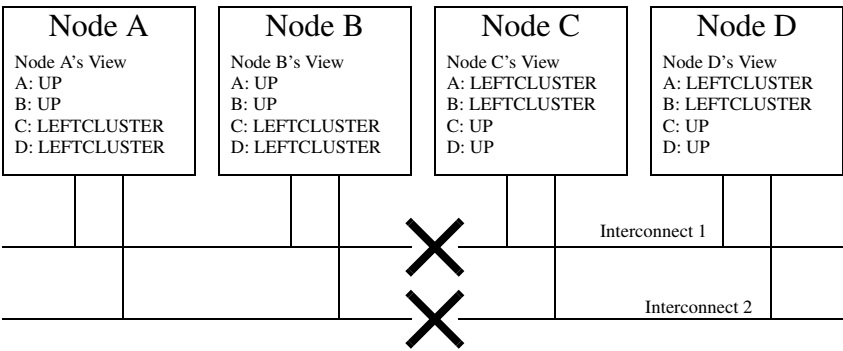


Figure 61: Four-node cluster with cluster partition

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 *ethn* 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
	<input checked="" type="checkbox"/> Int 1	<input checked="" type="checkbox"/> Int 2	<input type="checkbox"/> Int 3	<input type="checkbox"/> Int 4	
fuji2	<input checked="" type="checkbox"/> eth0 <input type="checkbox"/> eth2	<input checked="" type="checkbox"/> eth1	<input type="checkbox"/> eth3	<input type="checkbox"/> eth5	<input type="checkbox"/> eth4 <input type="checkbox"/> eth6
fuji3	<input checked="" type="checkbox"/> eth0	<input checked="" type="checkbox"/> eth2	missing	<input type="checkbox"/> eth1	
fuji4	<input checked="" type="checkbox"/> eth1	<input checked="" type="checkbox"/> eth2	<input type="checkbox"/> eth3	missing	<input type="checkbox"/> 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).

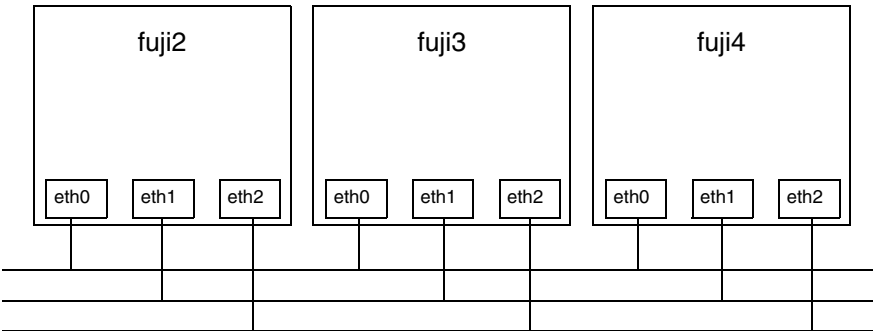


Figure 62: A three-node cluster with three full interconnects

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.

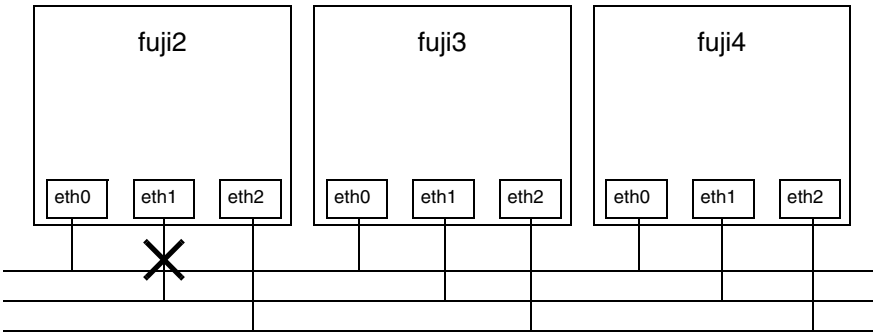


Figure 63: Broken ethernet connection for eth1 on fuji2

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.

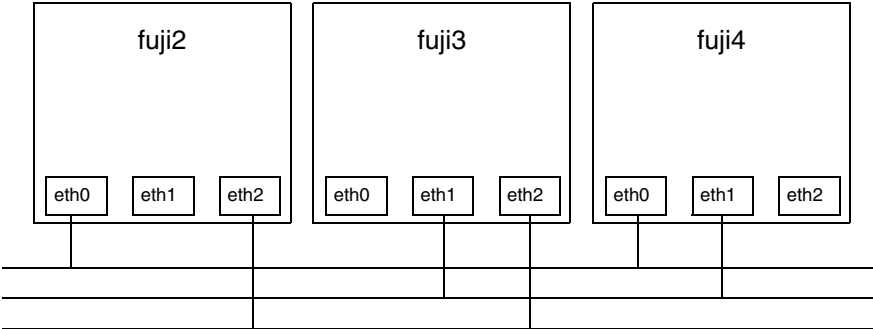


Figure 64: Cluster with no full interconnects

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 eliminating the nodes forcibly
- Resolving split-brain conditions
- Notifying other PRIMECLUSTER products that nodes were forcibly eliminated
- Checking the route that forcibly eliminates cluster nodes periodically (in 10-minute intervals)

The SD uses SAs to perform most of its work with regard to cluster node monitoring and forced node 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.

The SD starts SA periodically (in 10-minute intervals) to check the route that forcibly eliminates cluster nodes. The SD reflects the checked route status to the test status of each SA (Test State) displayed by the `sdtool(1M)` command.

## 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\_vmgrp 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\_vmchkhghost 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:

$$\text{timeout} = \text{boot\_delay} + (2 \times 2 \times \text{no\_of\_plugs}) + 10$$

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 unsupported 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\_mmbbr—Shutdown mechanism that resets nodes through the MMB.

The log file is stored in the following:

/var/opt/SMAWsf/log/SA\_mmbp.log

/var/opt/SMAWsf/log/SA\_mmbbr.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

The SF handles a split-brain by using the administrative LAN.

## 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 SMAW<sub>sf</sub> software.

To disable split-brain handling, 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

Split-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
- Specific Hardware Survival
- Specific Application Survival

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.

If no specific action was taken by the system administrator regarding split-brain resolution outcome from the values of both SF weight and RMS ShutdownPriority, the default "Largest Sub-cluster Survival" is selected.

## 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 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 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).

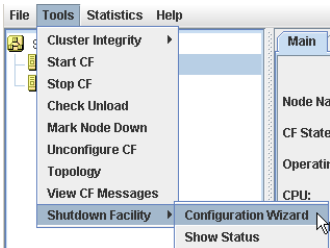


Figure 65: Starting the SF Configuration Wizard

Choose *Create* and click *Next*.

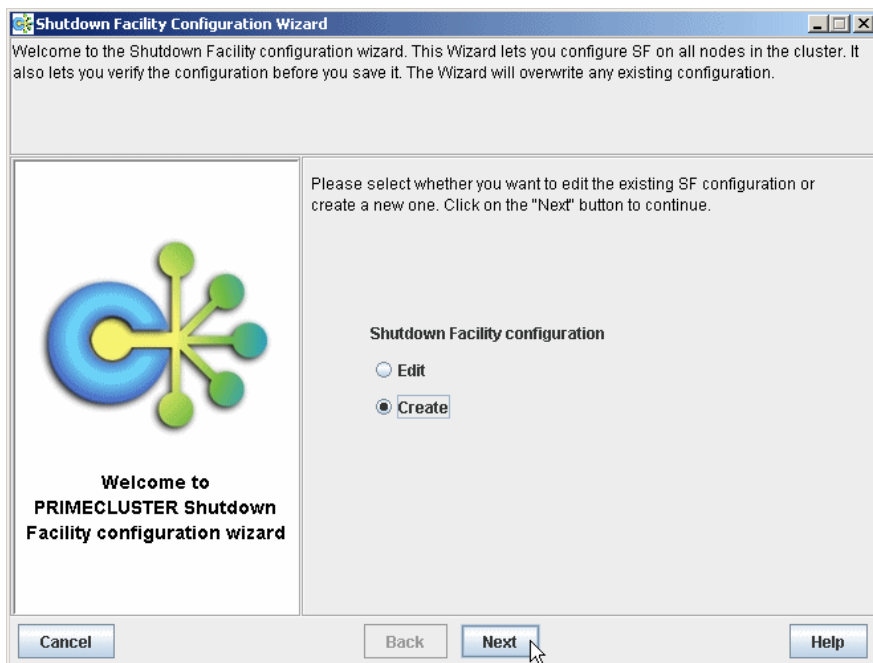


Figure 66: Creating the SF configuration

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*.

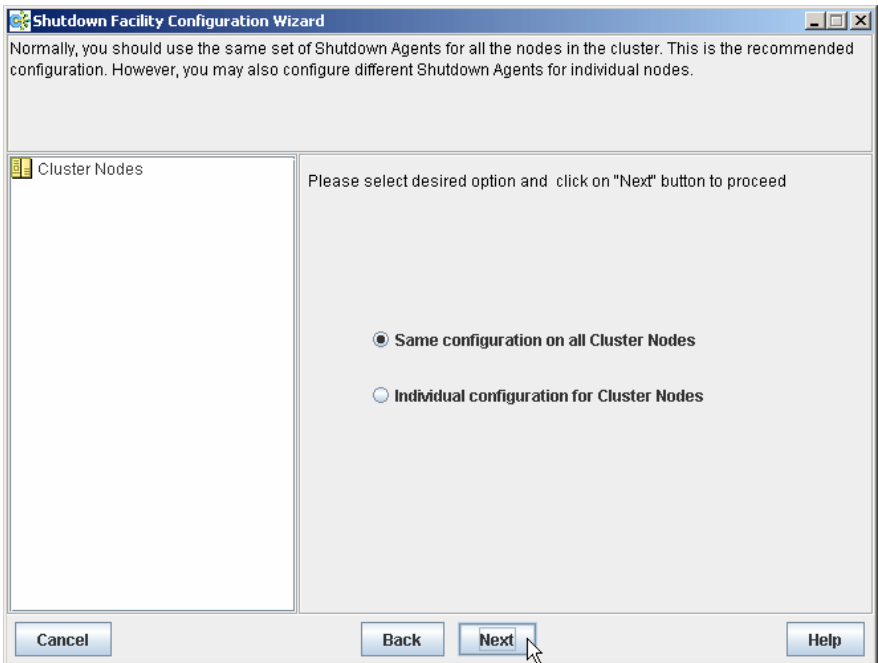


Figure 67: Choosing a common configuration for all nodes

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.

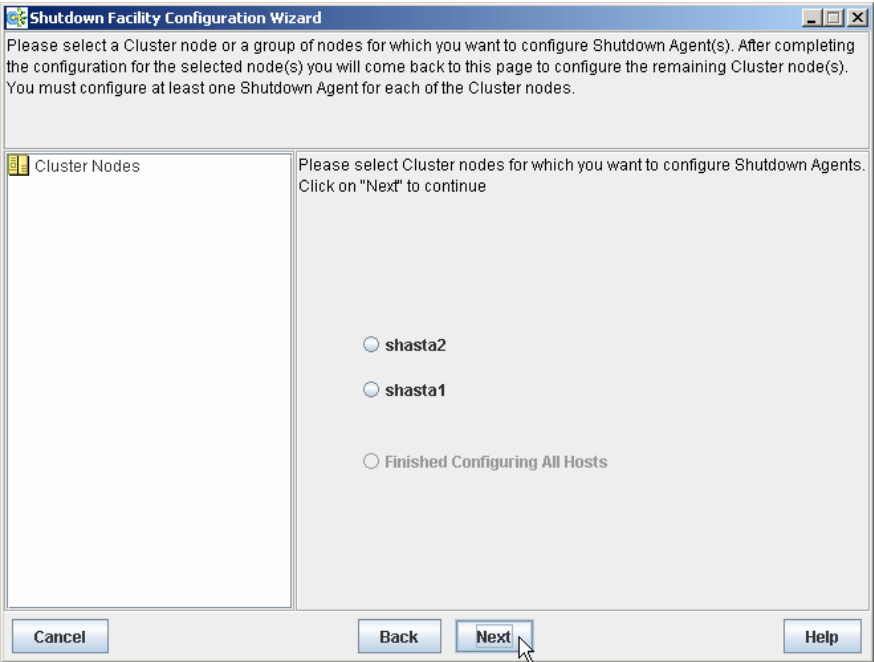


Figure 68: Selecting nodes to configure Shutdown Agents

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.

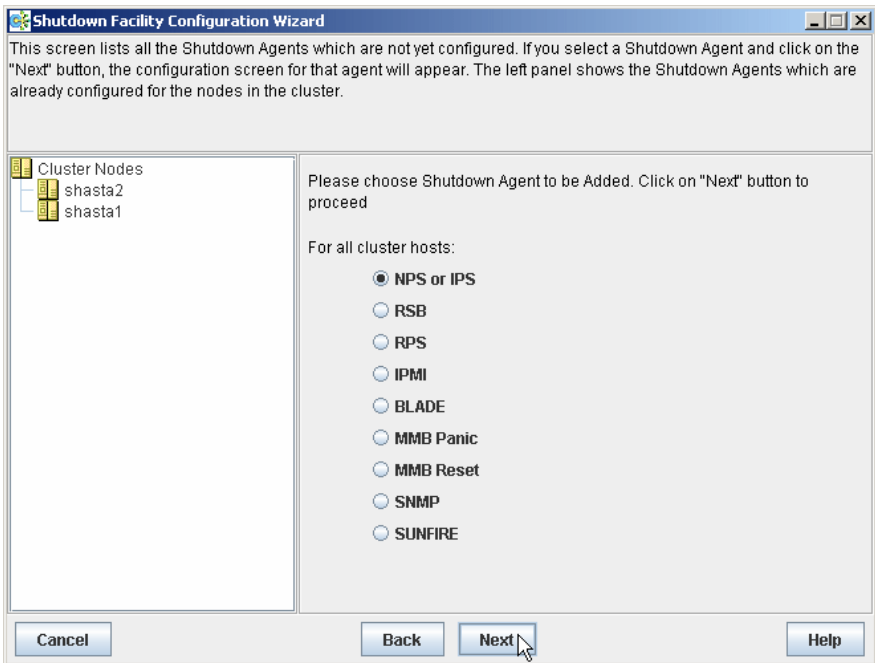


Figure 69: Choose Shutdown Agent to be added

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*.

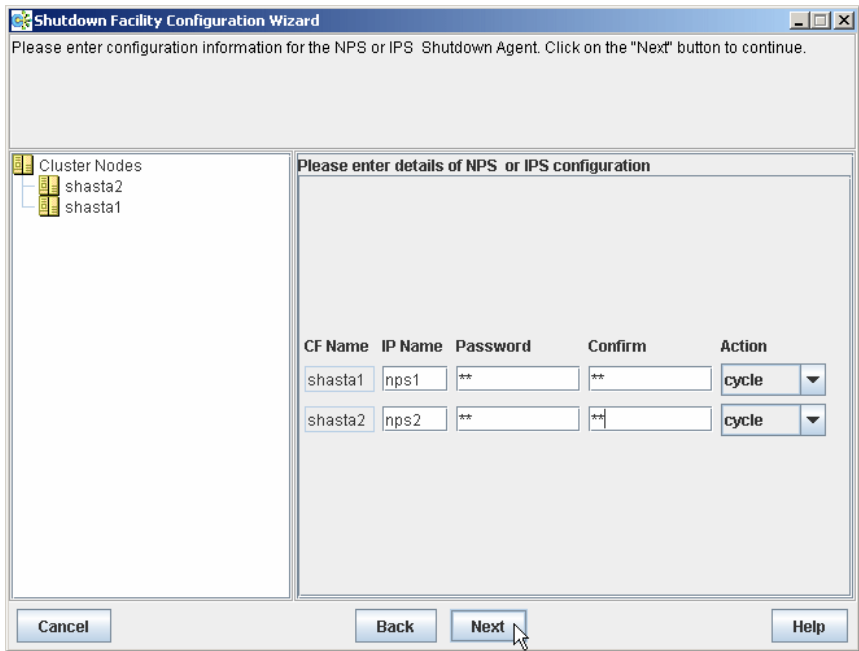


Figure 70: Configuring the IPS Shutdown Agent

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*.

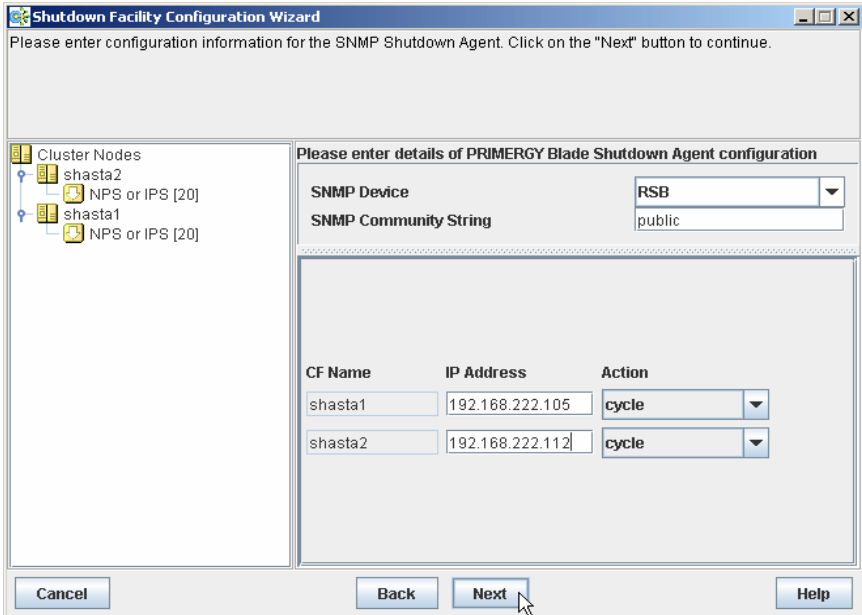


Figure 71: Selecting the SNMP Shutdown Agent options

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*.

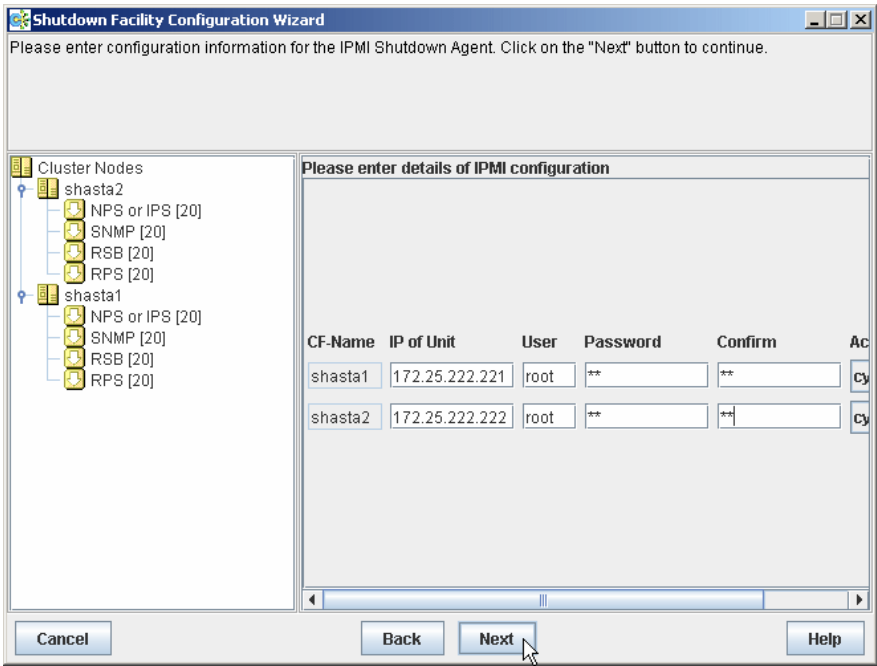


Figure 72: Configuring the IPMI Shutdown Agent

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

Shutdown Facility Configuration Wizard

Please enter configuration information for the PRIMERGY Blade Shutdown Agent. Click on the "Next" button to continue.

Cluster Nodes

- shasta2
  - NPS or IPS [20]
  - SNMP [20]
  - RSB [20]
  - RPS [20]
  - IPMI [20]
- shasta1
  - NPS or IPS [20]
  - SNMP [20]
  - RSB [20]
  - RPS [20]
  - IPMI [20]

Please enter details of PRIMERGY Blade Shutdown Agent configuration

Number of Management IPs: 1

Management Blade IP:

Management Blade Type: fsc

SNMP Community:

CF Name	Slot No.	Action
shasta1		cycle
shasta2		cycle

Cancel Back Next Help

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.

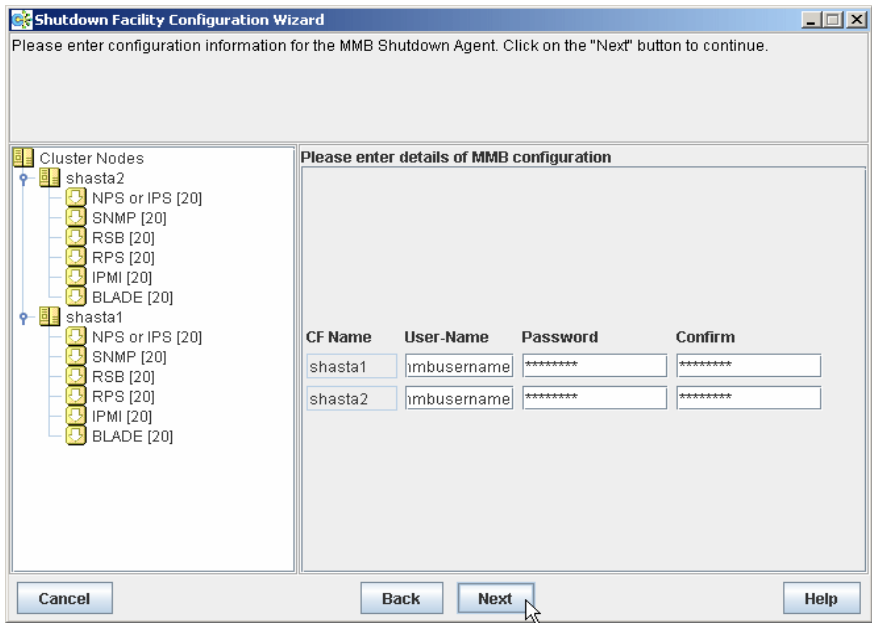


Figure 74: Configuring the MMB Shutdown Agents

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.

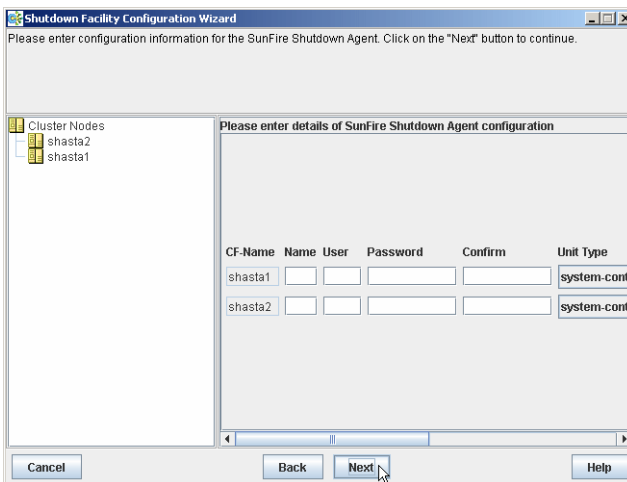


Figure 75: Configuring the SunFire Shutdown Agent

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.

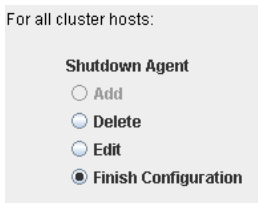


Figure 76: Adding, deleting, editing Shutdown Agents

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

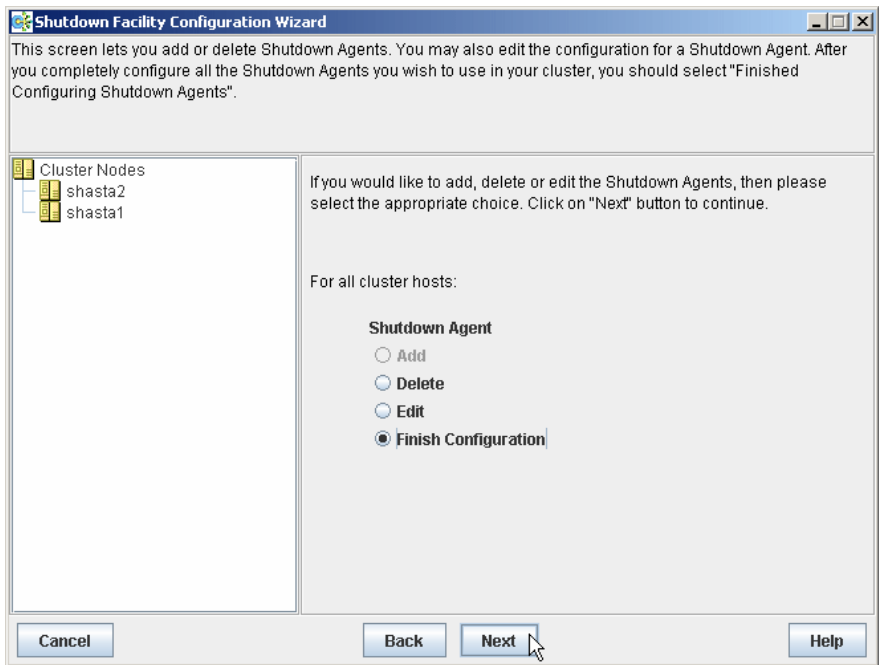


Figure 77: Finishing configuration



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*.

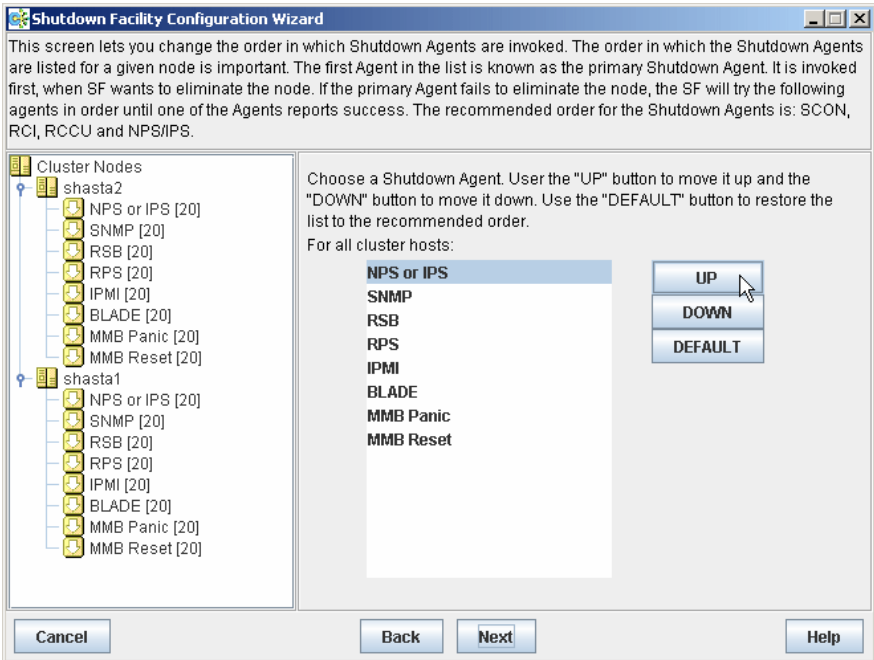


Figure 78: Changing the Shutdown Agent order

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.

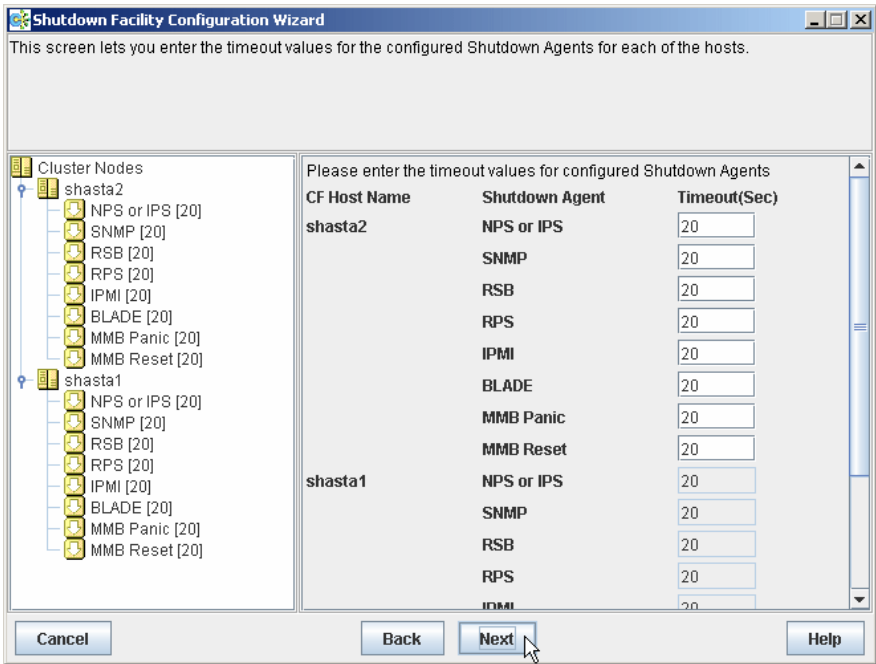


Figure 79: Specifying timeout values

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.

Shutdown Facility Configuration Wizard

This screen lets you enter relative weights for cluster hosts. These weights will be used to determine surviving sub-cluster in case of split brain scenario. Please enter values between 1 and 1000000 for each of the hosts. You should also enter the administrative IP address to be used for each of the cluster hosts.

Cluster Nodes

- shasta2
- shasta1

Please enter the Cluster Node weights and admin IPs

CF Host Name	Weight (>0)	Admin IP
shasta2	1	shasta2
shasta1	1	shasta1

Cancel Back Next Help

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 *nodeADM*; 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.

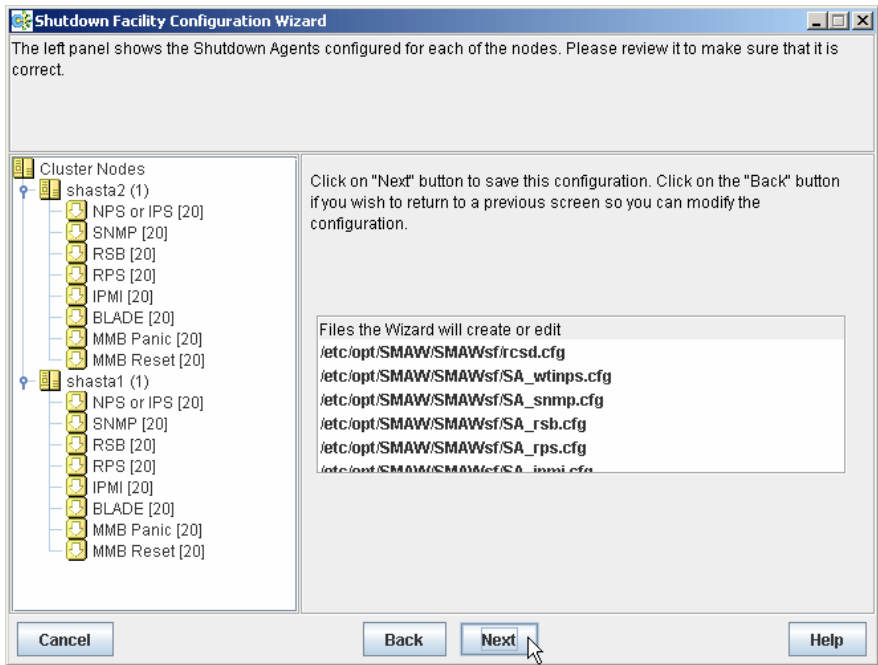


Figure 81: Confirming configuration file changes

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

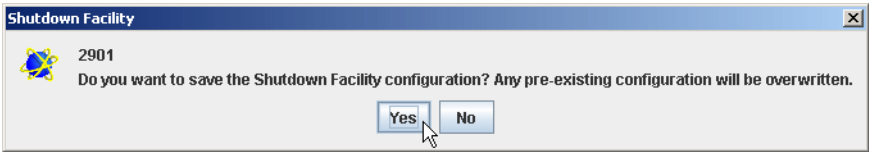


Figure 82: Saving SF configuration

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.

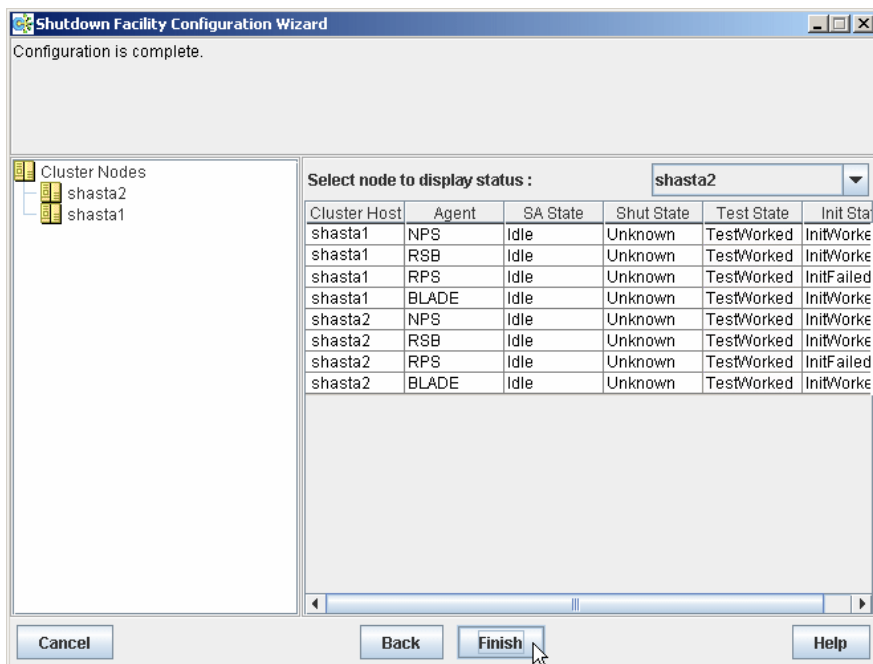


Figure 83: Status of Shutdown Agents

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/MAW/MAWsf/rcsd.cfg` on every node in the cluster.

A file, `rcsd.cfg.template`, is provided under the `/etc/opt/MAW/MAWsf` 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.
- *SAN* is the command name of a SA.
- *tN* is the timeout duration (seconds) of the SA.

The SA runs for the following cases:

- When a node is forcibly eliminated.
- When checking the connection to the option hardware used when a node is forcibly eliminated.

When a timeout occurs at the forced node elimination, the processing of the SA is stopped, and then the next SA is started. If all SAs fail to perform their processing, the node is left in the `LEFTCLUSTER` state.

When a timeout occurs when checking the connection to the option hardware used at the forced node elimination, the processing of the SA is stopped, and then the node becomes the `TestFailed` state.

- *wN* is the node weight.
- *admIPN* is the admin interface on the Administrative LAN on this cluster node.

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* is the IP name or address in dot notation of the Management Blade. Available IP addresses are IPv4 and IPv6 address. The link local address of IPv6 is not available.
- *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. The link local address of IPv6 is not available. When specifying the IPv6 address, enclose it in brackets "[ ]".

(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. The link local address of IPv6 is not available. When specifying the IPv6 address, enclose it in brackets "[ ]".  
(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/SMAwf/log/SA\_icmp.log

## VMCHKHOST

To configure the Shutdown Agent (SA\_vmchkhst) 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/SMAwf/SA\_vmchkhst.cfg

Templates of the configuration file are stored in the following:

/etc/opt/SMAW/SMAwf/SA\_vmchkhst.cfg.template

Each line of the configuration file SA\_vmchkhst 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. The link local address of IPv6 is 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\_vmchkhst configuration file:

fuij2 hostos2 192.168.200.1 FJSVvmSP 3CA1wxVXKD8a93077BaEkA==

fuij3 hostos3 192.168.200.2 FJSVvmSP 3CA1wxVXKD8a93077BaEkA==

The log file is stored in the following:

/var/opt/SMAWsf/log/SA\_vmchkhst.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/SMAwf/SA\_libvirtgp.cfg

/etc/opt/SMAW/SMAwf/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. The link local address of IPv6 is 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`.

8.5.2 Checking SA status

This section describes the procedures on how to check the current status of SA.

To check the current status of SA, execute the following command on the node where the status of SA is checked.

```
# sdtool -s
```

Example

```
# sdtool -s
Cluster Host Agent SA State Shut State Test State Init State
-----
node1      SA_xx Idle   Unknown   TestWorked InitWorked
node1      SA_xx Idle   Unknown   TestWorked InitWorked
node2      SA_xx Idle   Unknown   TestWorked InitWorked
node2      SA_xx Idle   Unknown   TestWorked InitWorked
```

The following items are displayed.

Cluster Host

Node name in the cluster

Agent

Name of the shutdown agent

SA State

Current status of SA.  
The status are as follows.

Status	Description
Idle	No currently running SA
Init-ing	Initializing
InitWorked	Initialization successfully completed
InitFailed	Initialization failed
Testing	Testing (10-minute interval)
TestWorked	Test successfully completed
TestFailed	Test failed
Killing	Force stop node ongoing
KillWorked	Force stop node successfully completed
KillFailed	Force stop node failed
UnInit-ing	Uninitializing
UnInitWorked	Uninitialization successfully completed
UnInitFailed	Uninitialization failed

Table 9: SA state

### Shut State

Status of force node stop.  
The status are as follows.

Status	Description
Unknown	Force stop node is not executed
Killing	Force stop node ongoing
KillWorked	Force stop node successfully completed
KillFailed	Force stop node failed

Table 10: Shut State

### Test State

Status of test executed every 10 minutes.  
The status are as follows.

Status	Description
Unknown	Test is not executed

Table 11: Test State

Status	Description
Testing	Testing
TestWorked	Test successfully completed
TestFailed	Test failed (force stop cannot be executed)

Table 11: Test State

Init State

Status of initialization/uninitialization of SA.

The status are as follows.

Status	Description
Unknown	Initialization is not executed
Init-ing	Initializing
InitWorked	Initialization successfully completed
InitFailed	Initialization failed
UnInit-ing	Uninitializing
UnInit-Failed	Uninitialization failed

Table 12: Init State

**Note**

- If Init State is displayed as InitFailed, it indicates that an error occurred in initializing the SA.
- If Test State is displayed as TestFailed, it indicates that an error occurred while the SA is testing if the node displayed as Cluster Host can be stopped, and the node cannot be stopped forcibly.  
In this case, an error may be occurring in software, hardware, or network resources that are used by the SA. Solve the error immediately.
- If Test State or Init State is displayed as Unknown, it indicates that SF has not yet stopped the node, tested the route, or initialized SA. Test State and Init State are temporary displayed as Unknown until the actual status is confirmed.
- If TestFailed or InitFailed is displayed, check the SA log file or /var/log/messages. In the log file, you can find the cause of the error why the SA testing or SA initialization failed. Once the cause of the error is solved and the cause of the error why the SA testing or SA initialization failed. Once the cause of the error is solved and SF is restarted, InitWorked or TestWorked will be displayed as the current status.

## 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 `rscd` 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 PRIME-CLUSTER 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 "4.5 Error Messages" in the *PRIMECLUSTER 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 "Chapter 5 CF Messages" in the *PRIMECLUSTER 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` or `pclsnap` 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` or `pclsnap`.

### 9.3.1 Executing the fjsnap or pcslsnap Command

The fjsnap or pcslsnap command is a tool which gathers system information necessary for analyzing the trouble at a time. When the trouble occurs by the PRIMECLUSTER system, the cause can be pursued by collecting necessary error information by the fjsnap or pcslsnap command.

You can execute this command as follows:

1. Log in with system administrator authority.
2. Execute the fjsnap or pcslsnap command.
  - For fjsnap  
`/usr/sbin/fjsnap -a output`
  - For pcslsnap  
`/opt/FJSVpcslsnap/bin/pcslsnap -a output`
    - The file name which becomes an output destination of system information collected by using the fjsnap or pcslsnap command for output is specified.
    - The following messages may be output to a switchlog and `/var/log/messages` when the fjsnap or pcslsnap command is executed while some nodes that consist of a cluster are suspended. However, no action is required for these messages.  
(BM, 8) Failed sending message `<message>` to object `<object>` on host `<host>`.  
(WRP, 11) Message send failed, queue id `<queueid>`, process `<process>`, `<name>`, to host `<node>`.

For details on the fjsnap command, see the README file included in the FJSVsnap package.

For details on the pcslsnap command, see the README file included in the FJSVpcslsnap package.

**Execution timings for the fjsnap or pclsnap command**

- For problems that occur during operation, for example, if an error message is output, execute the fjsnap or pclsnap command immediately after the problem occurs.
- If the fjsnap or pclsnap command cannot be executed because the system hangs, collect a crash dump. Then start the system in single user mode, and execute the fjsnap or pclsnap 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 or pclsnap command.
- If investigation information cannot be collected because the fjsnap or pclsnap command results in an error or the fjsnap or pclsnap command does not return, then collect a system dump.

### 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 Manual pages

This chapter lists the online manual pages for CCBR, CF, CIP, PAS, Resource Database, RMS, RMS Wizards, SF, and Web-Based Admin View.

To display a manual page, type the following command:

```
$ man man_page_name
```

### 10.1 CCBR

#### **System administration**

**cfbackup**

save the cluster configuration information for a PRIMECLUSTER node

**cfrestore**

restore saved cluster configuration formation on a PRIMECLUSTER node

### 10.2 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

**rcquery**

get quorum state of the cluster

## 10.3 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

## 10.4 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

## 10.5 PAS

### System administration

mipcstat

MIPC statistics

clmstat

CLM statistics



## 10.6 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

`clstartsrc`

activate resource (only GDS)

`clstopsrc`

deactivate resource (only GDS)

`clsyncfile`

distribute a file between cluster nodes

### User command

`clgettree`

display the tree information of the resource database

## 10.7 RMS

### System administration

hvassert

assert (test for) an RMS resource state

hvcn

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

hvutil

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

## 10.8 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

## 10.9 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)

## 10.10 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

---

## 11 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_vmchkhos" 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_vmchkhos.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 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.
17	4.3A30	February 2014	"CF, CIP, and CIM configuration" CF over IP	Deleted descriptions for CF over IP.

No	VL	Edition	Section	Description
18	4.3A30	February 2014	"CF, CIP, and CIM configuration"	Added Note "Restriction on the starting and stopping of GLS, and the rebooting for network service of System."
19	4.3A30	February 2014	"Differences between CIP and CF over IP"	Added notes that CF over IP is not supported.
20	4.3A30	February 2014	"Example of creating a cluster"	Changed descriptions when the pop-up window of the configuration window after completion is displayed.
21	4.3A30	February 2014	Cluster Configuration Backup and Restore (CCBR)	Deleted descriptions for CCBR.
22	4.3A30	February 2014	"Overview"	Added that the SD checks cluster nodes to be forcibly eliminated to the function of Shutdown Daemon.
23	4.3A30	February 2014	"MMB"	Deleted descriptions for the environment configuration.
24	4.3A30	February 2014	"Administrative LAN"	Changed descriptions for the administrative LAN.
25	4.3A30	February 2014	"Executing the fjsnap or pclsnap Command"	Changed descriptions for the procedure of information collecting.
26	4.3A30	February 2014	"RMS"	Deleted descriptions for the hvattr command.
27	4.3A30	February 2014	CF messages and codes	Deleted whole chapter.





---

# Glossary

Items in this glossary that apply to specific Cluster Foundation components are indicated with the following notation:

- (CF)-Cluster Foundation
- (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 the Wizard Tools or the RMS 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, and RMS. Cluster Admin is available from the Web-Based Admin View interface.

See also *Cluster Foundation (CF)*, *Scalable Internet Services (SIS)*, *Reliant Monitor Services (RMS)*, *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 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 RMS Wizard Tools, the RMS base monitor, and RMS detectors each maintain their own log files.

**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)*.

### **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 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.

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 *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

## Abbreviations

---

### **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

Figure 1:	CIP diagram . . . . .	13
Figure 2:	CF over IP diagram . . . . .	15
Figure 3:	Login pop-up . . . . .	21
Figure 4:	Main Web-Based Admin View window after login . . . . .	22
Figure 5:	Global Cluster Services window in Web-Based Admin View . . . . .	22
Figure 6:	Initial connection pop-up . . . . .	23
Figure 7:	CF is unconfigured and unloaded . . . . .	23
Figure 8:	CF loaded but not configured . . . . .	24
Figure 9:	Creating or joining a cluster . . . . .	25
Figure 10:	Adding a local node to a CF cluster not shown . . . . .	25
Figure 11:	Selecting cluster nodes and the cluster name . . . . .	26
Figure 12:	CF loads and pings . . . . .	27
Figure 13:	Edit CF node names . . . . .	28
Figure 14:	CF topology and connection table . . . . .	29
Figure 15:	CF over IP window . . . . .	31
Figure 16:	CIP wizard (IPv4) window . . . . .	33
Figure 17:	CIP wizard (IPv6) window . . . . .	34
Figure 18:	CIM configuration window . . . . .	36
Figure 19:	Summary window . . . . .	38
Figure 20:	Configuration processing window . . . . .	39
Figure 21:	Configuration completion pop-up . . . . .	39
Figure 22:	Configuration window after completion . . . . .	39
Figure 23:	Main CF window . . . . .	40
Figure 24:	Adding a new node . . . . .	59

## Figures

---

Figure 25: Cluster Admin start-up window . . . . .	66
Figure 26: Cluster Admin top window . . . . .	67
Figure 27: Initial connection choice window . . . . .	68
Figure 28: Cluster Admin main window . . . . .	68
Figure 29: CF route DOWN . . . . .	72
Figure 30: CF interface missing . . . . .	73
Figure 31: CF route table . . . . .	74
Figure 32: CF node information . . . . .	75
Figure 33: CF topology table . . . . .	76
Figure 34: Response Time monitor . . . . .	77
Figure 35: Starting CF . . . . .	79
Figure 36: Starting and stopping CF from the Tools menu . . . . .	79
Figure 37: Stopping CF and shutting down all nodes . . . . .	80
Figure 38: CF configured but not loaded . . . . .	81
Figure 39: Start CF services pop-up . . . . .	82
Figure 40: Start CF services status window . . . . .	82
Figure 41: Stop CF . . . . .	83
Figure 42: Stopping CF . . . . .	83
Figure 43: PRIMECLUSTER log viewer . . . . .	85
Figure 44: Search based on date/time . . . . .	86
Figure 45: Search based on keyword . . . . .	87
Figure 46: Search based on severity . . . . .	88
Figure 47: ICF statistics . . . . .	90
Figure 48: MAC statistics . . . . .	91
Figure 49: Node to Node statistics . . . . .	92
Figure 50: Selecting the Heartbeat monitor . . . . .	93



Figure 51: Heartbeat monitor .....	93
Figure 52: CIM options .....	95
Figure 53: Add to CIM .....	96
Figure 54: Unconfigure CF .....	97
Figure 55: CIM Override .....	98
Figure 56: CIM Override confirmation .....	98
Figure 57: Remove CIM Override .....	98
Figure 58: Three-node cluster with working connections .....	100
Figure 59: Three-node cluster where connection is lost .....	100
Figure 60: Node C placed in the kernel debugger too long .....	104
Figure 61: Four-node cluster with cluster partition .....	105
Figure 62: A three-node cluster with three full interconnects .....	110
Figure 63: Broken ethernet connection for eth1 on fuji2 .....	111
Figure 64: Cluster with no full interconnects .....	112
Figure 65: Starting the SF Configuration Wizard .....	127
Figure 66: Creating the SF configuration .....	128
Figure 67: Choosing a common configuration for all nodes .....	129
Figure 68: Selecting nodes to configure Shutdown Agents .....	130
Figure 69: Choose Shutdown Agent to be added .....	131
Figure 70: Configuring the IPS Shutdown Agent .....	132
Figure 71: Selecting the SNMP Shutdown Agent options .....	133
Figure 72: Configuring the IPMI Shutdown Agent .....	134
Figure 73: Configuring the Blade Shutdown Agent .....	135
Figure 74: Configuring the MMB Shutdown Agents .....	136
Figure 75: Configuring the SunFire Shutdown Agent .....	137
Figure 76: Adding, deleting, editing Shutdown Agents .....	138

**Figures**

---

Figure 77: Finishing configuration ..... 138

Figure 78: Changing the Shutdown Agent order ..... 139

Figure 79: Specifying timeout values ..... 140

Figure 80: Entering node weights and administrative IP addresses .... 141

Figure 81: Confirming configuration file changes ..... 142

Figure 82: Saving SF configuration ..... 143

Figure 83: Status of Shutdown Agents ..... 144

Figure 84: Exiting SF configuration wizard ..... 145

# Tables

Table 1: Local states ..... 70

Table 2: Remote states ..... 71

Table 3: PRIMECLUSTER log viewer severity levels..... 88

Table 4: Basic layout for the CF topology table ..... 108

Table 5: Topology table with check boxes shown ..... 109

Table 6: Topology table for 3 full interconnects ..... 111

Table 7: Topology table with broken Ethernet connection ..... 111

Table 8: Topology table with no full interconnects ..... 112

Table 9: SA state ..... 159

Table 10: Shut State . . . . . 159

Table 11: Test State ..... 159

Table 12: Init State ..... 160



---

# Index

## Symbols

/etc/cip.cf 54

/etc/hosts

    CIP configuration 10

    CIP Wizard 36

## A

adding

    new node 54

    nodes 26

    to CIM 96

applets, trusted 20

## B

backing up

    Resource Database 60

broadcast messages 14

broken interconnects 99

## C

CCBR commands

    cfbackup 177

    cfrestore 177

CF

*See also* Cluster Foundation

CF commands

    cfconfig 177

    cfregd 177

    cfset 177

    cftool 177

    rcqconfig 177

    rcquery 177

CF driver 24

CF over IP 13

    scenarios 14

CF Registry

    cfregd 45

    user-level daemon 45

CF Remote Services 37

CF Wizard

    bringing up 24

    edit node names 28

    new cluster 26

    running 41

    summary window 38

CF/CIP Wizard, starting 10

cfbackup 177

cfconfig 177

CFCP 16

cfcp 18, 37

CFReg 49

cfrestore 177

cfset 15, 177

    CFCP 16

    CFSH 16

    CLUSTER\_TIMEOUT 16

    maximum entries 16

    options 16

    tune timeout 17

CFSH 16

cfsh 37

cftool 177

cftool -n 99

CIM

*See* Cluster Integrity Monitor

CIP

*See* Cluster Interconnect Protocol

CIP commands

    cip.cf 178

    cipconfig 178

    ciptool 178

CIP Wizard

    /etc/hosts 36

    CIP interface 34, 35

    Cluster Admin 10

    configuration file 36

    numbering 35

    screen 33

    starting 10

cip.cf 43, 44, 178

cipconfig 178

ciptool 178

- clautoconfig 179
- clbackuprdb 60, 179
- cldevice 179
- cldevparam 178
- clexec 179
- clgettree 55, 56, 62, 63, 179
  - output 55
  - verify configuration 56
- CLI
  - See* Command Line Interface
- clinitreset 55, 56, 63, 179
- clmmbmonctl 178
- clmmbsetup 178
- clmtest 178
- clrestorerdb 64, 179
- clroot 21
- clsetparam 57, 179
- clsetup 55, 56, 61, 62, 63, 179
- clstartsrc 179
- clstopsrc 179
- clsyncfile 179
- cluster
  - additional node 53
  - CF states 70
  - CIP traffic 9
  - data file 45
  - interfaces 7
  - name 7
  - node in consistent state 46
  - number of interconnects 8
  - partition 104
- Cluster Admin 66, 67
  - administration 65
  - login window 22
  - main CF table 73
  - routes 72
  - Shutdown Facility 127
  - starting 22, 66
  - starting CF 78
  - stopping CF 78
- Cluster Foundation
  - administration 65
  - configuration 7
  - connection table 29
  - dependency scripts 83
  - device driver 167
  - devices 108
  - Heartbeat monitor 93
  - interface 7
  - loading driver 24
  - log viewer 84
  - main table 70
  - node information 75
  - node name 7, 54
  - quorum set 37
  - remote services 37
  - Response Time monitor 76
  - route tracking 72
  - security 18
  - topology table 29, 76, 107
  - unconfigure 97
- Cluster Integrity Monitor 46
  - adding a node 95
  - CF quorum set 37
  - cfc 37
  - cfsh 37
  - configuration window 37
  - node state 47
  - options 95
  - override 98
  - override confirmation 98
  - quorum state 48
  - rcqconfig 48
- Cluster Interconnect Protocol
  - /etc/cip.cf 54
  - /etc/hosts 10
  - CF Wizard 54
  - cfname 43
  - configuration 9
  - configuration error 62
  - configuration file 43
  - configuration reset 63
  - configuration verification 62
  - defining 9
  - file format 43
  - interfaces 9
  - name 54, 55
  - ping command 54
  - properly configured 54
  - subnetwork 54

- syntax 44
  - CLUSTER\_TIMEOUT 16
  - collecting troubleshooting information 174
  - COMINGUP state 84
  - Command Line Interface
    - configuring SA 147
    - configuring with 145
    - SD 145
  - commands
    - CCBR 177
    - CF 177
    - CIP 178
    - Monitoring Agent 178
    - PAS 178
    - Resource Database 179
  - config.us 180
  - configuration
    - changing 48
    - hardware 60
    - restore 58
    - verify 63
    - See also* configuring 63
  - Configuration Wizard, invoking 127
  - configuring
    - CF 10
    - CF driver 24
    - CIM 47
    - CIP 9, 10, 34, 43
    - CIP with CF Wizard 54
    - IPS 116
    - IPS shutdown agent 132
    - kernel parameters 51
    - resource database 54
    - SNMP Shutdown Agent 133
    - with CLI 145
    - See also* configuration 63
  - connection table 30
  - contents, manual 1
  - corrupt data 101
  - crash dump 176
  - creating
    - cluster, example 20
    - new cluster 26
- D**
- data, corrupt 101
  - debugging 161
  - defining virtual CIP interfaces 9
  - devices
    - displayed 107
    - Ethernet 109
    - unconnected 29
  - diagnostics 163
  - disabling split-brain handling 123
  - disk array units 53
  - display statistics 89
  - displayed devices 107
  - documentation, related 2
  - DOWN routes 72
  - DOWN state 84, 100, 101
- E**
- editing
    - CF node names 28
    - cip.cf file 43
    - cluster.config file 15
    - rcsd.cfg.template 146
  - errors, CIP configuration 62
  - Ethernet 171
    - adapter 171
    - cable 171
    - devices 109
    - driver errors 167
    - example 111
    - Gigabit 8
    - hub 171
    - topology table 107
    - utilities 171
- F**
- fjsnap command 174
    - error message 176
    - executing 175
  - fjsvwwbs 182
  - fjsvwwcnf 182
  - full interconnect 29, 108

### G

Global Disk Services 192

Global File Services 192

Global Link Services 193

GUI

*See* Cluster Admin

### H

Heartbeat monitor 93

hvassert 180

hvcn 180

hvconfig 180

hvdsp 180

hvdump 180

hvenv.local 180

hvlogclean 180

hvsetenv 180

hvshut 180

hvswitch 180

hvutil 180

### I

ICF statistics 90

ifconfig 43

init command 99

Initial Connection Choice window 68

interconnects

CF 8

Ethernet 109

full 29

number of 8

partial 29

topology table 108

interfaces 7

missing 73

network 72

Internet Power Switch 116

configuration 116

configuring SA 132

setup 116

Internet Protocol address

CIP interface 34, 35

INVALID state 84

IP name, CIP interface 35

IP over CF 13

IPMI 118

configure 151

GUI 134

IPS

*See* Internet Power Switch

### J

Java, trusted applets 20

join problems 168

joining a running cluster 56

### K

keyword, search based on 87

### L

Largest Sub-cluster Survival 124

LEFTCLUSTER 194

LEFTCLUSTER state 99, 102, 105,  
172, 191, 194

cluster partition 104

description 100

displaying 99

in kernel debugger too long 104

lost communications 101

node state 197

panic/hung node 102

purpose 101

recovering from 102

shutdown agent 101

libvirt 158, 159, 160

LOADED state 81

loading

CF driver 24

CF driver with CF Wizard 27

CF duration 28

local states 70

login

password 20

window 22

low latency 8

### M

MAC statistics 91

main CF table 70, 73



manual contents 1  
manual pages  
  display 177  
  listing 177  
marking down nodes 84  
micpstat 178  
Monitoring Agent commands  
  cldevparam 178  
  clmmbmonctl 178  
  clmmbsetup 178

**N**  
names  
  CF 75  
  CF cluster 108  
  cname 10, 61  
  cluster 7, 26, 75  
  configuration file 7  
  connections table 30  
  IP 35, 148  
  tuple entries 15  
  user 20  
  Web-Based Admin View 7  
network  
  interfaces 72  
  outages 8  
Node to Node statistics 92  
nodes  
  adding 26  
  adding a new 58  
  details 75  
  joining a running cluster 56  
  marking down 84  
  panicked 99  
  shut down 84

**P**  
panicked nodes 99  
partial interconnects 29, 108  
PAS commands  
  clmtest 178  
  micpstat 178  
passwords 20  
pclsnap command 174  
privileged user ID 21

**Q**  
quorum  
  CF 37  
  CIM override 98  
  reconfiguring 48  
  state 48

**R**  
RAID 202  
RC\_sf 158  
rc2.d directory 158  
rcqconfig 47, 48  
RC-script 158  
rcsd 181  
rcsd log 161  
rcsd.cfg 146, 181  
rdb.tar.Z 60, 61  
rebooting  
  clusterwide 45  
  reboot command 99  
  shut down CF 99  
reconfiguring Resource Database 61  
remote states 70  
Resource Database 54  
  adding new node 58  
  backing up 60  
  clgettree 55  
  csetup 61  
  configure on new node 62  
  initializing 58  
  new node 58  
  reconfiguring 58, 61  
  restoring 63, 64  
  SEMMNS 52  
  SHMMNI 53  
  start up synchronization 56  
  StartingWaitTime 57  
Resource Database commands  
  clautoconfig 179  
  clbackuprdb 179  
  cldeldevice 179  
  clexec 179  
  clgettree 179  
  clinitreset 179  
  clrestorerdb 179

- clsetparam 179
- clsetup 179
- clstartsrc 179
- clstopsrc 179
- clsynccfile 179
- Response Time monitor 76
- restoring
  - cluster configuration 177
  - Resource Database 63, 64
- RFC 1918 9
- RMS commands
  - config.us 180
  - hvassert 180
  - hvcn 180
  - hvconfig 180
  - hvdsp 180
  - hvdump 180
  - hvenv.local 180
  - hvlogclean 180
  - hvsetenv 180
  - hvshut 180
  - hvswitch 180
  - hvutil 180
- RMS Wizard Tools 125
- root 21
- route tracking 72

## S

### SA

*See* Shutdown Agents

- SA specific log files 161
- SA\_blade.cfg 181
- SA\_icmp 115, 120, 153, 181
- SA\_ipmi.cfg 181
- SA\_libvirtgp.cfg 182
- SA\_libvirtgr.cfg 182
- SA\_vmchkhosst.cfg 182
- SA\_vmSPgp.cfg 181
- SA\_vmSPgr.cfg 182
- SA\_wtinps.cfg 181
- saving
  - cfbackup command 177
  - RMS configuration 180
  - SF configuration 143

### SD

*See* Shutdown Daemon

- sdtool 181
- sdtool command 113, 158
- sdtool -d onÉRÉ}ÉiÉh 158
- search
  - keyword 87
  - severity levels 88
  - time filter 86

### security

CF 18

selecting devices 109

SEMMNS 52

### SF

*See* Shutdown Facility

### SF commands

rcsd 181

sdtool 181

sfcipher 181

### SF Wizard 7

opening 127

starting 40

sfcipher 181

SHMMAX 51

SHMMNI 53

### Shutdown Agents 113

configuring with CLI 147

configuring with GUI 127

with LEFTCLUSTER 101

shutdown command 99

### Shutdown Daemon 113

configuration file 146

configuration file format 146

configuring with CLI 145

rcsd.cfd 146

### Shutdown Facility 7, 113

configuring with GUI 127

internal algorithm 123

node weight 125

RMS Wizard Tools 125

saving 143

split-brain handling 121

starting and stopping 157

starting automatically 157

- starting manually 157
  - stopping automatically 157
  - stopping manually 157
  - weight assignment 123
  - ShutdownPriority attribute 122
  - signed applets 20
  - simple virtual disks 201
  - single user mode 60
  - SMAWsf directory 146
  - special priority interfaces 9
  - Specific Application Survival 124
  - Specific Hardware Survival 124
  - split-brain 121
    - handling 123
    - LSS 124
    - SAS 124
    - SHS 124
  - sshconf 19
  - start up synchronization 56
    - new node 58
    - StartingWaitTime 63
  - starting
    - CF 78, 79
    - CF Wizard 24
    - Cluster Admin 10
    - GUI 22
    - SF Wizard 40, 127
    - Web-Based Admin View 20
  - StartingWaitTime 56, 59
    - default value 57
    - value 57
  - start-up window 66
  - states
    - COMINGUP 70, 84
    - DOWN 71, 100, 101
    - INVALID 70, 84
    - LEFTCLUSTER 71, 99, 102, 105
    - LOADED 70, 81
    - table of 100
    - UNCONFIGURED 71
    - UNKNOWN 70, 71
    - UNLOADED 70
    - UP 70, 71, 100
  - statistics, display CF 89
  - stopping
    - CF 78, 79
    - CF, third-party products 84
    - SD 115
    - SF automatically 158
    - SF manually 158
    - valid CF states 84
  - subnet mask, CIP interface 35
  - synchronization phase 56
  - synchronization, start up 56
  - sysctl 52
  - system dump 174
- T**
- table of states 100
  - third-party product, shut down 84
  - time filter, search 86
  - timeout, tune 17
  - timestamp 57
  - top window 67
  - topology table 107
    - basic layout 108
    - CF 29, 76
    - CF cluster name 108
    - displayed devices 107
    - displaying 76
    - examples 110
    - flexibility 30
    - interconnects 108
    - Response Time monitor 76
    - selecting devices 109
  - troubleshooting 163
    - beginning 163
    - collecting information 174
    - diagnostics 163
    - join related problems 167
    - symptoms and solutions 167
  - tunable parameters 15
  - tune timeout 17
  - tuple entries
    - name 15
    - value 15
- U**
- unconfigure CF 97

## Index

---

unconnected devices 29, 108  
UNKNOWN state 70  
UNLOADED state 81  
UP state 100  
updating CFReg 49  
user ID 21  
user name 21

## V

virtual disks  
    simple 201

## W

Web-Based Admin View  
    known nodes 41  
    node list 23  
    starting 20  
Web-Based Admin View commands  
    fjsvwwbs 182  
    fjsvwwcnf 182  
    wvCntl 182  
    wvGetparam 182  
    wvSetparam 182  
    wvstat 182  
wvCntl 182  
wvGetparam 182  
wvroot 21  
wvSetparam 182  
wvstat 182