

PRIMECLUSTER™

Scalable Internet Services (SIS) (Solaris® , Linux®) Configuration and Administration Guide

Fujitsu Siemens Computers GmbH, Paderborn
33094 Paderborn

E-Mail: email_mangels@fujitsu-siemens.com

Tel.: (0440) 606-00010

Fax: (+49) 700 7 072 0001

U42126-J42100-3-76

Sprachen: En

Comments... Suggestions... Corrections...

The User Documentation Department would like to know your opinion of this manual. Your feedback helps us optimize our documentation to suit your individual needs.

Fax forms for sending us your comments are included in the back of the manual.

There you will also find the addresses of the relevant User Documentation Department.

Certified documentation according DIN EN ISO 9001:2000

To ensure a consistently high quality standard and user-friendliness, this documentation was created to meet the regulations of a quality management system which complies with the requirements of the standard DIN EN ISO 9001:2000.

cognitas. Gesellschaft für Technik-Dokumentation mbH
www.cognitas.de

Copyright and Trademarks

Copyright © 2002, 2003, 2003 Fujitsu Siemens Computers Inc. and Fujitsu LIMITED.

All rights reserved.

Delivery subject to availability; right of technical modifications reserved.

Solaris and Java are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and other countries.

Linux is a registered trademark of Linus Torvalds.

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

This manual is printed on paper treated with chlorine-free bleach.

Preface

Introduction

Configuration

Satellite nodes

NIC failover

Administration

Syntax rules

Debugging and troubleshooting

Manual pages

Index

Contents

1	Preface	1
1.1	Contents of this manual	1
1.2	Related documentation	2
1.2.1	Suggested documentation	3
1.3	Conventions	4
1.3.1	Notation	4
1.3.1.1	Prompts	4
1.3.1.2	Manual page section numbers	4
1.3.1.3	The keyboard	4
1.3.1.4	Typefaces	5
1.3.1.5	Example 1	5
1.3.1.6	Example 2	5
1.3.2	Command syntax	6
1.4	Important	6
2	Introduction	7
2.1	SIS overview	7
2.1.1	Service nodes	8
2.1.2	Gateway nodes	8
2.1.3	Primary database node	9
2.1.4	Backup database node	9
2.2	Satellite nodes	9
2.3	Benefits	10
2.4	SIS architecture	12
2.5	VIP	13
2.6	PROXY	14
2.7	PRIVATE	14
2.8	Service node failover	14
2.9	Cluster Admin	15
3	Configuration	17
3.1	Concepts	17
3.2	Configuration file	18
3.2.1	Variables	19
3.2.2	NODES declaration	20
3.2.3	GATEWAYS declaration	20
3.2.4	Interface definitions	21
3.2.4.1	VIP	21
3.2.4.2	PROXY addresses	26
3.2.4.3	PRIVATE addresses	27
3.3	Configuring with Cluster Admin	28

Contents

3.3.1	Starting SIS	28
3.3.2	Logging in to Cluster Admin	28
3.3.3	Displaying the SIS GUI main window	33
3.4	Creating a new configuration file	34
3.4.1	Adding nodes	37
3.4.2	Configuring satellite nodes	38
3.4.3	Defining VIP, PROXY, and PRIVATE addresses	42
3.4.3.1	VIPs	43
3.4.4	Completing the configuration	61
3.5	Starting with an existing configuration file	68
3.6	Examples and configuration files	70
4	Satellite nodes	73
4.1	Overview	73
4.2	Software	74
4.3	Hardware	74
4.4	Setting up satellite nodes	75
4.4.1	Specifying cluster name on Windows systems	76
4.4.2	Specifying cluster name on Linux systems	82
4.4.3	Specifying cluster name on Solaris OE systems	82
5	NIC failover	83
5.1	Introduction	83
5.2	SIS NIC failover module	84
5.2.1	Monitoring	84
5.2.2	Failover mode	84
5.2.3	Restore actions	85
5.2.4	Starting and restarting	85
5.2.5	Trusted host configuration	85
6	Administration	87
6.1	Administering with Cluster Admin	87
6.1.1	Using the GUI	87
6.1.1.1	SIS configuration tree	88
6.1.2	Using the menu bar	101
6.1.2.1	File	101
6.1.2.2	Tools	102
6.1.2.3	Help	115
6.2	Administering with the CLI	118
6.3	Displaying the status of SIS	121
6.3.1	Status by node	122
6.3.2	Status by service	122
6.3.3	Status of SIS connections	124
6.3.4	Showing the gateway node	126

6.4	SIS daemon	126
6.5	Debug messages	127
7	Syntax rules	129
8	Debugging and troubleshooting	133
8.1	dtcpdbg	133
8.2	Troubleshooting	135
9	Manual pages	137
Index		139

Contents

1 Preface

This guide provides instructions on how to configure and administer the Fujitsu Siemens Computers Inc., SIS (Scalable Internet Services®) product.

The primary audience for this guide is the system administrator.

1.1 Contents of this manual

This manual is organized as follows:

- The Chapter “Introduction” provides a brief overview of SIS (Scalable Internet Services), including terms, concepts, functions, and components.
- The Chapter “Configuration” describes SIS configuration files and how to configure them. In addition, some example configurations are supplied.
- The Chapter “Satellite nodes” details the requirements for setting up satellite node configurations.
- The Chapter “NIC failover” describes SIS Network Interface Card (NIC) failover, including how SIS recognizes a failure, how it responds, and what you need to do to configure the SIS NIC failover module.
- The Chapter “Administration” describes how to administer SIS with Cluster Admin and details the SIS utilities.
- The Chapter “Syntax rules” lists the syntax rules for a SIS configuration file.
- The Chapter “Debugging and troubleshooting” details the `dtcpdbg` command and answers some common configuration and administration questions.
- The Chapter “Manual pages” lists the manual pages for SIS.

1.2 Related documentation

The documentation listed in this section contains information relevant to PRIMECLUSTER and can be ordered through your sales representative.

In addition to this manual, the following manuals are also available for PRIMECLUSTER:

- Release notices for all products—These documentation files are included as HTML files on the PRIMECLUSTER Framework CD. Release notices provide late-breaking information about installation, configuration, and operations for PRIMECLUSTER. Read this information first.
- *Concepts Guide (Solaris, Linux)*—Provides conceptual details on the PRIMECLUSTER family of products.
- *Installation Guide (Solaris)*—Provides instructions for installing and upgrading PRIMECLUSTER products.
- *Installation Guide (Linux)*—Provides instructions for installing and upgrading PRIMECLUSTER products.
- *Reliant Monitor Services (RMS) with Wizard Tools (Solaris, Linux) Configuration and Administration Guide*—Provides instructions for configuring and administering RMS using PRIMECLUSTER Wizard Tools.
- *Reliant Monitor Services (RMS) with PCS (Solaris, Linux) Configuration and Administration Guide*—Provides instructions for configuring and administering RMS using PRIMECLUSTER Configuration Services (PCS).
- *Reliant Monitor Services (RMS) (Solaris, Linux) Troubleshooting Guide*—Describes diagnostic procedures to solve RMS configuration problems, including how to view and interpret RMS log files. Provides a list of all RMS error messages with a probable cause and suggested action for each condition.
- *Cluster Foundation (CF) (Solaris) Configuration and Administration Guide*—Provides instructions for configuring and administering the PRIMECLUSTER Cluster Foundation.
- *Cluster Foundation (CF) Configuration and Administration Guide (Linux)*—Provides instructions for configuring and administering the PRIMECLUSTER Cluster Foundation.
- *Global Disk Services (Solaris) Configuration and Administration Guide*—Provides information on configuring and administering Global Disk Services (GDS).

- *Global Disk Services (Linux) Configuration and Administration Guide*—Provides information on configuring and administering Global Disk Services (GDS).
- *Global File Services (Solaris) Configuration and Administration Guide*—Provides information on configuring and administering Global File Services (GFS).
- *Global File Services (Linux) Configuration and Administration Guide*—Provides information on configuring and administering Global File Services (GFS).
- *Global Link Services (Solaris) Configuration and Administration Guide: Redundant Line Control Function*—Provides information on configuring and administering the redundant line control function for Global Link Services (GLS).
- *Global Link Services (Solaris) Configuration and Administration Guide: Redundant Line Control Function*—Provides information on configuring and administering the redundant line control function for Global Link Services (GLS).
- *Global Link Services (Linux) Configuration and Administration Guide: Multipath Function*—Provides information on configuring and administering the multipath function for Global Link Services (GLS).
- *Web-Based Admin View (Solaris/Linux) Operation Guide*—Provides information on using the Web-Based Admin View management GUI.
- *SNMP Reference Manual (Solaris, Linux)*—Provides reference information on the Simple Network Management Protocol (SNMP) product.
- *Data Management Tools (Solaris) Configuration and Administration Guide*—Provides reference information on the Volume Manager (RCVM) and File Share (RCFS) products.
- *RMS Wizards documentation package*—Available on the PRIMECLUSTER CD. These documents deal with topics such as the configuration of file systems and IP addresses. They also describe the different kinds of wizards.

1.2.1 Suggested documentation

The following manuals contain information relevant to PRIMECLUSTER administration and can be ordered through your sales representative (not available in all areas):

- *ANSI C Programmer's Guide*
- *LAN Console Installation, Operation and Maintenance*
- *Terminal TM100/TM10 Operating Manual*

- *PRIMEPOWER User's Manual* (operating manual)



Your sales representative will need your operating system release and product version to place your order.

1.3 Conventions

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) rights 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 *fujii2#* would mean that the command was run as user `root` on the node named `fujii2`.

1.3.1.2 Manual page section numbers

References to the operating system commands are followed by their manual page section numbers in parentheses — for example, `cp(1)`.

1.3.1.3 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.

1.3.1.4 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>	Variables that you must replace with an actual value.
Bo1d	Items in a command line that you must type exactly as shown.

Typeface conventions are shown in the following examples.

1.3.1.5 Example 1

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

```
root:x:0:1:0000-Admin(0000):/:/sbin/ksh
sysadm:x:0:0:System Admin.:/usr/admin:/usr/sbin/sysadm
setup:x:0:0:System Setup:/usr/admin:/usr/sbin/setup
daemon:x:1:1:0000-Admin(0000):/:
```

1.3.1.6 Example 2

To use the `cat(1)` 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

Material of particular interest is preceded by the following symbols in this manual:



Contains important information about the subject at hand.



Caution

Indicates a situation that can cause harm to data.

2 Introduction

This section discusses the following:

- The Section “SIS overview” introduces the concepts of SIS and the various SIS components.
- The Section “Satellite nodes” describes the major differences between satellite nodes and regular SIS nodes.
- The Section “Benefits” lists the SIS features.
- The Section “SIS architecture” explains the functions of SIS in detail.
- The Section “VIP” introduces Virtual Interface Providers (VIPs), and it discusses the predefined scheduling algorithms used by SIS to schedule client requests.
- The Section “PROXY” discusses the concepts and uses of PROXY addresses.
- The Section “PRIVATE” details the uses of PRIVATE addresses.
- The Section “Service node failover” describes the failover mechanisms available in SIS.

2.1 SIS overview

SIS provides scalable and fault tolerant network services based on the underlying PRIMECLUSTER technology. SIS enables PRIMECLUSTER to act as a scalable, reliable, and easily managed network server system. Some or all of the nodes in PRIMECLUSTER can be configured as the SIS Cluster.

SIS provides the following three kinds of access to network services that are not usually available on standard servers:

- VIP—A single, virtual address that provides transparent public access to network services running on a list of nodes. The nodes can be configured to distribute the load per service, and you can fine-tune unique application and site needs in a variety of ways.

- PROXY—A virtual address that provides public access to all network services on a single node without accessing the node directly
- PRIVATE—A virtual address that provides protected access to all network services running on a single node from within the cluster

SIS eliminates single points of failure and ensures availability as follows:

- If any of the SIS nodes or services fail, SIS schedules requests around the failed nodes (VIP).
- VIP and PROXY allow the definition of failover nodes that will provide the services in case the primary service node or nodes fail.
- After a failed node or service is restarted, it will seamlessly become available again to restore maximum performance of the SIS cluster.
- Important internal functions of SIS, namely gateway nodes and database nodes, recover transparently in case of failures. This includes failure of a public network interface card (NIC).

The sections that follow discuss the various types of SIS nodes.

2.1.1 Service nodes

Service nodes offer network services such as web services and directory services. If a service node fails, services are scheduled around it. When a failed node comes back up, it joins the SIS cluster.

2.1.2 Gateway nodes

There is one gateway node per VIP address. All incoming packets are received by this node and forwarded to the service node, depending on the scheduling algorithm for the service. If a gateway node fails, another node assumes the role of the failed gateway node without any interruption.

2.1.3 Primary database node

The primary database node keeps the static and dynamic data of the SIS cluster. The static information may include the list of nodes in the SIS cluster, the VIP address and services offered, and the scheduling algorithms. The dynamic information includes the current list of connections and the current status of the SIS cluster.

2.1.4 Backup database node

The backup database node assumes the role of the primary database node when the primary database node fails for any reason. There can be more than one backup database node. One of the backup database nodes will become the primary database node if the primary database node fails. Each backup database node contains the static configuration details, and SIS collects the dynamic configuration items from all the available nodes; therefore, network disruption is kept to a minimum.

2.2 Satellite nodes

SIS cluster nodes can connect with other cluster nodes by means of CF or they can join the SIS cluster as cluster members without CF. A core node connects to a SIS cluster using CF, while satellite nodes do not use CF. The following properties differentiate the two types of nodes:

- Core nodes are aware of other nodes in the cluster and can be any of the various types of SIS nodes.
- Satellite nodes can only be service nodes.
- SMAWcf must not be installed on a satellite node.
- Windows 2000 nodes can only be used as satellite nodes.

Figure 1 illustrates a five-node SIS configuration that has three core nodes, one Linux satellite node, and one Windows 2000 satellite node.

Refer to the Chapter “Satellite nodes” for more information.

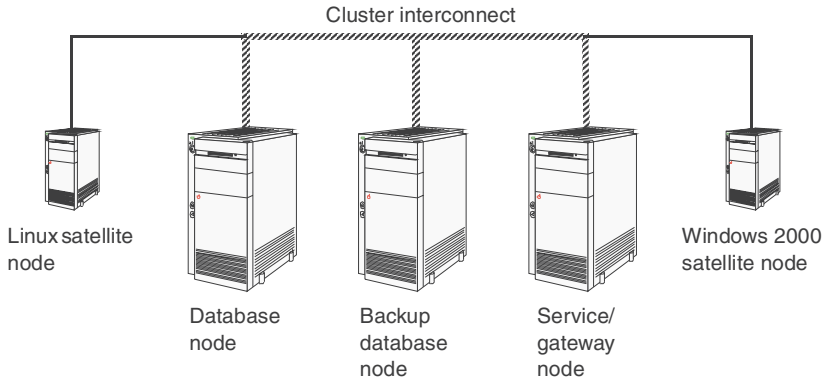


Figure 1: SIS cluster with satellite nodes

2.3 Benefits

The benefits of SIS are as follows:

- Provides scaling and load balancing for network services
- Allows network access to a group of servers through one address
- Supports multiple scheduling options for fine-tuned, granular load balancing
 - No fixed scaling limits
 - No hardware under-utilization
- Supports all TCP/IP protocols—http, ftp, proxy, SSL, POP3, and SMTP
- Supports the UDP protocol
- Offers highly available, seamless access to running service systems and applications
- Integrates into a full-featured cluster with high availability and scaling for applications
- Provides a dynamic and easy way to configure interface

- User friendly GUI
- Runs on the same server as the application
- Does not require special hardware
- Does not need network changes
- Applications do not need to be changed
- Supports satellite nodes
- Support for Windows 2000 service nodes as satellite nodes

2.4 SIS architecture

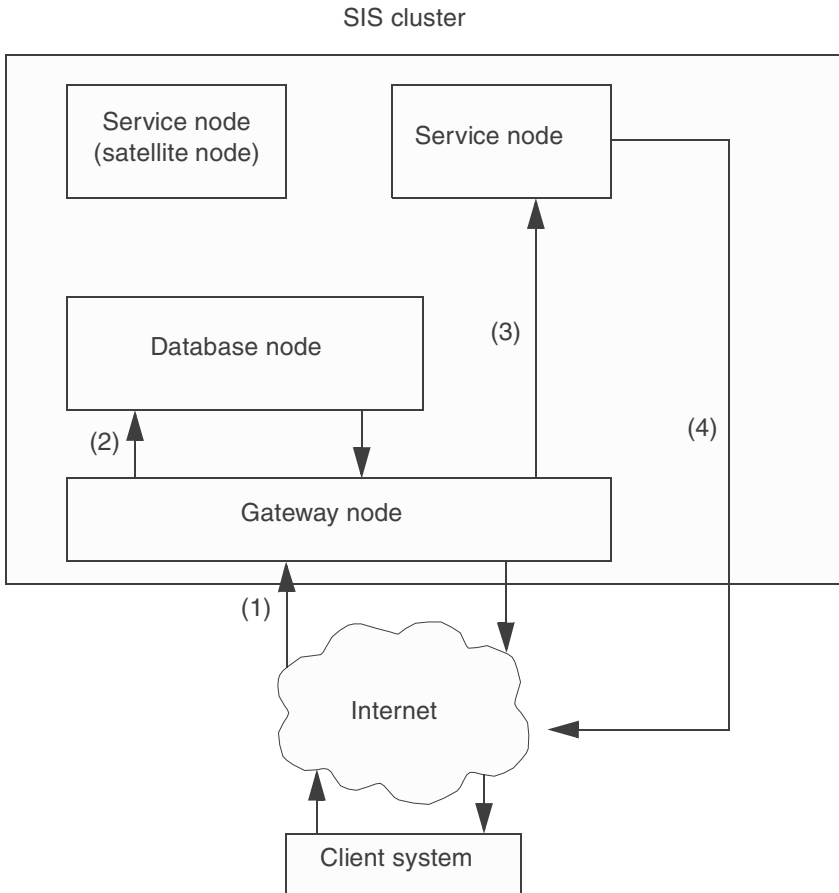


Figure 2: Example of how a SIS cluster processes requests

In Figure 2, the SIS cluster consists of one gateway node, two service nodes, and one database node (the service node, the database node, and the gateway node can all reside in one physical node). When the client sends a network request, the SIS cluster responds as follows:

1. The request reaches the gateway node.

2. The gateway node queries the database node to determine which service node will reply to this network request. The database node determines this according to the current configuration and the availability of nodes and services.
3. The gateway node keeps the result cached for future packets of the same connection.
4. The IP packet is then sent to the service node where it is forwarded on to the application.
5. The application on the same service node replies directly to the client (without going through the gateway node).

2.5 VIP

A Virtual Interface Provider (VIP) is a single virtual address, channelling access to network services provided by one or more nodes in the cluster. A VIP requires definition of port (service), port protocol, scheduling algorithm, and a list of one or more service nodes that are providing this service.

The following scheduling algorithms govern the distribution of connection requests according to various load requirements, or they associate a client with a special service node:

- Keep local—The gateway node answers the request without any processing overhead.
- Client based—The service node is chosen based on the client's IP address.
- System load—The system with the minimum system load is chosen. The system load is calculated by SIS using an internal algorithm.
- Round robin—All available nodes are chosen in a circular way.
- Spill over—If the system load on all primary nodes is equal or greater than the configured threshold, a backup node is chosen to lessen the load.
- Weighted connection count—SIS chooses the node with the least number of connections. By assigning a weight to parts of the node list, this number may be recalculated to reflect certain preferences.

2.6 PROXY

PROXY addresses are public virtual addresses to a single node. Since PROXY addresses have failover capabilities, they also provide high availability. The possible uses are as follows:

- Co-hosting multiple addresses to one node
- Assigning external connectivity to nodes that do not have connections to the Internet
- Allocating backup nodes to a node

2.7 PRIVATE

PRIVATE addresses provide virtual IP addresses to communicate by means of the CF cluster interconnect. By using the scaling and failover capabilities of the cluster interconnect, PRIVATE provides High Availability to internode communication. These addresses cannot be routed to or from external networks.

2.8 Service node failover

A VIP service defines a list of primary service nodes, and a PROXY is defined to connect to a single primary service node, either of which will receive the client requests.

The FAILOVER construct provides a method to associate a list of secondary service nodes to one or more primary service nodes.

The first available secondary service node will receive the client request if the scheduled primary node cannot provide the service for one of the following reasons:

- The node has shutdown
- The node has ceased to be a cluster member (CF status not UP—core nodes only)
- SIS has been deactivated on the node (core nodes only)
- The node has been expelled from the cluster (satellite nodes only)
- The configured service has been inactivated on the node (VIP only)

2.9 Cluster Admin

Cluster Admin is an administrative graphical user interface (GUI) that is reached through Web-Based Admin View. For creating and editing configuration files, Cluster Admin contains a SIS configuration wizard. Cluster Admin can manage the following SIS procedures:

- Configuration wizard
- Administration
- Operations and diagnostics services



A Java-enabled Web browser serves as the administrative interface; a conventional command-line interface (CLI) is also available. To provide a consistent configuration, we recommend using Cluster Admin as opposed to the CLI.

3 Configuration

This chapter describes the configuration syntax for SIS, and it provides examples of configuration files to assist users with the configuration process. In addition, it describes how to configure SIS using the Cluster Admin GUI.

This chapter discusses the following:

- The Section “Concepts” introduces some basic concepts for SIS.
- The Section “Configuration file” provides a description of the configuration file, including variable assignments, node and gateway declarations, and interface definitions.
- The Section “Configuring with Cluster Admin” describes how to configure SIS using the Cluster Admin GUI.
- The Section “Examples and configuration files” provides examples of some configuration files.

3.1 Concepts

The following concepts and definitions are important in the configuration of SIS:

- **SIS cluster**—A subset of the CF cluster that provides the scalable and highly available network services of the PRIMECLUSTER suite.
- **Database node**—Acts as a repository of the current state of the system. It keeps the current configuration, keeps track of the current list of available nodes and services, knows about the current connections and their state, and other housekeeping duties, which ensure the smooth running of SIS.
- **Backup database node**—A node in the current cluster that takes up the role of the primary database node if the primary database node goes down for any reason. There can be more than one backup database node in the SIS cluster.
- **Gateway node**—Any node that can communicate to the external world using the IP protocol.
- **Services**—Network services such as `http` or `ldap`.
- **Service node**—Provides a network service.

- Core node—Any regular SIS node based on CF. Core nodes can assume the role of any of the various SIS node types.
- Satellite node—An irregular SIS node that does not use CF and can only function as a service node.
- Failover node—Any node which can take over the role of a service node if the service node goes down for any reason.
- Virtual Interface Provider (VIP)—IP address or name provided by SIS to define a single network address for a SIS cluster.
- PROXY—IP address or name assigned to a SIS node on the public network, which make all services of that node highly available.
- PRIVATE—IP address or name assigned to a SIS node to communicate privately and securely using the CF interconnect.

3.2 Configuration file

The SIS configuration file is used to configure the SIS cluster to provide the scalable and highly available network services of the PRIMECLUSTER suite. The SIS configuration file should be created using the Cluster Admin GUI. (Since it is a text file, it can also be created or edited by using a text editor; however, this is recommended only for experienced users.)

The SIS configuration file is read once from the top down and requires a minimum of one interface definition. The items contained within the configuration file must be maintained in the following order:

- Variable assignments—This is an optional section where variables can be assigned for later use.
- Nodes declaration—This section lists all the nodes in the SIS cluster.
- Gateways declaration—This section lists all the nodes in the SIS cluster that have an interface on the public network.
- Interface definitions—This list defines VIP addresses, PROXY addresses, and PRIVATE addresses used by the SIS cluster.



The hash mark (#) indicates a comment. Everything following the hash mark (#) is ignored until End of Line (EOL).

3.2.1 Variables



The graphical user interface, Cluster Admin, does not support assignment or management of variables. This section is only applicable if configuration files are edited manually.

This section, if present, must be the first section in the configuration file.

A variable declaration has the following syntax:

```
variable_name=string_0...string_n;
```

A variable declaration provides a string substitution mechanism, which makes it more convenient to confine changes in the configuration to a single location. For example, you can define a group of nodes in the variable section of the configuration file without having to change service definitions in the body of the VIP definition.

Rules and limitations

The following rules and limitations apply to variable definitions:

- Variable definitions are evaluated only once, so their sequence is important. If, as in the following example,

```
A = X;  
X = fuji2;  
B = X;
```

then, in the sections of the configuration file that follow, A will resolve to X, but B will resolve to fuji2.

- The following characters can be used by a variable definition:
 - All letters and numbers, including underscores (`_`), dashes (`-`), and spaces
 - Parentheses may be used, but only if they contain a `FAILOVER` statement
- Character types other than those above cannot be used. Some common examples of this are slashes (`/`), and colons (`:`).
- The following keywords cannot be part of a variable definition and will produce syntax errors:

```
VIP, GATEWAYS, SERVICE, PROXY, PRIVATE, NODES, KEEPLOCAL, CLBASED,  
SYSLOAD, ROUNDROBIN, FAILOVER, CONCOUNT, SPILLOVER, AT, TO, udp,  
tcp, NONE, DCL, and DPO.
```

However, `FAILOVER` may be used in a variable definition if it is contained within parentheses and part of a `FAILOVER` list.

The following are examples of valid variable assignments:

```
HTTP_NODES=fuji2 fuji3;
```

```
FUJI3 = fuji3;
```

3.2.2 NODES declaration

`NODES` defines the membership of the SIS configuration (includes satellite nodes). It has the following syntax:

```
NODES nodedef_1.....nodedef_n
```

nodedef can be a node name or a variable (the variable resolves to one or more node names).

The following are examples of valid node definitions:

```
NODES HTTP_NODES
```

```
NODES fuji2 FUJI3
```

3.2.3 GATEWAYS declaration

`GATEWAYS` defines the SIS nodes that have an external interface on the public network. It has the following syntax:

```
GATEWAYS nodedef_1 . . . nodedef_n
```

nodedef can either be a node name or a variable.

The following are examples of valid gateway definitions:

```
GATEWAYS HTTP_NODES
```

```
GATEWAYS fuji2 FUJI3
```



If a node has an external interface on the same network as any of the public SIS services (that is, `VIP` or `PROXY`), then the node name must be in the `GATEWAYS` list.

3.2.4 Interface definitions

Interface definitions can be of the following types:

- VIP
- PROXY
- PRIVATE

Interface definitions can appear in any order, but they may not be contained in one another.

3.2.4.1 VIP

A VIP provides a public address for selected services that are provided by the service nodes of a SIS cluster. The client requests are transparently assigned to a service node, according to a predefined scheduling algorithm. The syntax is as follows:

```
VIP (interface_1)...(interface_n) { vip_body }
```

The following terms and conditions apply:

- *interface* has the form *IPAddress netmask [preferred_gateway]*.
 - *IPAddress* is a resolvable name or IP address.
 - *netmask* is the netmask associated with *IPAddress*.
 - *preferred_gateway* is an optional node name that indicates which node should receive incoming data for the VIP address.
- *IPAddress* can be a name, dot notation, or hexadecimal notation.

The following are examples of valid *interface* definitions:

```
www.siscluster.com 255.255.255.0 fuji2  
www.clustersis.com 255.255.255.0
```

VIP body

The body of the VIP defines the network services available on this VIP and how client requests for these services are to be scheduled. Each service is defined on a separate line. A service definition has one of the following formats:

- SERVICE *portdef scheduling*
- SERVICE *portdef* DCL *scheduling*

portdef defines a port or range of ports (network service) and a protocol in the following format: *port/protocol*. The *protocol* can be either `tcp` or `udp`. If you do not specify a protocol, the default is `tcp`. *port* defines the network service and can be any of the following:

- Number (such as 80)
- Range of numbers (such as 1:17)
- Symbolic name (such as `http`) from `/etc/services` or other resolution schemes
- Range of symbolic names from `/etc/services` or other resolution schemes

The following are examples of *portdef*:

- `80/tcp`
- `http`
- `8080:8090/udp`
- `sunrpc/udp`

DCL (depends on client) directs SIS to remember a client such that all connections of the client for that service will go to the same node. The first connection is scheduled based on the scheduling algorithm (for example, `ROUNDROBIN` or `SYSLOAD`).

Scheduling for services

scheduling defines how incoming requests from clients get assigned to the list of nodes that provide the service. The following algorithms are available:

- Keep local—A connection request to this service will be established on the gateway node itself with little overhead. The syntax is as follows:

```
KEEPLOCAL
```

- Client based—The service node is calculated based on the client's IP address. The syntax is as follows:

```
CLBASED ServiceNodeList
```

- System load—The node with the lowest system load is chosen for placing a connection. The syntax is as follows:

```
SYSLOAD ServiceNodeList
```

- Round robin—All nodes are used in a forced sequence. The syntax is as follows:

`ROUNDRROBIN ServiceNodeList`

- Spill over—the service node is chosen from one of two node lists, depending on a threshold value `load`. The node chosen is the one from `ServiceNodeList_1` with the lowest load. If this load is higher than the defined threshold value, the node with the lowest load from the second group is chosen (`ServiceNodeList_z`). The syntax is as follows:

`SPIILLOVER ServiceNodeList_1 AT load TO ServiceNodeList_z`

SIS uses a sophisticated algorithm for calculating system load, which returns a numeric value between 0 and 1. The value is calculated from the available hardware and various load parameters and depends on the current configuration and system load.

Approximate values derived from internal tests and calculations are as follows:

- 0.0 to 0.5 is a system that has a low load (almost idle)
- 0.5 to 0.7 is a system that has a moderate load (busy)
- 0.7 and above is a system that has a high load (saturated)

Choose a number for `SPIILLOVER` that will help to avoid saturation.

- Weighted connection count—The node with the lowest number of open connections is used. The syntax is as follows:

`CONCOUNT ServiceNodeListElement...ServiceNodeListElement`

`ServiceNodeListElement` is defined as one of the following:

- `node:weight`
- `(ServiceNodeList FAILOVER ServiceNodeList):weight`

The addition of a weight to a list member will recalculate the number of open connections of this node before it is compared. This allows to give a preference to certain nodes. A node with a higher weight will appear to have fewer open connections and may be selected more often than a comparatively less busy node with a lower weight. A default value of 1 is assumed if no weight is given.

The following example shows a weighted connection count `SERVICE` definition that schedules `telnet` among different nodes:

```
SERVICE telnet CONCOUNT fuji1:2 fuji2 fuji3:3 (fuji4 FAILOVER fuji5):4
```

In this example, `fuji4` will be selected 50% of the time if all of the nodes are of similar power and have similar connection loads because `fuji4` is now the node with a weight of 4.



If the definition of a weighted connection count contains variables, the weight, including the colon (:), cannot be part of the variable. The variable facility provides simple string substitution. You can substitute variables as follows:

```
A = fuji2 fuji3;
B = (fuji4 FAILOVER fuji5);
SERVICE telnet CONCOUNT fuji1:2 A:3 B:4
```

This example will therefore resolve to the following:

```
SERVICE telnet CONCOUNT fuji1:2 fuji2 fuji3:3 (fuji4 FAILOVER fuji5):4
```

ServiceNodeList as used in most of the preceding scheduling algorithms is defined as one of the following:

- *node_l...node_n*
- (*node_h...node_r* FAILOVER *node_s...node_z*)

Scheduling for UDP-based services

To load-balance UDP services, SIS supports the same scheduling algorithms as for TCP.

Unlike TCP, UDP does not have connection semantics built into the protocol. Instead, the SIS UDP algorithms provide a configurable timeout, within which incoming requests from the same client and the same client port number will be assigned to the same server, establishing the concept of a pseudo-connection.

If no further request is received within the timeout, the pseudo-connection is closed. Any requests for the same service that arrive after this interval are treated as a new pseudo-connection and are scheduled accordingly.

Refer to the Chapter “Manual pages” and to the Section “Administering with the CLI” for how to change the timeout value using the `dtcpadmin(1M)` command. The default value is 5 seconds.

The following are examples of UDP-based scheduling:

```
SERVICE 2049/udp SYSLOAD fuji1 fuji4
SERVICE 118/udp SYSLOAD (fuji1 fuji4 FAILOVER fuji2 fuji3)
```

Failover

The `FAILOVER` construct is used in place of a simple list of nodes when additional fault tolerance is required in `SERVICE` or `PROXY` definitions.

The syntax of `FAILOVER` for VIP is as follows:

```
( node_a...node_m FAILOVER node_n...node_z )
```

If the *service*, `SIS`, or the node itself fail on one or more nodes from *node_a ...node_m*, then a functional node from *node_n ...node_z* will replace the failed nodes. When the failed node or service becomes available, the scheduling reverts back to the original node.

Examples of `FAILOVER` for a VIP can be found in the examples for various scheduling algorithms.



Only one node will be used to replace all failed nodes within the same `FAILOVER` declaration even if more than one replacement node is defined as in the above syntax.

DCL

DCL is an optional qualifier which can be used after the portdef of a `SERVICE` definition. Its function is to bind all further connections for a client to the same server, once the very first connection has been selected according to the scheduling algorithm for this `SERVICE`.

The following are examples for the use of DCL:

```
SERVICE ftp DCL ROUNDROBIN fuji2 fuji3
SERVICE 8080:8090/udp DCL ROUNDROBIN fuji2 fuji3
```

FTP notes

SIS supports the FTP protocol for VIP with the following special limitations:

- You can configure port 21 (`ftp`) but not port 20 (`ftp-data`). Service declarations that include port 20 will be rejected.
- You can configure `KEEPLOCAL` and `CLBASED` service for port 21 without restrictions.
- The use of the DCL qualifier is mandatory for port 21 `SERVICE` declarations with scheduling algorithms other than `KEEPLOCAL` and `CLBASED`.

The following are valid configurations:

```
SERVICE ftp DCL ROUNDROBIN fuji2 fuji3
SERVICE 21 KEEPLOCAL
SERVICE ftp CLBASED fuji2 fuji3
```

The following are invalid configurations:

```
SERVICE ftp-data CONCOUNT fuji1 fuji2 #illegal use of port 20
SERVICE 20 CONCOUNT fuji2 fuji3
SERVICE 19:25 CONCOUNT fuji2 fuji3
SERVICE ftp ROUNDROBIN fuji2 fuji3 #DCL required
```

3.2.4.2 PROXY addresses

`PROXY` addresses are provided by SIS to associate a virtual IP address to a single node. All client requests to a `PROXY` address are forwarded to the `PROXYNode`. One or more failover nodes can be associated with this node. The first available node on this list will take over in case the primary `PROXY` node fails.

`PROXY` has the following syntax:

```
PROXY IPAddress netmask PROXYNode [preferred_gateway]
```

- *IPAddress* is a resolvable name or IP address.
- *netmask* is the netmask associated with the above IP address.
- *PROXYNode* is a node or failover list.

The following are examples of `PROXYNode`:

```
fuji2
(fuji1 FAILOVER fuji2 fuji3)
```

The syntax for PROXY FAILOVER is as follows:

```
( node_a FAILOVER node_n...node_z )
```

If SIS on *node_a* or the node itself fails, then a node from *node_n...node_z* will replace the failed node. When the failed node becomes available, the connections stay on the replacement node.



Since PROXY does not allow limiting the services that can be requested on the node, it provides failover only for a failure of the node itself, not any failed service.

- *preferred_gateway* is an optional node name that indicates which node should receive incoming data for the IP address.
- *IPAddress* can be a name, dot notation, or hexadecimal notation.

The following are valid examples of PROXY configurations:

```
PROXY ftp.mycompany.com 255.255.255.0 fuji1
PROXY ftp.mycompany.com 255.255.255.0 (fuji1 FAILOVER fuji2)
PROXY ftp.mycompany.com 255.255.255.0 fuji1 fuji3
PROXY ftp.mycompany.com 255.255.255.0 (fuji1 FAILOVER fuji2) fuji3
```

In the first example, packets destined for `ftp.mycompany.com` are sent to `fuji1`.

In the second example, packets destined for `ftp.mycompany.com` are sent to `fuji1`, but if `fuji1` fails, all the packets for `ftp.mycompany.com` are sent to `fuji2`.

The third and fourth examples are like example 1 and 2 but `fuji3` acts as the preferred gateway node for `ftp.mycompany.com`.

3.2.4.3 PRIVATE addresses

The PRIVATE internal interface is used for communication among the nodes in the cluster. When you configure a PRIVATE address for a node, SIS creates a virtual interface on the interconnect network. The PRIVATE address cannot be accessed from the outside.

PRIVATE has the following syntax:

```
PRIVATE IPAddress netmask PRIVATENode
```

For example, if `fuji2` and `fuji3` are the only nodes of the cluster, then they can be defined in the configuration file as follows:

```
PRIVATE 192.168.0.1 255.255.255.0 fuji2
PRIVATE 192.168.0.2 255.255.255.0 fuji3
```

The following is strongly recommended for PRIVATE configurations:

- Configure a PRIVATE interface for each node.
- Select IP addresses from one of the available non-routable subnets.

3.3 Configuring with Cluster Admin

The following sections discuss how to create the SIS configuration file using the Cluster Admin GUI, which has a wizard for creating a new configuration file (refer to Section “Cluster Admin” for a brief description of Cluster Admin).

3.3.1 Starting SIS

Start SIS with Cluster Admin as follows:

1. Log in to Cluster Admin.
2. Display the SIS GUI main window and choose one of the following:
 - a) Load a pre-saved configuration file
 - b) Start the SIS configuration wizard

3.3.2 Logging in to Cluster Admin

Cluster Admin requires a functional Web-Based Admin View installation on all nodes of the cluster. Refer to either the *PRIMECLUSTER Installation Guide (Solaris)* or the *PRIMECLUSTER Installation Guide (Linux)* for information on installing and configuring the Web-Based Admin View GUI. Once you have installed and configured Web-Based Admin View, do the following:

1. Open your browser.
2. Type in the address of the URL for Web-Based Admin View by entering its address into your Java-enabled browser. For example, if the initial node chosen is `fujj2`, enter the following URL to pull up the Web-Based Admin View java-based applet:

```
http://fujj2:8081/Plugin.cgi
```

After the Java applet has fully loaded, a small top-level window requires you to log in to the GUI (see Figure 3).

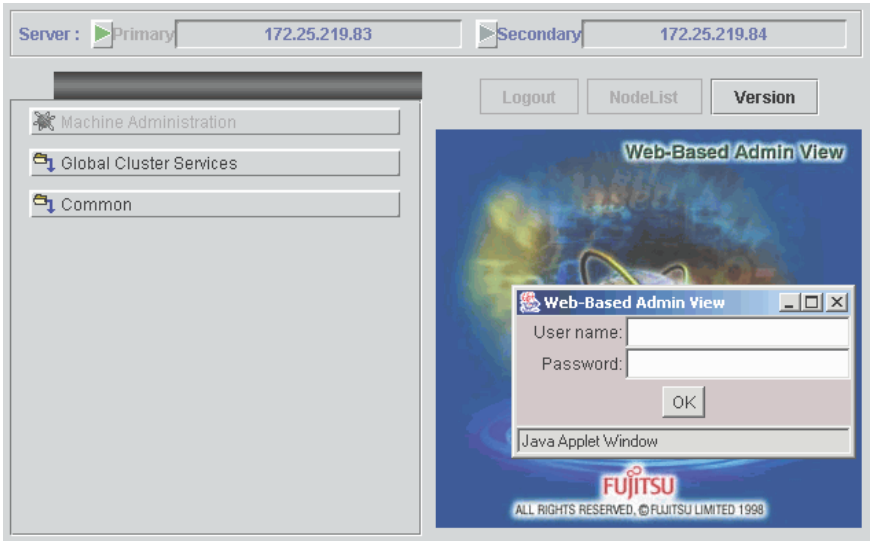


Figure 3: Login window

Enter the administrator name and the password for Web-Base Admin View and click *OK*. Once you have logged in, select *Global Cluster Services*. The window for selecting Cluster Admin appears (see Figure 4).

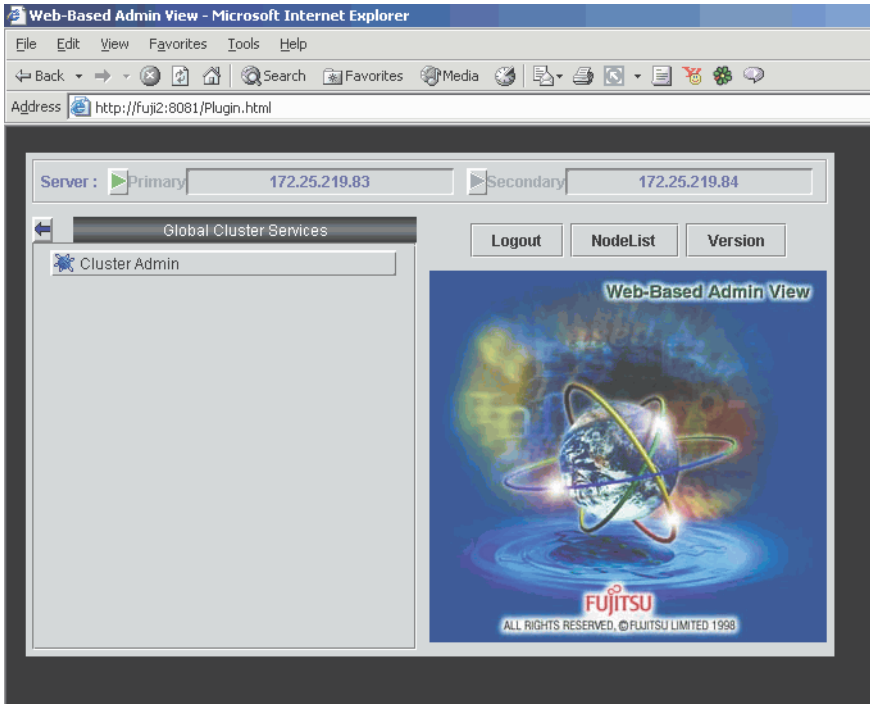


Figure 4: Global Cluster Services window

Select *Cluster Admin*, and then in the resulting pop-up window, choose a node for initial connection (see Figure 5).

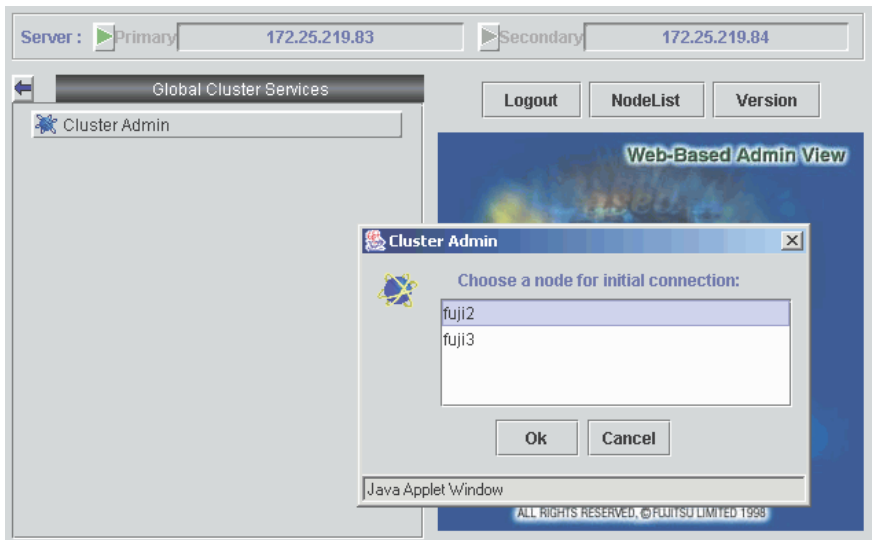


Figure 5: Choosing a node for initial connection

The Cluster Admin main window appears (see Figure 6).

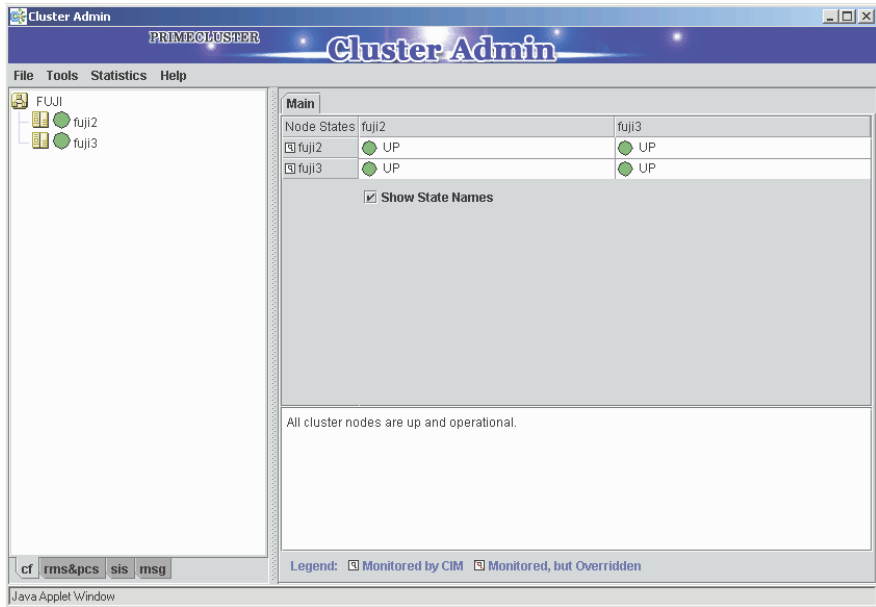


Figure 6: PRIMECLUSTER Cluster Admin window

3.3.3 Displaying the SIS GUI main window

To display the SIS GUI main window, click on the *SIS* tab in the bottom of the left panel to start the SIS portion of the Cluster Admin GUI.

If SIS is configured and running, you will see the SIS main window displaying the configuration. In this case, refer to Section “Administering with Cluster Admin” for further details.

If SIS is not configured on the node, click on the *sis* tab, select *Start* from the *Tools* menu, and the *SIS Startup Menu* pop-up window appears (see figure Figure 7) with the following options:

- *Search all configuration files*—Allows you to load a pre-saved configuration file and start SIS with that configuration. This option is also available after you have stopped SIS by means of the *Stop* option (refer to the Section “Tools”).
- *Start configuration wizard*—Invokes the SIS configuration wizard to create a new configuration.

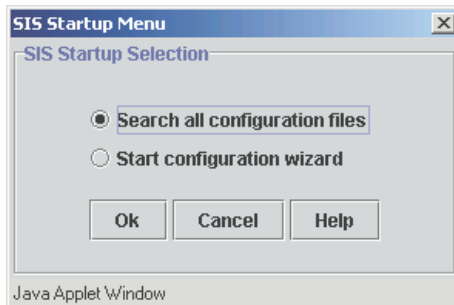


Figure 7: SIS startup selection

3.4 Creating a new configuration file

After selecting *Start configuration wizard* from the *SIS Startup Menu*, the welcome window appears (see Figure 8). Click on the *Next* button to move on to the next window. Refer to the Section “Concepts” for details on all of the terminology used.

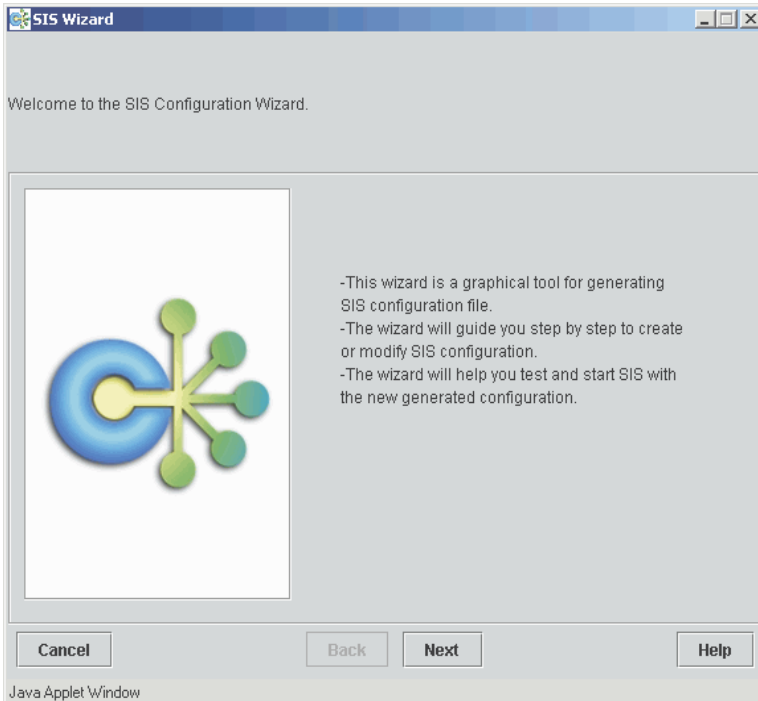


Figure 8: SIS Wizard welcome window

The windows that follow the welcome window are divided into two panels as follows:

- Left panel—Shows the configuration as you create it, either in a tree format (see Figure 9) or as a textual file (see Figure 10), depending on the tabs you choose at the bottom of the panel.
- Right panel—Takes the configuration input and reflects it in the left-hand panel.

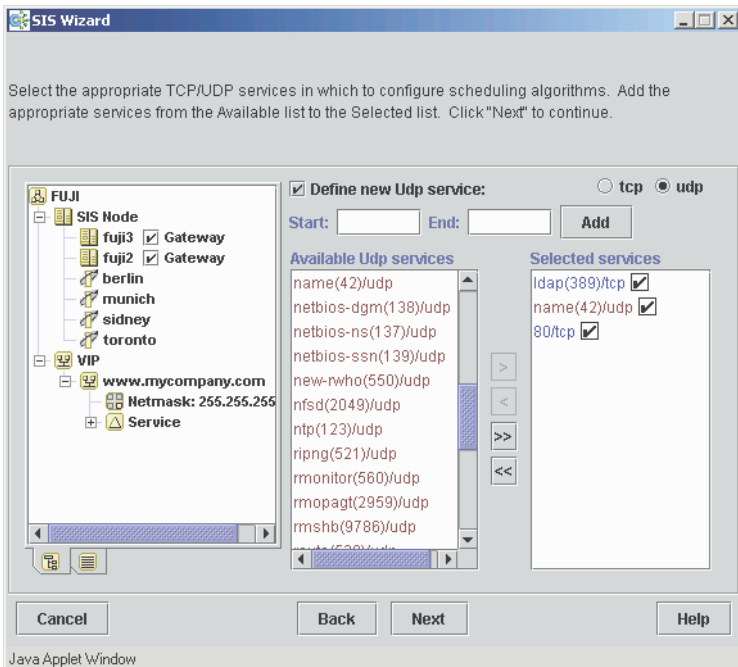


Figure 9: Viewing the configuration in the tree format

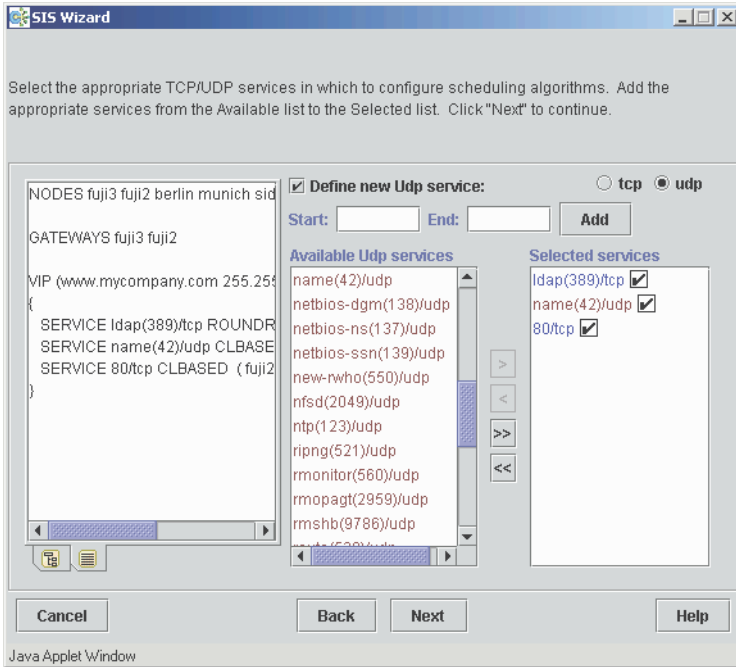


Figure 10: Viewing the configuration as a text file

3.4.1 Adding nodes

The window for adding nodes to the SIS cluster follows the welcome window (see Figure 11). This window is divided as follows:

- The first column, *CF Nodes*, shows the available CF nodes.
- The second column, *Selection*, is for selecting which nodes you want to be members of the SIS cluster.
- Two buttons on the right side of the window allow you to select or deselect all nodes in the first column as SIS nodes.
- The *Satellite node...* button opens an additional window for selecting satellite nodes.

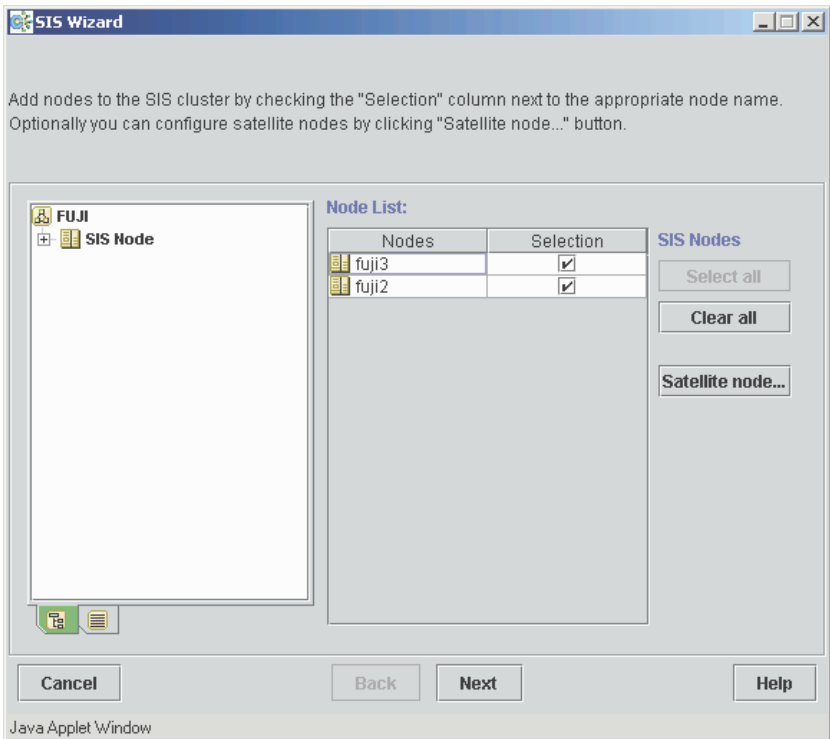


Figure 11: Selecting nodes

For example, if you select the nodes `fuj12` and `fuj13` to be in the SIS cluster, the configuration will have the following entries:

```
NODES fuj12 fuj13
```

3.4.2 Configuring satellite nodes

Click on the *Satellite node...* button to add satellite nodes. The *Satellite Node Definition* window appears (see Figure 12).

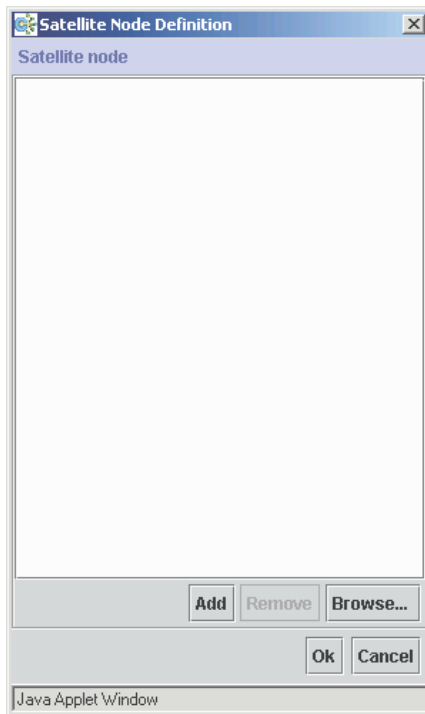


Figure 12: Adding satellite nodes

You can manually add satellite nodes into the SIS cluster by clicking on the *Add* button or you can choose from a list of available nodes by clicking on the *Browse* button.

Manually adding nodes

Click on *Add* and the *Add Satellite Nodes* popup window appears. Enter a set of node names into the text area. Use spaces or commas to separate node names (see Figure 13).

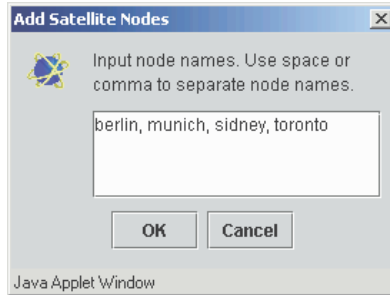


Figure 13: Manually adding satellite nodes

Click on the *OK* button to add the satellite nodes to the configuration.

Selecting nodes from file list

This feature allows Cluster Admin to evaluate regular configuration files or files with the `.sat` extension. These files must be in `/etc/opt/SMW/SMWdtcp`.

The *Browse* option (see Figure 14) extracts a list of node names by reading either selected files, all files of a selected node, or all files on all nodes of the cluster. From the list you can select nodes as satellite nodes that are not currently known as core nodes.

In regular configuration files the `NODES` line is evaluated, while in a file with the `.sat` extension, all uncommented lines are expected to contain node names that are separated by commas or by white space.

Click on *Browse...* and the file browser appears (see Figure 14).

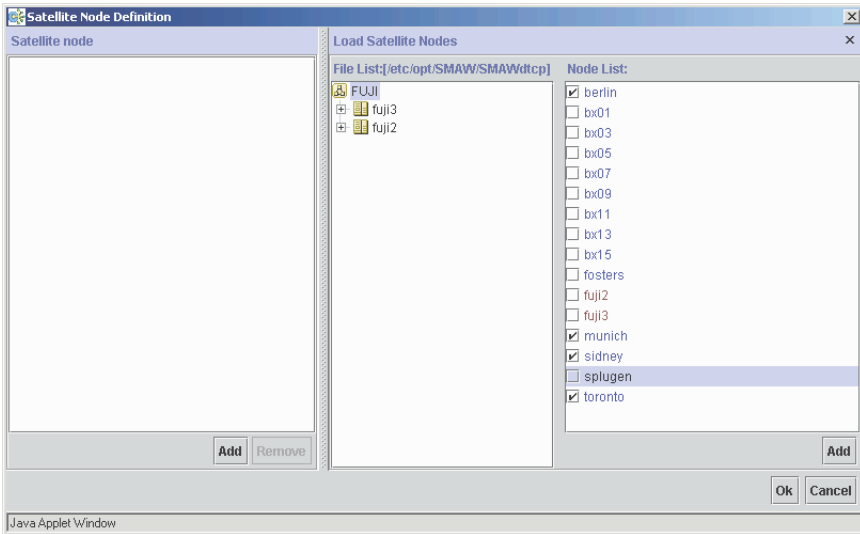


Figure 14: Browsing for satellite nodes

The window is divided as follows:

- Left panel—Shows the satellite nodes after you have added them to the configuration.
- Middle panel—Contains the cluster-node tree. If you click on the cluster name, the window shows all the nodes found in the cluster. Expand the tree further to reveal the SIS configuration files for each node. If you click on a node or a file, the available satellite nodes are listed in the right panel.

- Right panel—Lists the available satellite nodes. By default, all possible satellite nodes are selected. The selection check box is disabled if a node is already a CF or SIS node or if it is already in the node list.

Select the satellite nodes that you want for your configuration and click on the *Add* button to add them to the configuration. The nodes appear in the left panel. Click on the *Ok* button to go to the next window.

The window for selecting nodes reappears (see Figure 15).

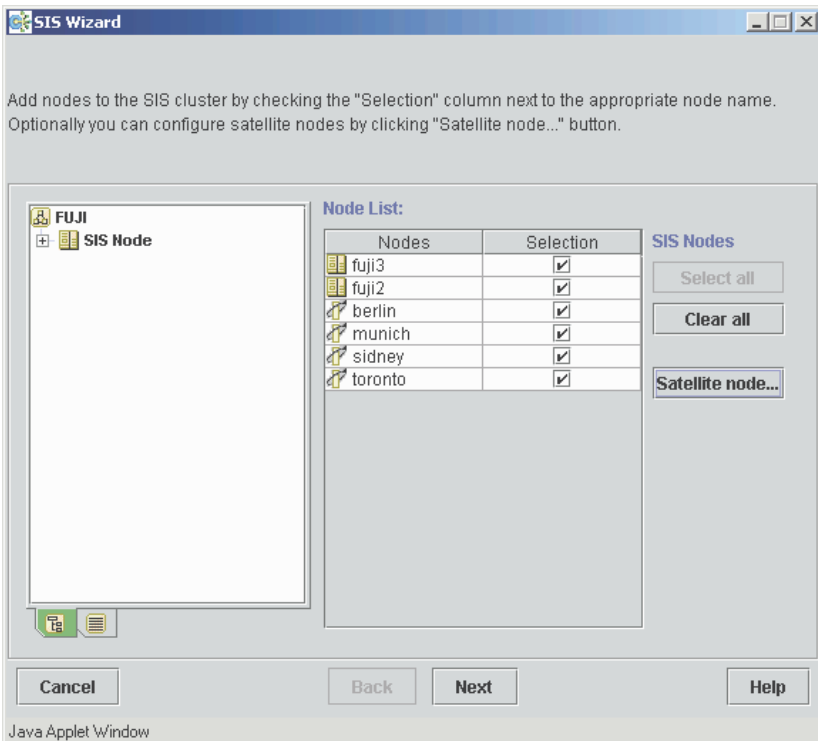


Figure 15: SIS Wizard node selection window

Click on the *Select all* button to add the nodes. Click on the *Next* button to create the VIP, PROXY, and PRIVATE interface definitions.

3.4.3 Defining VIP, PROXY, and PRIVATE addresses

The base window for defining virtual SIS addresses is shown in Figure 16.

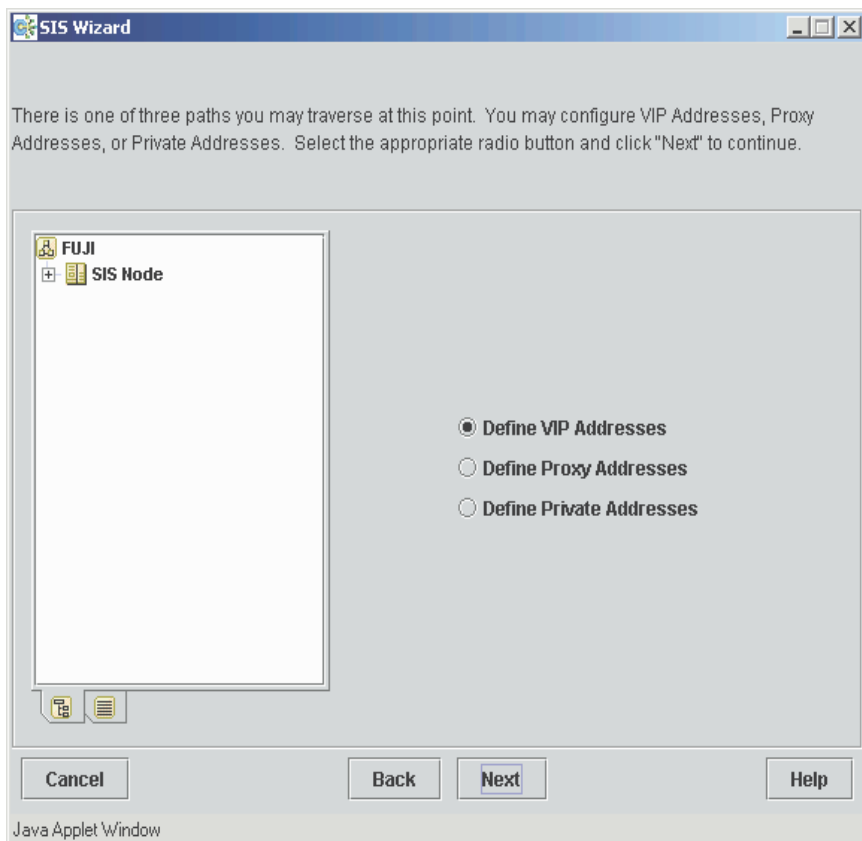


Figure 16: Define VIP, PRIVATE, and PROXY addresses

The following options are available:

- *Define VIP Addresses*
- *Define Proxy Addresses*
- *Define Private Addresses*

3.4.3.1 VIPs

Creating a VIP includes the following procedures:

- Defining one or more VIP addresses
- Defining services for each VIP
- Assigning a schedule to each service
- Checking the result
- Saving and activating the configuration

Defining the VIP address

To define a VIP, select the *Define VIP Addresses* radio button and click on the *Next* button (see Figure 16). The window for creating new VIP addresses appears (see Figure 17).

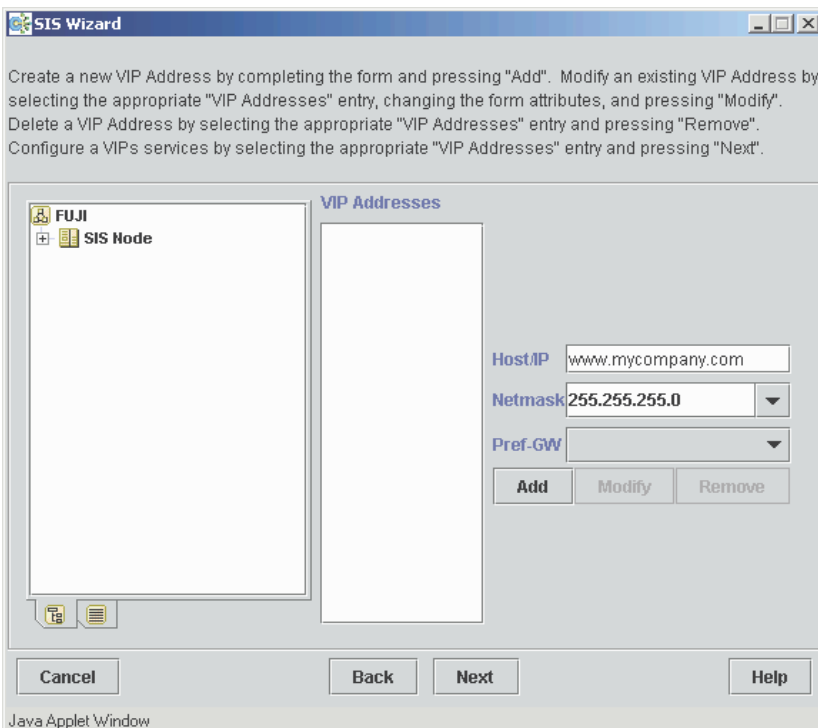


Figure 17: Define VIP addresses window

Define VIPs as follows:

1. Enter the *Host/IP* address as a resolvable host name or IP address (see Figure 17).
2. Select or edit the appropriate netmask in the *Netmask* field.
3. Select a node name in the *Pref GW* field if desired.
4. Click the *Add* button to add the VIP to the *VIP addresses* column.
5. Repeat Steps 1 through 4 for each VIP.

The resulting window should look similar to Figure 18.

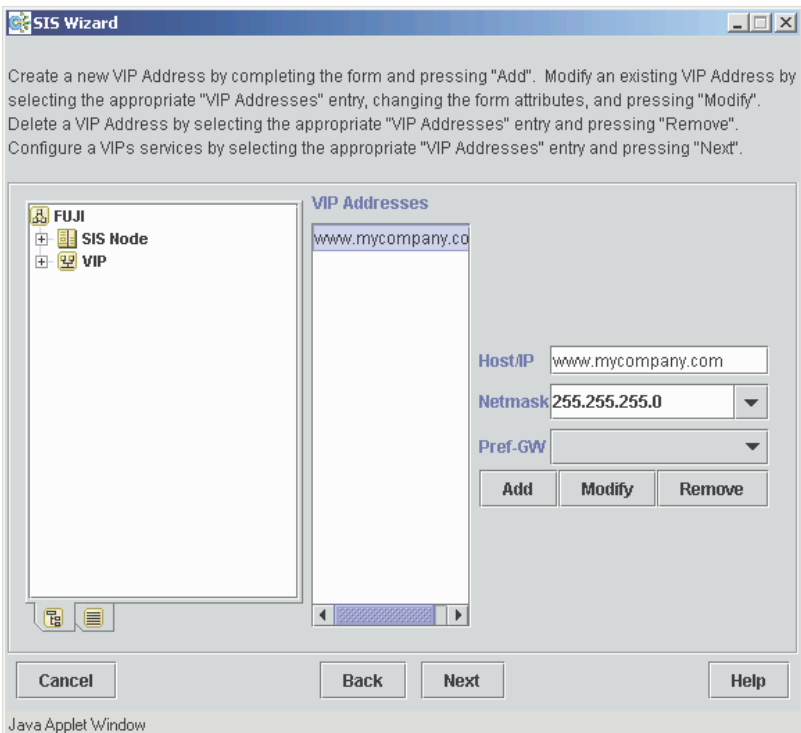


Figure 18: Define VIP Addresses window with VIP added

Defining services for the VIP

To add services to a VIP definition, select the VIP from the *VIP Addresses* list and click on the *Next* button. This takes you to the *select services* window (see Figure 19).

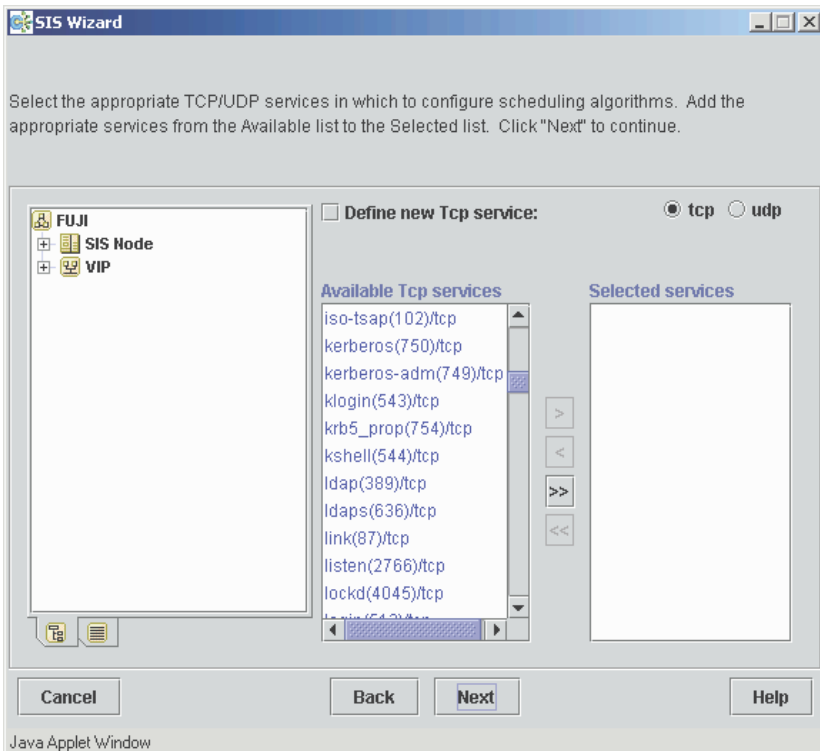


Figure 19: Select services window

A list of available TCP services from `/etc/services` is already in the *Available TCP services* list (see Figure 19). To see the UDP services, select the *udp* radio button. You can define additional services as follows:

1. Click the *Define new TCP/UDP service* check box. Additional fields appear (see Figure 20).
2. Enter the starting and ending port numbers.
3. Click the *Add* button. The newly defined service is added to the *Available services* list.

i If you define a new TCP or UDP service but some of the new port numbers are already present in the *Available TCP services* or *Available UDP services* lists, Cluster Admin will resolve the collision when the configuration file is generated.

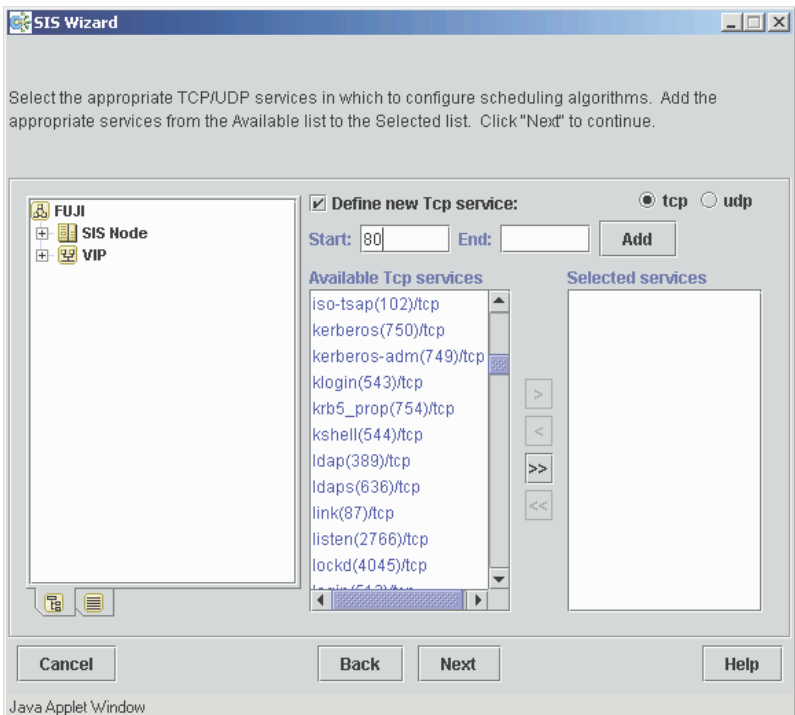






Figure 20: Define new service window

Next, select one or more of the available TCP or UDP services as follows:

1. Mark a service from the *Available services* list.
2. Click on the right arrow  button to move the marked service to the *Selected services* window.
3. Repeat the process for each service that you want to define.

Use the left arrow  button to remove services from the *Selected services* list. Click on the double right  button to select all services in the *Available services* list. To remove all the *Selected services*, click on the double left arrow  button.

When you have finished your selections, you will see a window similar to Figure 21.

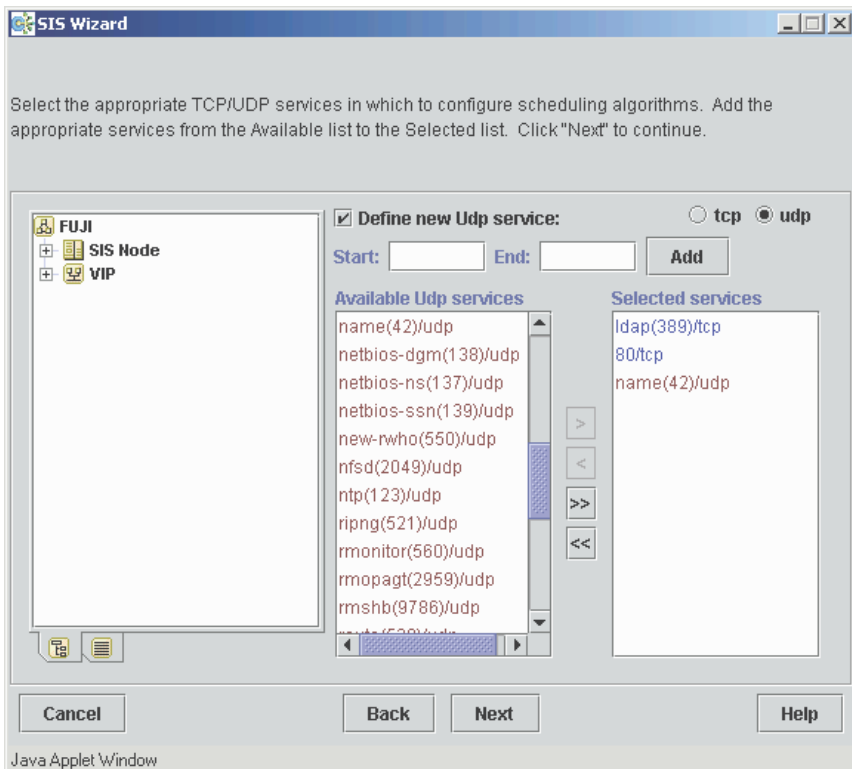


Figure 21: Selected services

Once you have finished, click on the *Next* button, which takes you to the window to pick services (see Figure 22).

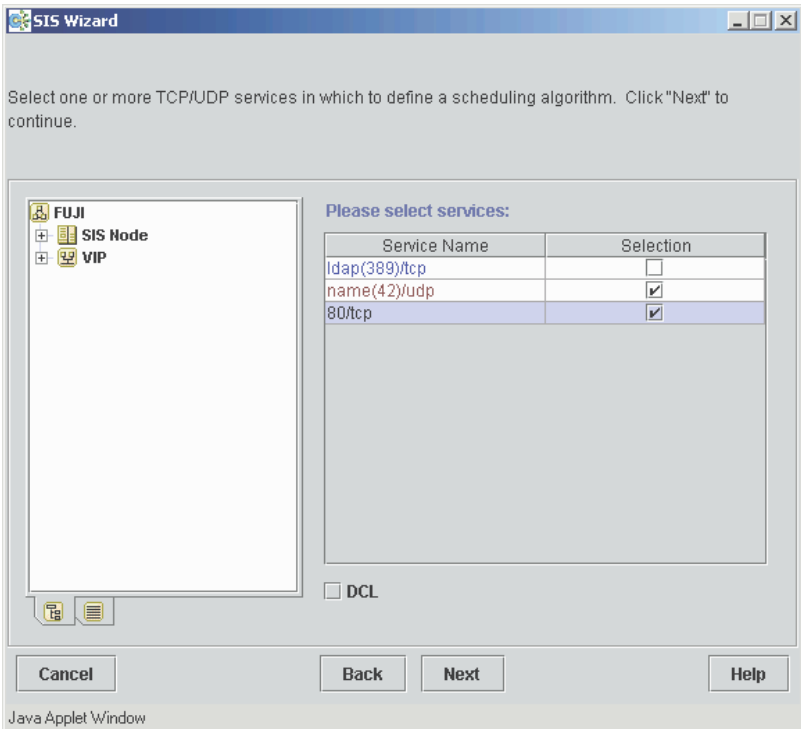


Figure 22: Pick services window

The *pick services* window lists all the services you selected. To define schedules to the services, select them by clicking on the *Selection* check box. If you want them to have the same scheduling algorithm and node lists, you can select more than one service; otherwise, select them one at a time. Click on the *DCL* check box if required, and then click the *Next* button.

Next you see the *define scheduling algorithm* window (see Figure 23).

Define schedules for each service as follows:

1. Select a scheduling algorithm from the menu list at the top. Changing the scheduling algorithm can cause the data entry fields to change as well.
2. Select the service nodes by clicking on the corresponding check boxes in the *Service Nodes* list.
3. If required, click on the desired *Failover Nodes* check boxes. You can only select nodes that have not been selected as service nodes.
4. Click on the arrow pointing down to add the definition to the area below. Remove a definition by marking it in the area below and clicking on the up arrow.
5. After you are finished with the definition for this service, click on the *Next* button, which takes you back to the *select services* window (see Figure 24). A configured service is now recognizable by a check box.

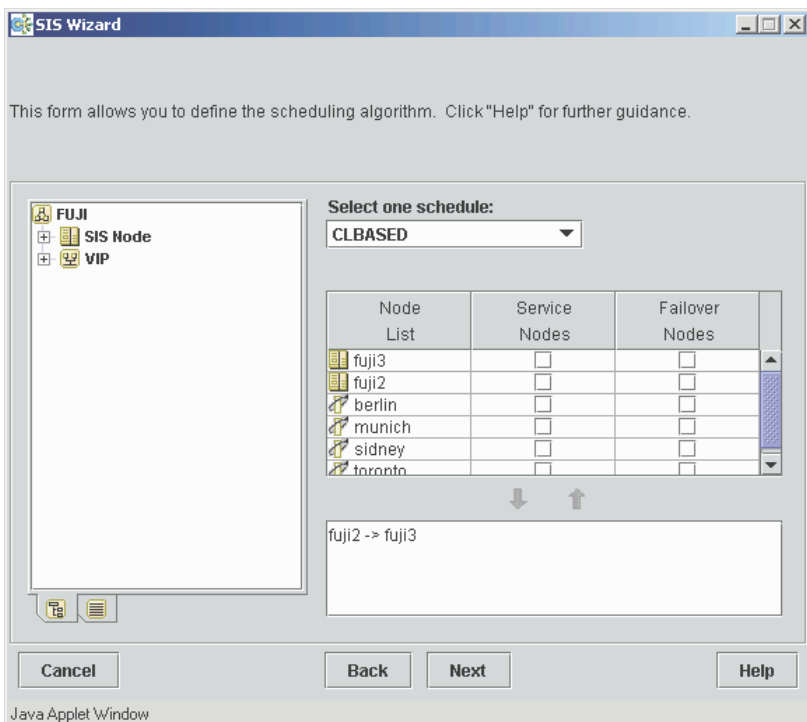


Figure 23: Define scheduling algorithm window

You can add more services here if desired. Continue configuring the services until they are all done. Once you are done configuring services, click *Next* to go to the following window.

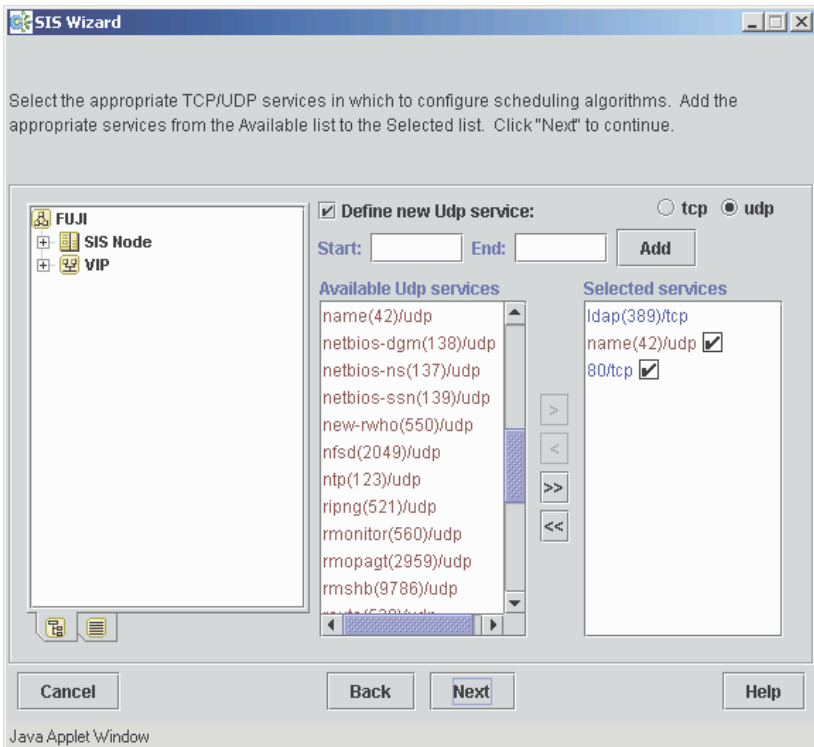


Figure 24: Select services—configured

In the *complete configuration* window you can choose to continue defining SIS addresses or to complete the configuration (see Figure 25). The left-hand panel shows the current configuration file. Click *Next* to add more interface definitions the file. Refer to the Section “Completing the configuration” for information on saving the file.

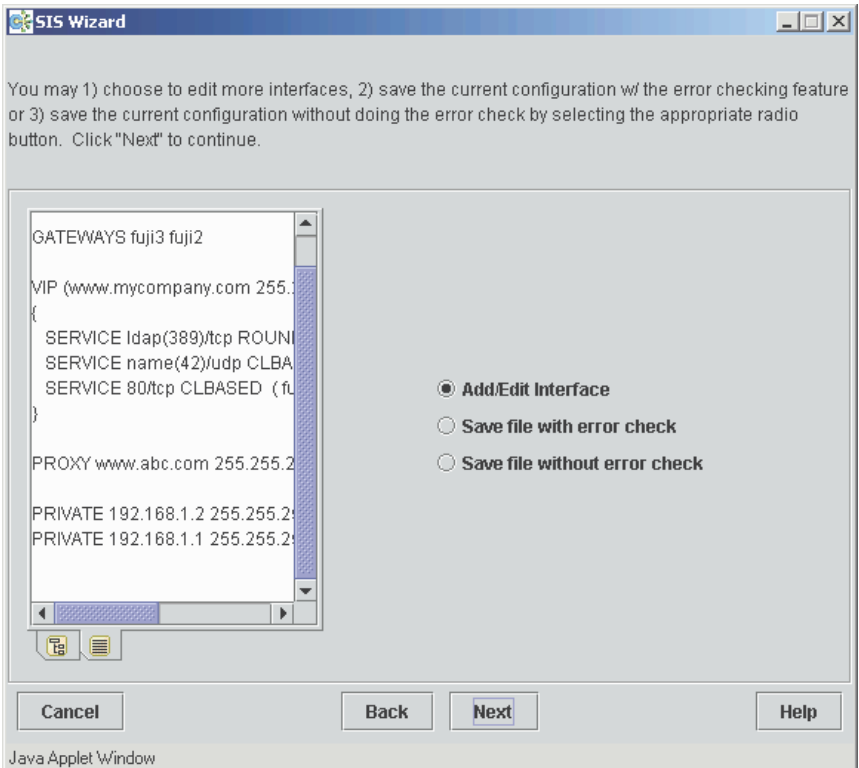


Figure 25: Complete configuration window

The window that follows allows you to continue defining VIPs or to add PROXY or PRIVATE addresses (see Figure 26). Select *Define VIP Addresses* and click *Next* to define more VIPs and to check your configuration. PRIVATE and PROXY addresses should be defined once the VIP address definitions are complete.

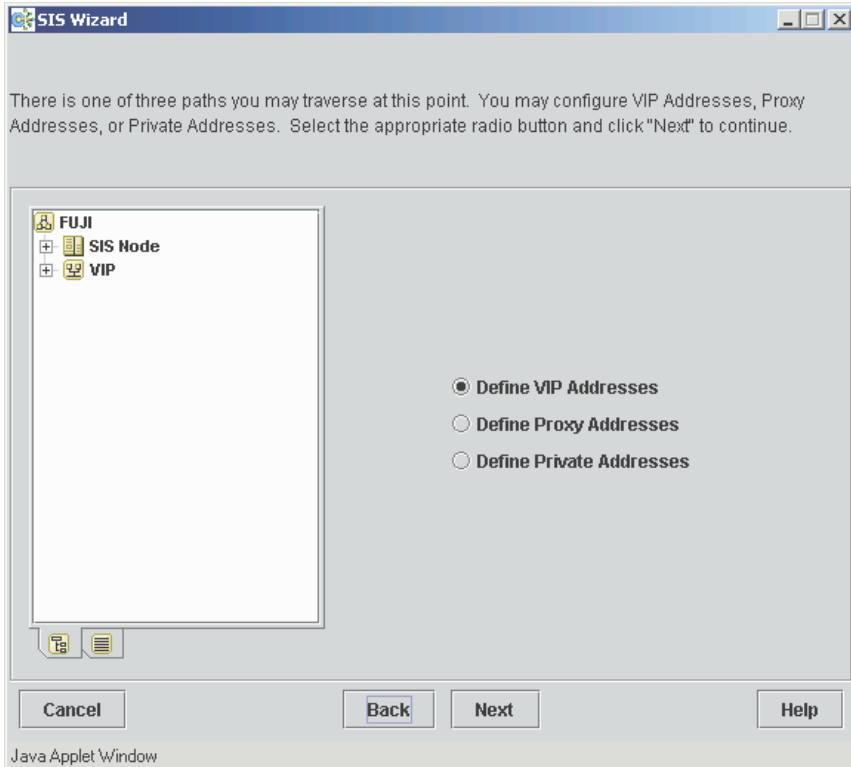


Figure 26: Define addresses window

Returning to the *define VIP addresses* window shows that there is a checkmark in the check box next to each fully configured VIP (see Figure 27). If you want to modify a VIP, select the VIP, click on *Modify*, and make your changes as necessary. Complete all VIPs in a similar way. If you want to remove an already configured VIP, select the VIP, and click on *Remove*.

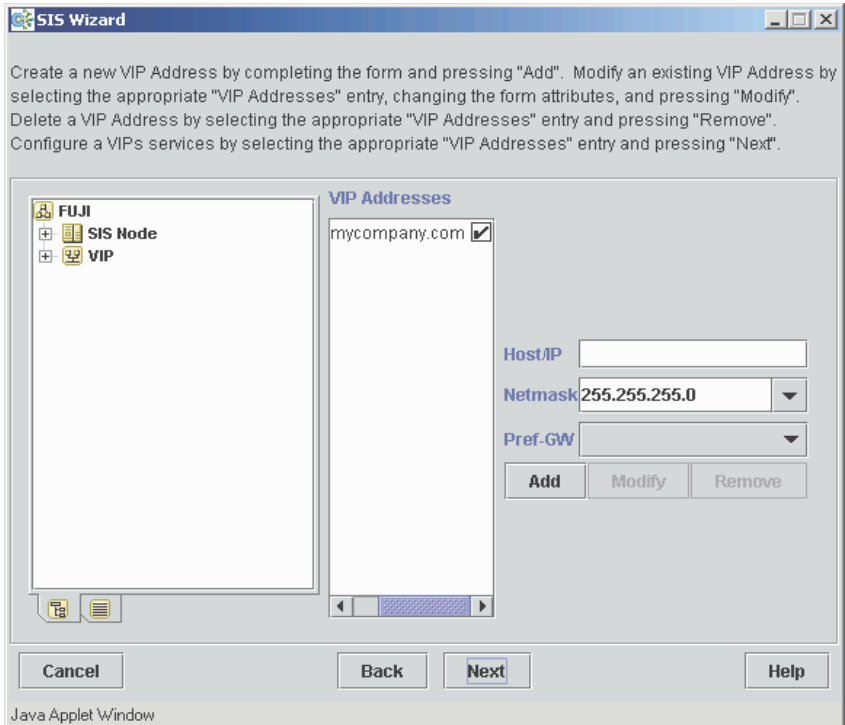


Figure 27: Fully configured VIP window

After completing the configuration of all VIPs, our sample configuration file now has the following contents:

```
NODES fuji2 fuji3 berlin munich sidney toronto

GATEWAYS fuji2 fuji3

VIP (www.mycompany.com 255.255.255.0)
{
    # 389/tcp --> ldap/tcp
    SERVICE 389/tcp ROUNDROBIN ( fuji3 FAILOVER berlin )
    # 42/udp --> name/udp
    SERVICE 42/udp CLBASED ( fuji2 FAILOVER fuji3 )
    SERVICE 80/tcp CLBASED ( fuji2 FAILOVER fuji3 )
}
```

FTP Notes

Due to the special limitations when configuring `ftp` for a VIP (refer to Section “VIP”), the GUI recognizes FTP configurations and enforces the following special rules:

- `CLBASED` and `KEEPLOCAL` scheduling are allowed without restrictions.
- For all other scheduling algorithms, the GUI inserts `DCL` (even if you do not check the `DCL` button).
- When you try to configure `ftp` together with other services or ports, you get a pop-up warning and the configuration will not be possible until you deselect the other services from this window.
- The configuration of port 20 will not be discovered until the syntax check is selected during the save or activate the configuration process.

PRIVATE and PROXY providers

Create a PRIVATE or PROXY provider as follows:

1. Enter the *Host/IP* address.
2. Select or edit the appropriate netmask in the *Netmask* field.
3. For PROXY addresses, you can select the following (see Figure 29):
 - Preferred gateway by selecting a node from the *Pref-GW* pull-down list.
 - Failover node from the list of available nodes
4. Click on the *Add* button.

Select the *Define Private Addresses* option and click on *Next* to bring up the *define PRIVATE addresses* window (see Figure 28).

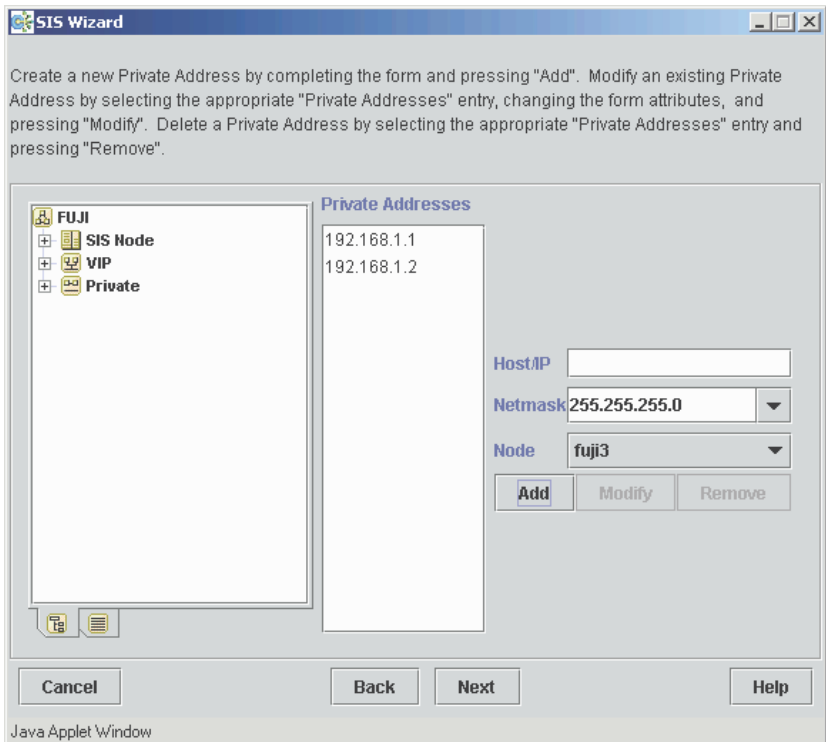


Figure 28: Define PRIVATE addresses window

After entering the required data, save the information by selecting *Add*. Figure 28 shows the window to define a PRIVATE address and Figure 29 shows the window to create a PROXY address.

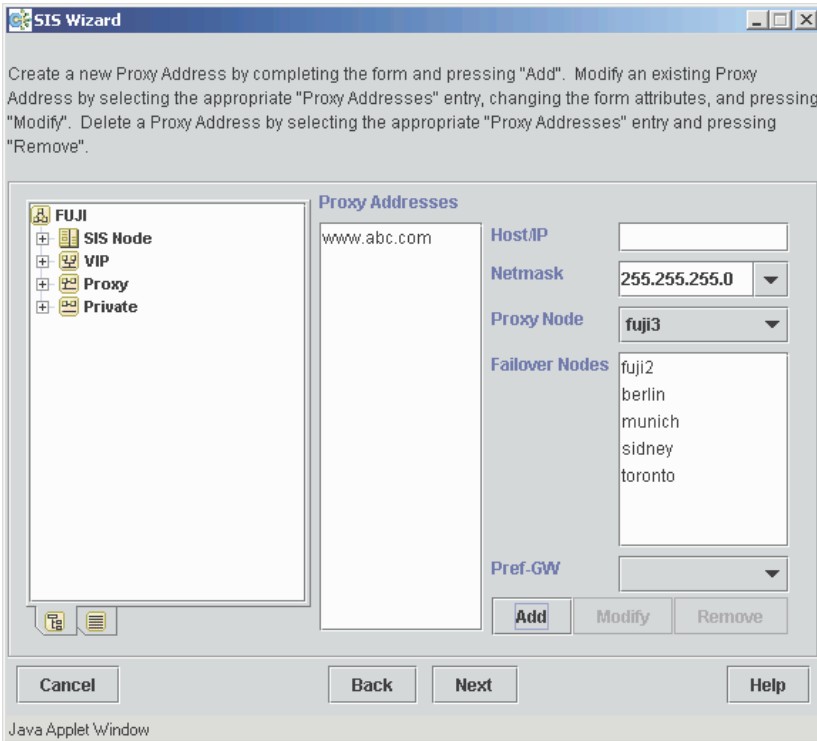


Figure 29: Create PROXY addresses

Once the PROXY or PRIVATE configurations are done, the information is added to the list on the left portion of the window and the *complete configuration* window appears (see Figure 30).

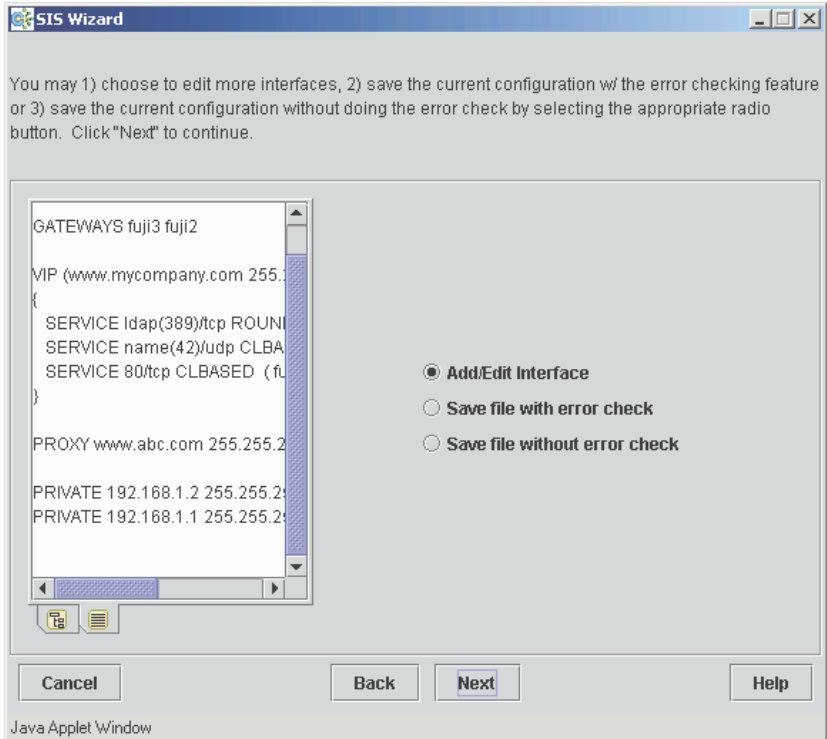


Figure 30: Creating additional addresses

We recommend that you first assign a PRIVATE address for each node in the cluster. After you have finished creating all of the PRIVATE addresses, you can create PROXY addresses in the same manner. This is done by clicking on the *Next* button to continue creating the configuration file and choosing from the following options (see Figure 31):

- *Add/Edit Interface*
- *Save file with error check*
- *Save file without error check*

Select *Add/Edit Interface* to continue adding VIP, PRIVATE, and PROXY addresses. After you have finished all configurations, you will return to the *complete configuration* window to save the file with or without an error check (see Figure 31). The example configuration file would look similar to the following:

```
NODES fuji2 fuji3

GATEWAYS fuji2 fuji3

VIP (www.mycompany.com 255.255.255.0)
{
    SERVICE ldap/tcp ROUNDROBIN fuji2 fuji3
    SERVICE name/udp CLBASED (fuji2 FAILOVER fuji3)
    SERVICE 80/tcp CLBASED (fuji2 FAILOVER fuji3)
}

PROXY www.abc.com 255.255.255.0 fuji2

PRIVATE 192.168.1.1 255.255.255.0 fuji3
PRIVATE 192.168.1.2 255.255.255.0 fuji2
```

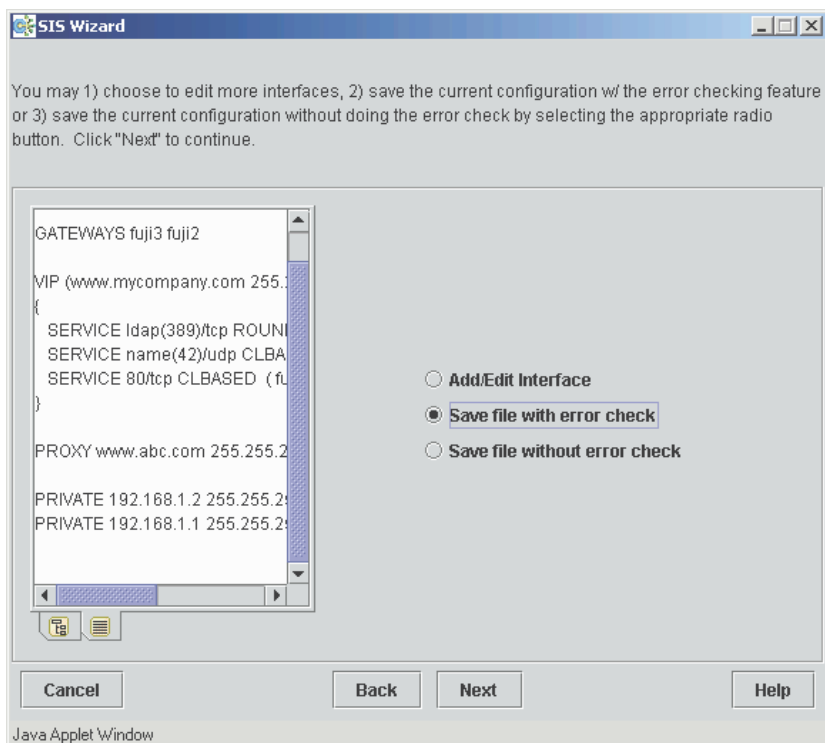


Figure 31: Complete configuration window

You should now refer to Section “Completing the configuration” for information on saving your file.

3.4.4 Completing the configuration

After creating and defining all of the VIP, PROXY, and PRIVATE interfaces the window in Figure 31) appears. If you have finished adding all of the interfaces, select the *Save file with error check* radio button and click *Next*. This returns a pop-up box with the error check result of either *OK* (see Figure 32) or *Error* (see Figure 33).

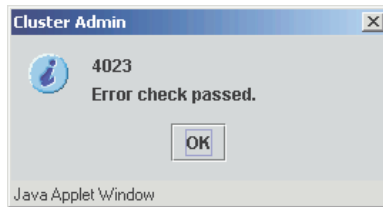


Figure 32: Error check result window: OK



An error only occurs if an IP address or service name is not resolvable or if you are editing a configuration file that already contained an error.

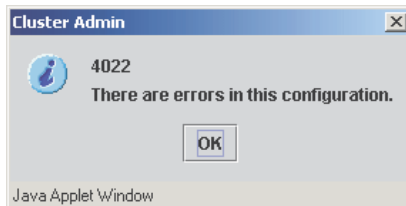


Figure 33: Error check result window: Error

Click on the *OK* button to see the details of the syntax check in the right-hand side panel (see Figure 34). The left panel shows the configuration that you created.

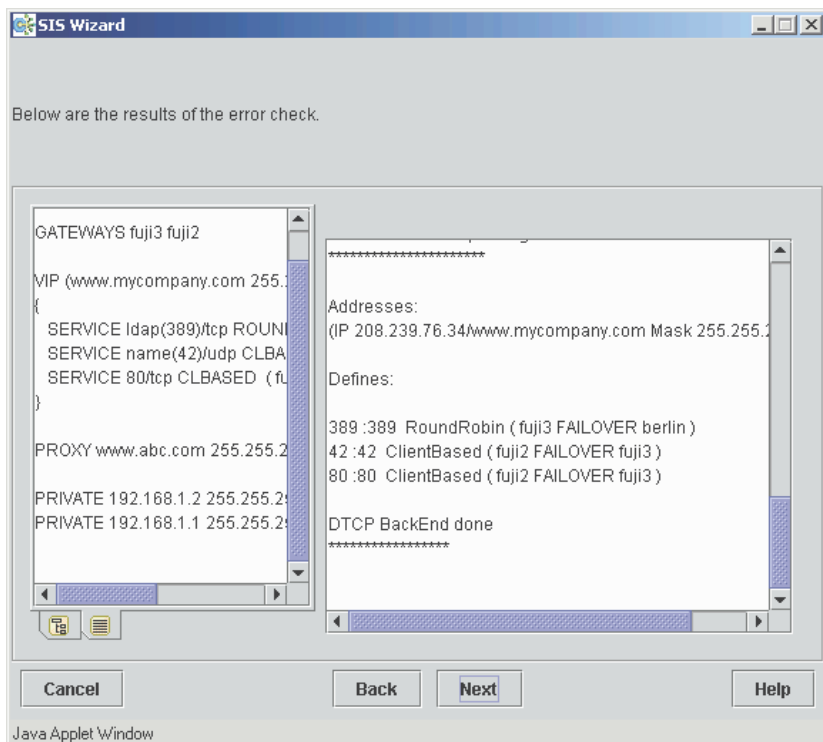


Figure 34: Error check result window

Click on the *Next* button to go to the *save configuration* window (see Figure 35). Enter a file name in the *File name* text box. Select the *Start SIS with the new created configuration* check box if you also want to activate the new configuration. Click on the *Next* button.

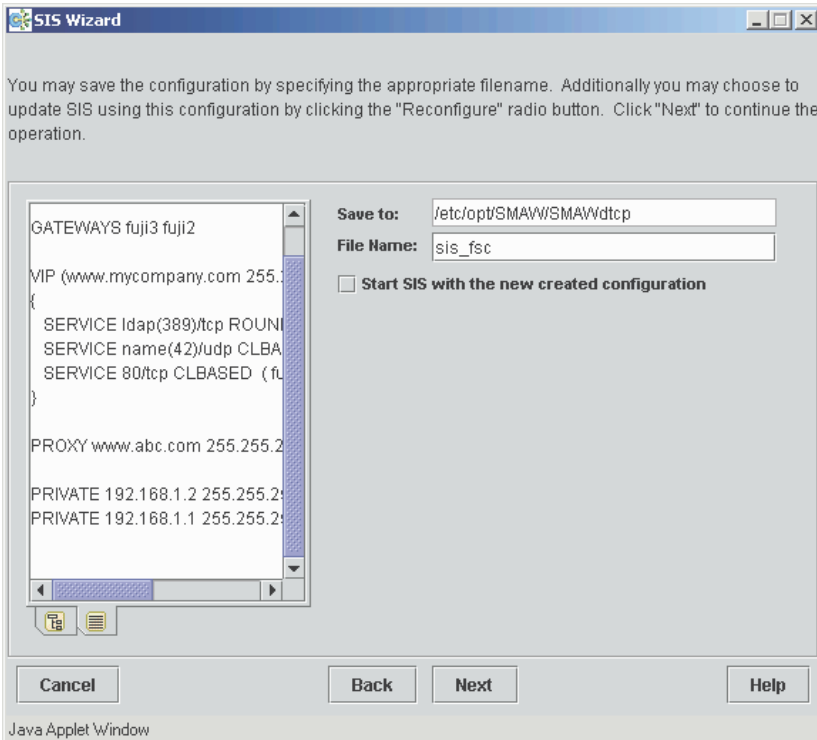


Figure 35: Save configuration window

Clicking on the *Next* button after entering the filename takes you to the *SIS Wizard completed* window as shown in Figure 36. Click the *Finish* button to exit from the wizard.

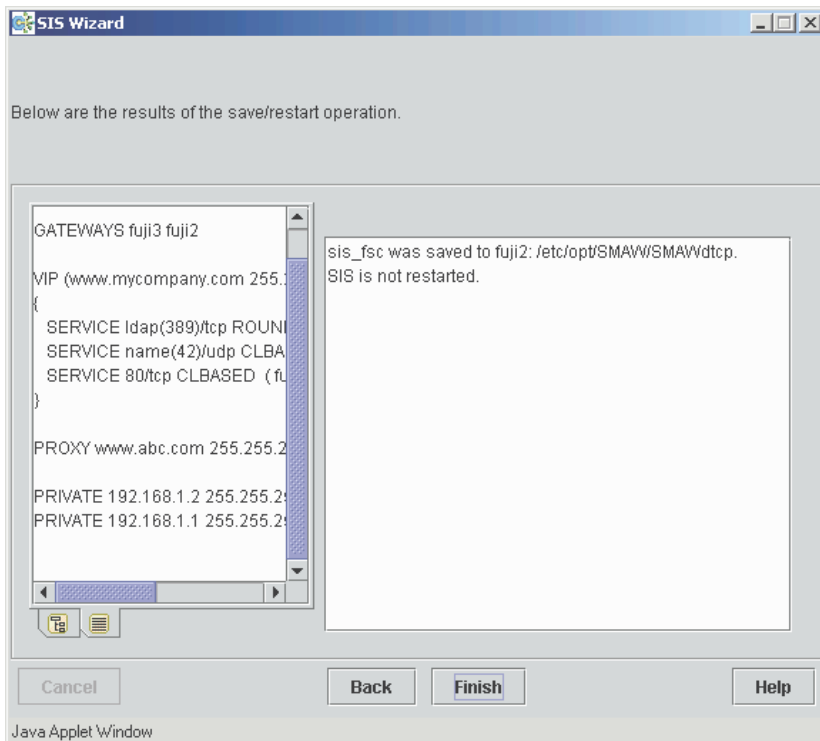


Figure 36: SIS Wizard completed window

If you chose to start SIS with the new configuration, then you have the option to either reconfigure or rebuild the configuration (see Figure 37) The *Reconfigure* option is available only if SIS is running.

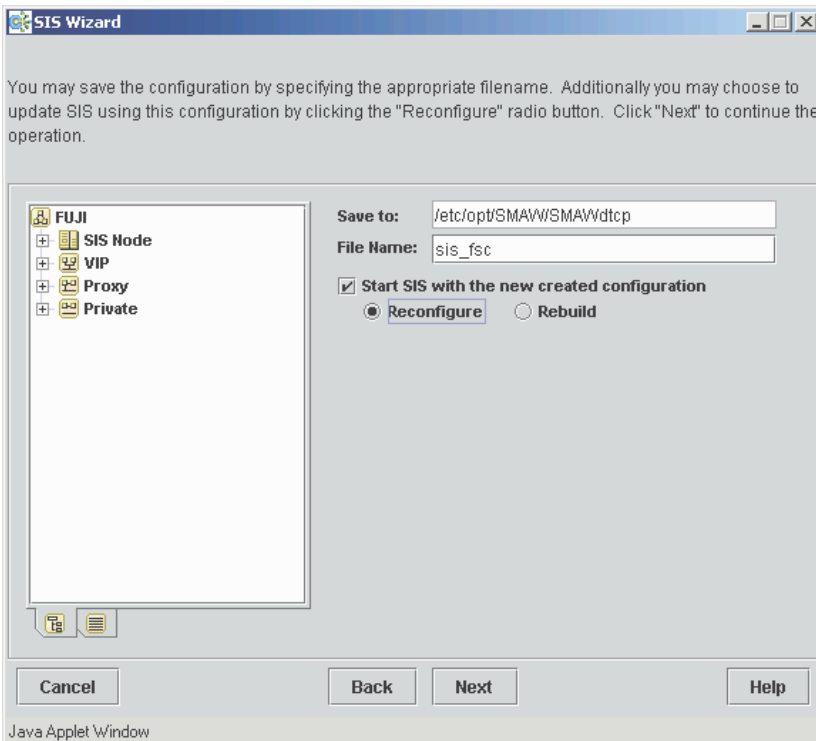


Figure 37: Restarting SIS from the save configuration window

i SIS does not allow the `NODES` list to be changed by reconfiguration. If your new configuration does not contain the same nodes as the currently running SIS configuration, you must choose the *Rebuild* option.

Reconfiguring SIS by means of the *Reconfigure* radio button has the following effects:

- Primary and backup database nodes do not change
- TCP connections survive
- UDP pseudo connections are terminated

Click on the *Next* button, and the *SIS Wizard completed* window appears (see Figure 38). Click on *Finish* to exit from the wizard.

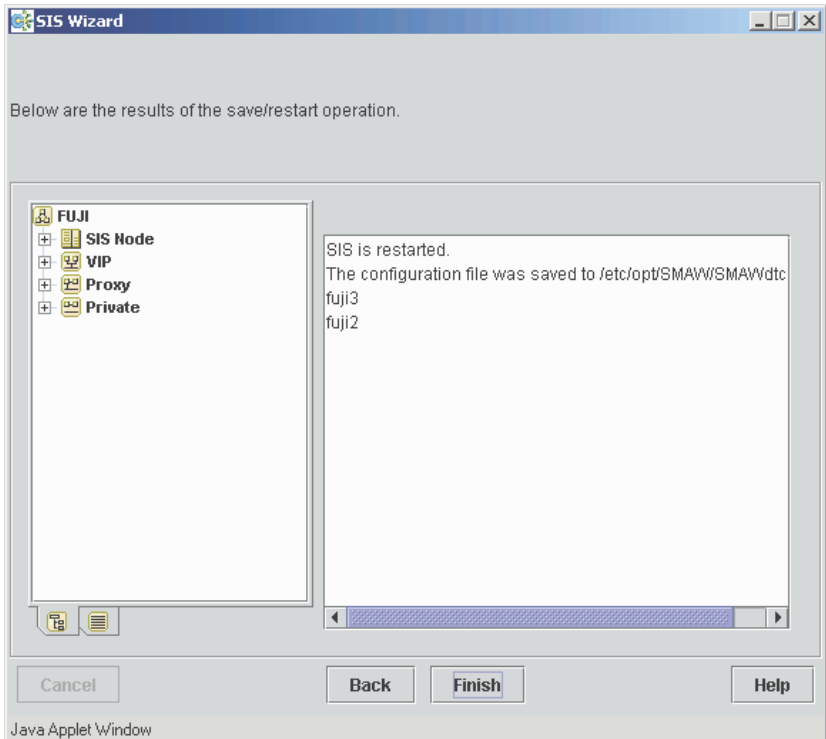


Figure 38: Restart from the SIS Wizard completed window

If you select the *Rebuild* option, you are given additional options as follows:

1. Select the primary database node from the *Primary DB Node* list.
2. Select backup database nodes from the *Secondary DB Nodes* list by clicking on their selection check boxes.

When you click on the *Next* button, all SIS nodes are started with the configuration specified. If a SIS configuration is running on the cluster, SIS will first be stopped on all active nodes and then restarted according to your configuration.

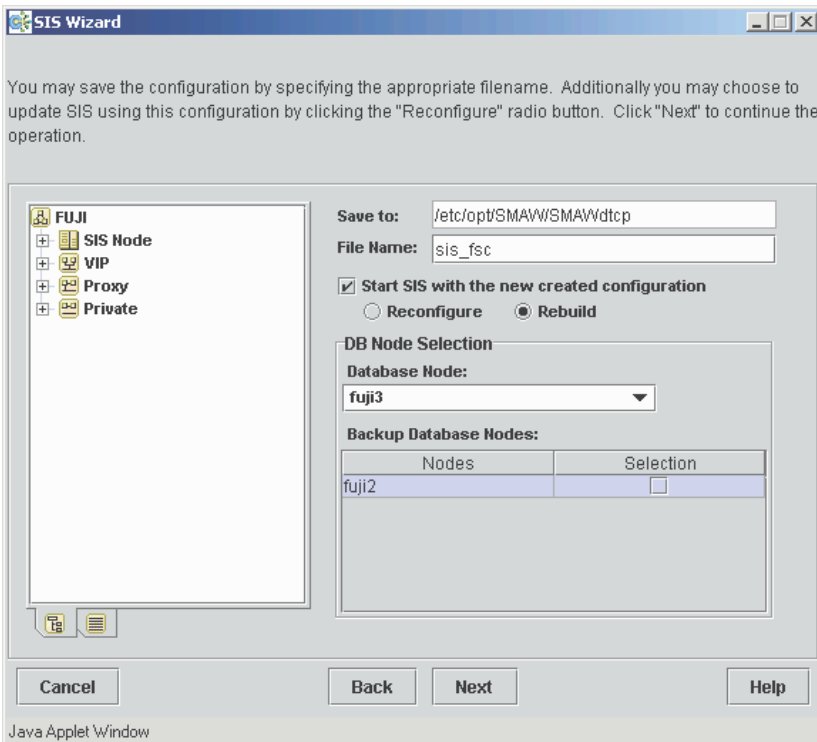


Figure 39: Rebuild option

Next, the *SIS Wizard completed* window appears (see Figure 36). Click the *Finish* button to exit from the SIS Wizard.

3.5 Starting with an existing configuration file

Instead of selecting the SIS Wizard, you can choose the *Search all configuration files* from the *SIS Startup Menu*. The *SIS Clusterwide Startup* window appears (see Figure 40).

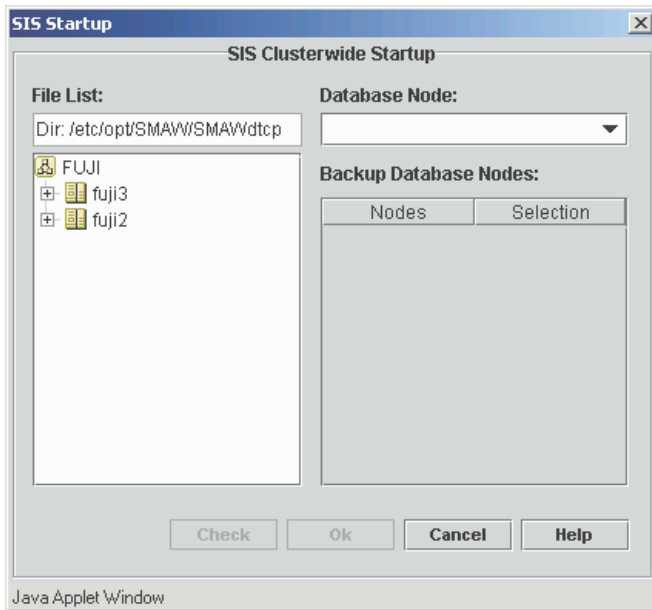


Figure 40: SIS Clusterwide Startup window

To select an existing configuration file, perform the following:

1. Click on the plus symbol of the node with the desired file.
2. Select the file by clicking on it.

The primary *Database Node* selection list is populated (see Figure 41).

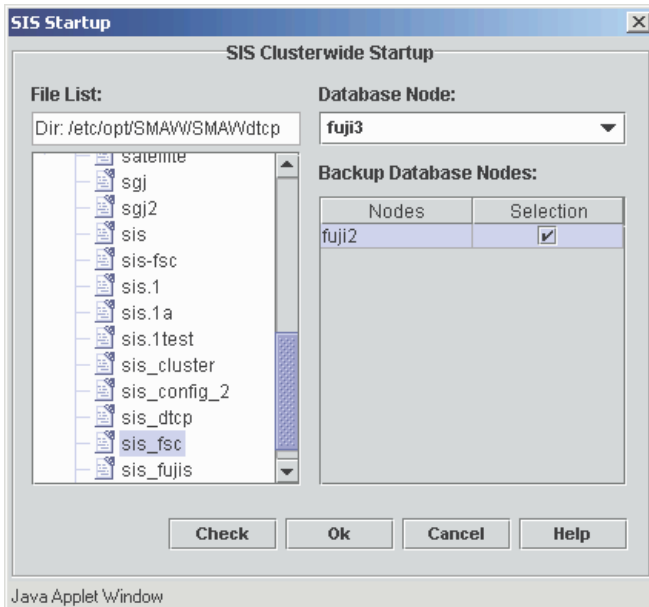


Figure 41: Primary Database Node selection list

To enable the configuration file perform the following:

1. Select the primary database node from the list from the selection box.
2. Choose one or more nodes from the *Backup Database Nodes* table by clicking on the *Selection* check box for the node.
3. Click on *Check* to check the file for syntax errors.
4. A small window indicates whether the error check passed or failed. Select *OK* to return to the *SIS Clusterwide Startup* window.
5. Select the *OK* button. SIS starts on all nodes in the correct order.

3.6 Examples and configuration files

The following are example scenarios and their corresponding configuration files. These examples are provided to illustrate how an actual configuration file would look; however, an actual configuration file should use the IP addresses for the cluster being configured, not the ones that appear in these examples.

Example 1

This example shows an LDAP server (`ldap.mycompany.com`) on a two-node cluster (`fujii2` and `fujii3`). The LDAP database is replicated on both nodes. Both nodes reply to LDAP queries alternately for the TCP protocol. For UDP, `fujii2` replies to all queries, and if `fujii2` fails, `fujii3` will reply to all UDP queries.

The sample configuration is as follows:

```
NODES fujii2 fujii3
GATEWAYS fujii2 fujii3
VIP (ldap.mycompany.com 255.255.255.0)
{
SERVICE ldap ROUNDROBIN fujii2 fujii3
SERVICE ldap/udp ROUNDROBIN (fujii2 FAILOVER fujii3)
}
PRIVATE 192.168.1.1 255.255.255.0 fujii2
PRIVATE 192.168.1.2 255.255.255.0 fujii3
```

Example 2

This example shows a four-node cluster with two powerful systems, `fujii3` and `fujii4`, and two not so powerful systems, `fujii1` and `fujii2`. `fujii1` and `fujii2` answer simple http queries and `fujii3` and `fujii4` answer the secure https requests. In addition, one customer is hosting a very large web site (`www.abc.com`) and another customer is hosting a smaller web site (`www.def.com`). To duplicate this example, you need to use a text editor like `vi`.

The sample configuration is as follows:

```
# Powerful systems
M400_NODELIST= fuji3 fuji4;

# Not so powerful systems
M200_NODELIST= fuji1 fuji2;

#All Nodes
NODELIST= M400_NODELIST M200_NODELIST

NODES NODELIST
GATEWAYS NODELIST

# Vip definitions
VIP (www.mycompany.com 255.255.255.0)
{
    SERVICE http ROUNDROBIN M200_NODELIST
    SERVICE https SYSLOAD M400_NODELIST
}

#PROXY definitions
PROXY www.abc.com 255.255.255.0 (fuji3 FAILOVER fuji4)
PROXY www.def.com 255.255.255.0 fuji1

#PRIVATE definitions
PRIVATE 192.168.1.1 255.255.255.0 fuji1
PRIVATE 192.168.1.2 255.255.255.0 fuji2
PRIVATE 192.168.1.3 255.255.255.0 fuji3
PRIVATE 192.168.1.4 255.255.255.0 fuji4
```

4 Satellite nodes

This chapter details the requirements for setting up satellite node configurations.

This chapter discusses the following:

- The Section “Overview” introduces the concepts of satellite and core nodes.
- The Section “Software” details the software requirements for satellite node configurations.
- The Section “Hardware” specifies the types of hardware required for satellite node configurations.
- The Section “Setting up satellite nodes” discusses how to setup satellite node configurations.

4.1 Overview

Satellite nodes run a version of SIS that supports most of the same network services as the complete package. All the scheduling algorithms work the same as standard nodes.

Satellite nodes differ from regular SIS nodes; therefore, regular SIS nodes are known as core nodes. The major differences between satellite nodes and core nodes are as follows:

- Satellite nodes cannot also be members of the CF cluster.
- Satellite nodes cannot act as database, backup database, or gateway nodes.

Figure 42 illustrates a five-node SIS configuration with three core nodes, a Linux satellite node, and a Windows 2000 satellite node in the cluster.

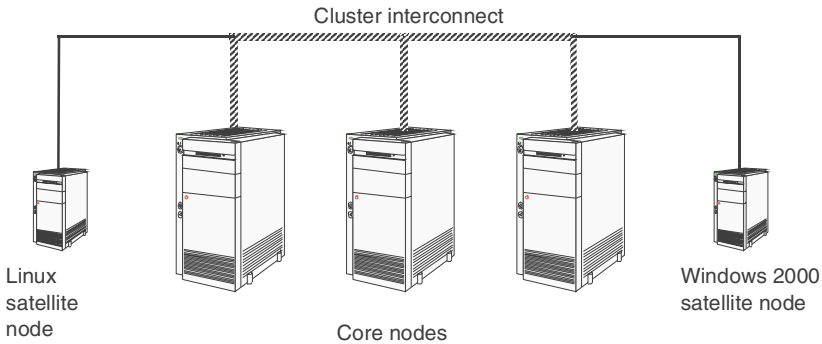


Figure 42: SIS cluster with satellite nodes

4.2 Software

SIS satellite nodes run on the following hardware and software combinations:

- Standard Intel-based hardware running the following versions of Linux:
 - SuSe SLES 7 and SLES 8
 - United Linux
 - Redhat Advance Server 2.1
- Standard Intel-based hardware running the following versions of Windows:
 - Windows 2000 Advanced Server
 - Windows 2000 Server
 - Windows 2000 Professional
- Solaris 8 OE or Solaris 9 OE

4.3 Hardware

The hardware requirements for SIS clusters with satellite node are as follows.

- The recommended satellite node platform is the Fujitsu Siemens Computers BX300 blade server.
- It is expected that both core nodes and satellite nodes have a direct connection to the internet.
- Core nodes discover satellite nodes by broadcasting. For this reason, at least one interface for all the nodes in the SIS cluster should be on the same Ethernet segment.

Figure 43 shows a supported configuration in which all the core nodes and the satellite nodes share a common Ethernet segment.

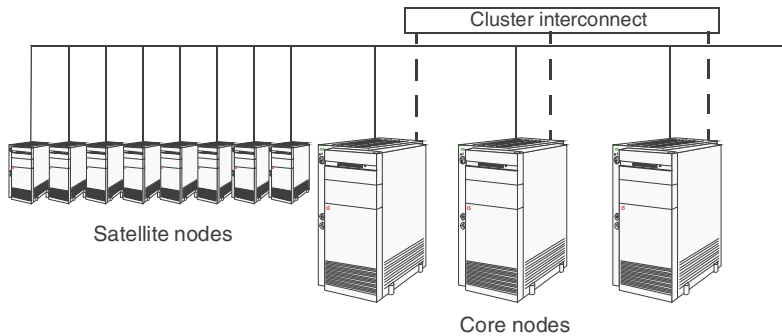


Figure 43: Supported satellite node hardware configuration

4.4 Setting up satellite nodes

- i** Refer to the Section “Configuring satellite nodes” for details on how to configure satellite nodes by means of Cluster Admin.
- i** You need to set the cluster name for a satellite node if you have more than one cluster on the same subnet.

Satellite nodes are ready to join a cluster after they are booted, and they are removed from service when they are shutdown. In addition, you can expel or activate satellite nodes from the core cluster as desired. Satellite nodes will try to join any cluster that broadcasts. However, a cluster will accept a satellite node only if it is explicitly configured in the `NODES` list of the cluster.

The default for satellite nodes is to try and join the first cluster broadcast they receive. To prevent a satellite node from joining multiple clusters, specify a cluster name as described in the sections that as follow.

4.4.1 Specifying cluster name on Windows systems

To specify the cluster name for a Windows satellite node, open the *Network and Dialup Connections* window. Depending on your Windows options, this is normally found in the *Settings* option of the Windows *Start* menu (see Figure 44) or as a link in the *My Computer* window (not shown).

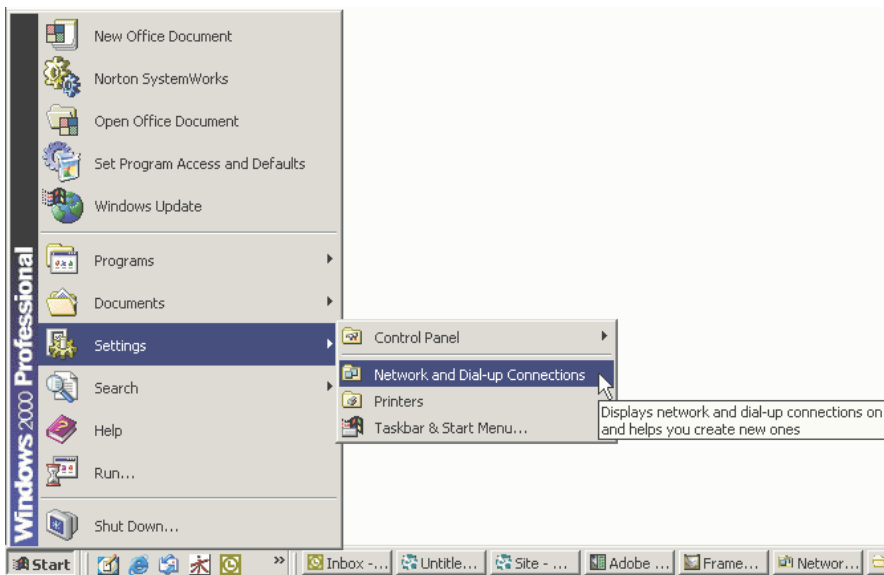


Figure 44: Windows Network and Dial up Connections

Locate the Local Area Connection for the Fujitsu Siemens VIP adapter (see Figure 45).

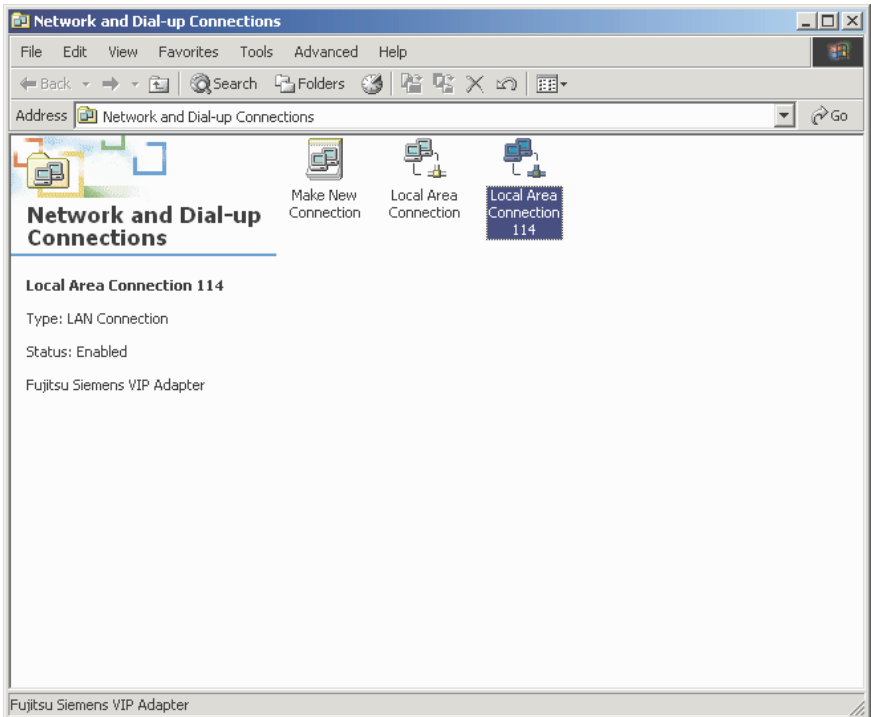


Figure 45: Local Area Connection for Fujitsu Siemens VIP Adapter

Double-click on the Local Area Connection icon.

The Status window for the VIP adapter appears (see Figure 46).

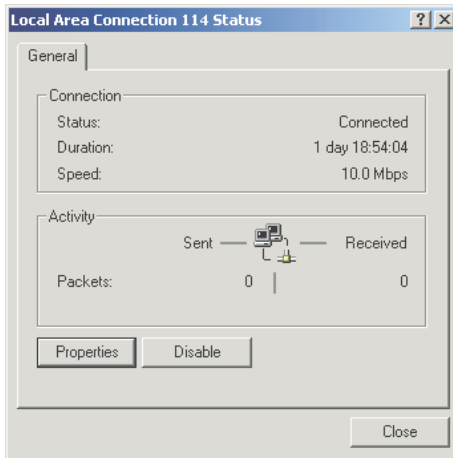


Figure 46: VIP adapter status window

Click on the *Properties* button.

The properties window for the VIP adapter Local Area Connection appears (see Figure 47).

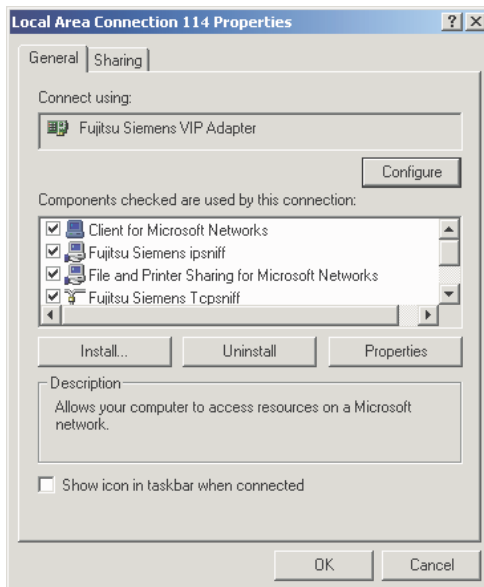


Figure 47: VIP adapter properties window

Click on the *Configure* button.

The *Fujitsu Siemens VIP Adapter Properties* window appears (see Figure 48).

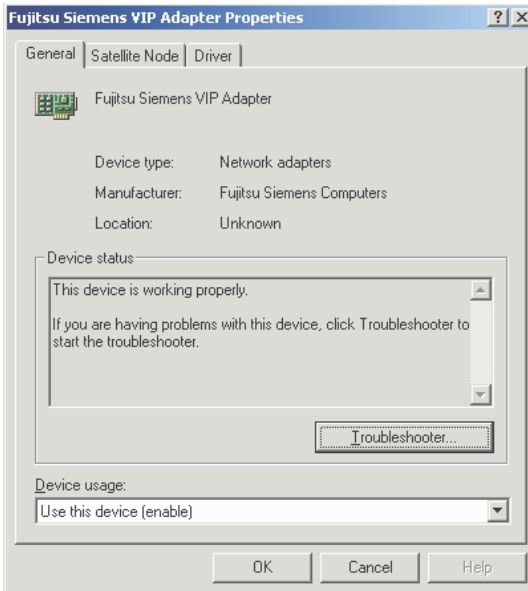



Figure 48: Fujitsu Siemens VIP Adapter Properties window

Click on the *Satellite Node* tab.

The *Satellite Node* properties window for specifying the cluster name appears (see Figure 49).

-  If the *Connect to any cluster* check box is selected, then the cluster name will not be set, and the satellite node will join the first cluster from which it receives a broadcast message.

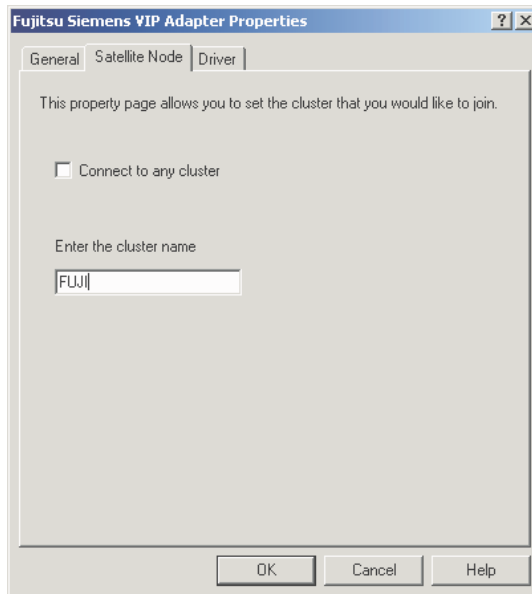


Figure 49: Satellite Node properties window

Enter the cluster name in the text box and click on the *OK* button. Now the Windows SIS satellite node will only join the specified cluster.

4.4.2 Specifying cluster name on Linux systems

For Linux systems, set the `CLUSTERNAME` variable in the `/etc/init.d/dtccp` file as follows:

1. Change the `CLUSTERNAME` variable to `CLUSTERNAME=PRIMECLUSTER`.



The cluster name should be in all upper case letters.

2. Reboot the system.

4.4.3 Specifying cluster name on Solaris OE systems

For Solaris systems, add or edit the following line in the `/etc/system` file as follows:

1. set `dtccp:sis_cluster_name="cluster_name"`

cluster_name is the name of the CF cluster.

2. Reboot the system.

5 NIC failover

This chapter describes SIS Network Interface Card (NIC) failover, including how SIS recognizes a failure, how it responds, and what you need to do to configure the SIS NIC failover module.

This chapter discusses the following:

- The Section “Introduction” introduces the problems and solutions for a SIS NIC failover.
- The Section “SIS NIC failover module” details how the SIS NIC failover module works and how to use it.

5.1 Introduction

For the SIS cluster, the NIC card on the gateway node is extremely important. If the gateway node cannot communicate with other systems, then the Virtual Interface Provider (VIP) interface, which depends on this gateway node, will be non-functional. Additionally, if a NIC fails on a service node, network services handled on that service node will not respond to client requests.

These conditions result in a single point of failure for the SIS cluster (at least to a VIP). To prevent this, SIS needs to use other mechanisms to recognize NIC failure and act on it. Standard system interfaces do not provide any method for detecting a NIC failure. The SIS NIC failover module recognizes NIC card failures and keeps the SIS cluster functional.

The following solutions to this problem are possible:

- SIS NIC failover module—The SIS gateway node will switchover to another node. This is tightly integrated with SIS and does not require a second or a passive NIC in each of the nodes.
- Synfinity Link/GLS—This provides NIC redundancy by having a passive NIC on the same node.
- Solaris IP multipathing or Linux NIC bonding—Both these methods can work with an active/active model or an active/passive model on the same node.

SIS has been tested and works in conjunction with all the solutions mentioned above. The SIS NIC Card failover module solution is optimal because it does not require a second NIC and is tightly integrated with SIS.

5.2 SIS NIC failover module

The SIS NIC failover module monitors NICs on its member nodes and reacts accordingly if it cannot contact one of its monitored NICs.

5.2.1 Monitoring

The basic mechanism used is the `ICMP_ECHO_REQUEST` (ping). A set of *trusted hosts* that are likely to be available may additionally be provided in a file.

Example:

A cluster consists of SIS nodes `fuj1`, `fuj2`, and `fuj3`. Monitoring is done as follows:

1. Each node monitors the NICs on the following node by doing regular pings to that node; for example, `fuj1` pings `fuj2`, `fuj2` pings `fuj3`, `fuj3` pings `fuj1`.
2. If the pings from `fuj1` to `fuj2` fail, then `fuj1` tries to ping a host from its list of trusted hosts. If this succeeds, then `fuj1` assumes that `fuj2`'s NIC has failed and takes action.
3. If `fuj1` cannot reach any other host, it will do nothing.

5.2.2 Failover mode

If `fuj1` determines that the NIC on `fuj2` has failed, the SIS NIC failover module takes actions as in the following example:

Example:

1. `fuj1` starts monitoring the external interfaces of `fuj3` while continuing to try to reach `fuj2`.
2. If `fuj2` is a gateway node, the next available node on the same subnet is ordered to be the gateway node.

3. Outgoing packets from `fuji2` are rerouted through the next available node on the same subnet as the failed NIC. This is done only for data packets that have virtual addresses managed by SIS as the source IP address. Native connections through that NIC will be dropped unless the SIS cluster was configured for NIC redundancy.

5.2.3 Restore actions

If the failed NIC on `fuji2` is restored, then the SIS NIC failover module will take the following actions as in the following example:

Example:

1. The gateway node does not revert back to the original gateway node, `fuji2`.
2. Outgoing packets of `fuji2` are no longer rerouted.
3. `fuji1` will again only monitor `fuji2`, while `fuji2` resumes monitoring `fuji3`.

5.2.4 Starting and restarting

The NIC failover monitoring is done by a userland daemon. This daemon is started when the package is installed. It is also started automatically by the `dtcp rc` script when the system is rebooted.

You can use the command `/opt/SMAW/bin/dtcpnfd` to start the daemon manually.

5.2.5 Trusted host configuration

If a node cannot communicate with the next node and if there are no other cluster nodes in the same subnet, the NIC failover module cannot determine if its own connection to the subnet has failed or if the remote NIC has failed.

To resolve this situation, each node should know at least one trusted host for each monitored subnet. The default gateway for each node, if present, is automatically considered to be a trusted host.

An optional list of trusted hosts can be provided in the `/etc/opt/SMAW/SMAWdtcp/dtcpnfd.hosts` file.

Each trusted host is listed on a separate line. Valid entries are resolvable host names or IP addresses in dotted decimal format.

Text following a hash (#) is ignored up to the end of the line.

The following is an example file:

```
# Example configuration file
# Specify one host address per line.
#
172.25.218.1
#router
router.mycompany.com
```

It is recommended to use hosts that are known to be available at all times; for example, company DNS servers, routers, and so forth. The administrator is responsible to provide adequate routing to the trusted nodes for each cluster node.



The NIC failover cannot be verified by shutting down the interface using the `ifconfig down` command.

6 Administration

This chapter describes the administration utilities included in SIS. These utilities are commands for such SIS administration tasks as starting, stopping, reconfiguring, and checking SIS, and for displaying debug messages.

This chapter discusses the following:

- The Section “Administering with Cluster Admin” details how to use the PRIMECLUSTER graphical user interface (GUI).
- The Section “Administering with the CLI” describes the `dtcpadmin` command.
- The Section “SIS daemon” describes the `dtcpd` command.
- The Section “Displaying the status of SIS” describes the `dtcpstat` command.
- The Section “Debug messages” describes the `dtcpdbg` program.

6.1 Administering with Cluster Admin

You can administer SIS using the Cluster Admin GUI by logging on to Cluster Admin (refer to the Section “Cluster Admin”) and clicking on the *SIS* tab at the bottom of the left panel (refer to the Section “Starting SIS”).

6.1.1 Using the GUI

After launching the SIS GUI, you will see the main administration window (see Figure 50). The main administration window consists of the following:

- SIS configuration tree on the left—Displays the configuration, the status of the nodes and services, and some basic statistics.
- Clusterwide summary table on the right—Shows a summary table of all of the configured interfaces, their type, IP addresses, netmasks, and the scheduled nodes configured for each service. The details of the services include the name, port number, scheduling and the nodes offering the service.

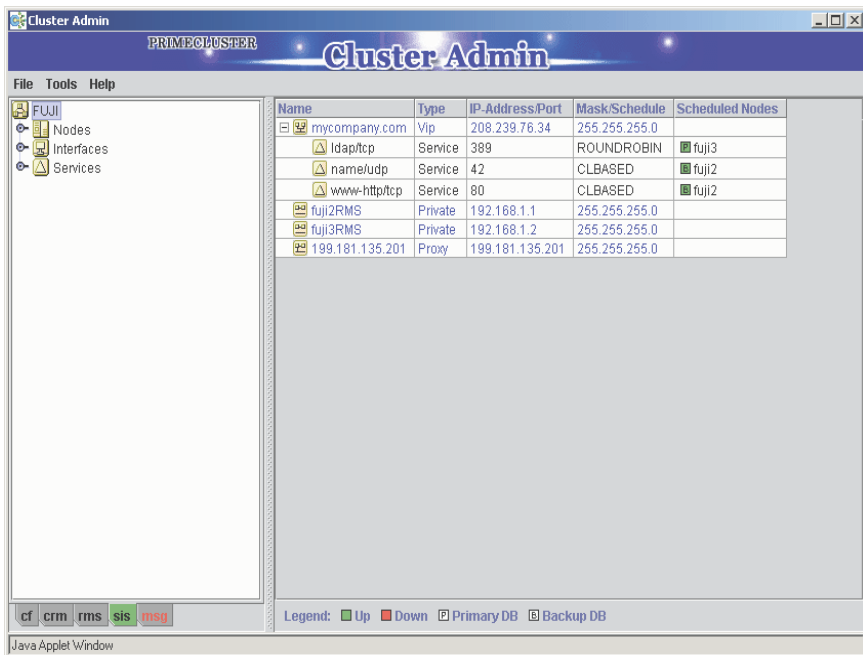


Figure 50: Main administration window

6.1.1.1 SIS configuration tree

The left panel shows a tree depicting the SIS configuration from the following views of the cluster:

- *Nodes*—Shows the node-interfaces-services hierarchy
- *Interfaces*—Shows the interface-nodes-services hierarchy
- *Services*—Shows the service-interfaces-nodes hierarchy

Nodes view

Click *Nodes* to see the node details configured for SIS (see Figure 51). The right panel shows the node-summary table with the following node details:

- *Node Name*
- *Type*
- *State*

The screenshot shows the Cluster Admin application window. The title bar includes 'Cluster Admin' and 'PRIMECLUSTER'. The main window has a menu bar with 'File', 'Tools', and 'Help'. On the left is a tree view showing a folder 'FUJI' containing 'Nodes', 'Interfaces', and 'Services'. The 'Nodes' folder is expanded. The main area on the right displays a 'Node Summary Table' with the following data:

Node Name	Type	State
sidney		UP
berlin	Service	UP
munich		UP
fuji3	Database, Gateway, Service	UP
fuji2	Backup database, Gateway, Service	UP
toronto		UP

At the bottom of the window, there is a status bar with several indicators: 'cf', 'rms&pcs', 'sis', and 'msg'. The text 'Java Applet Window' is visible at the very bottom.

Figure 51: Nodes view

Click the expansion icon to display the nodes (see Figure 52). Select a node to see the following node details in the right panel:

- *Node Name*
- *Node Type*
- *Node Status*
- List of services with their details

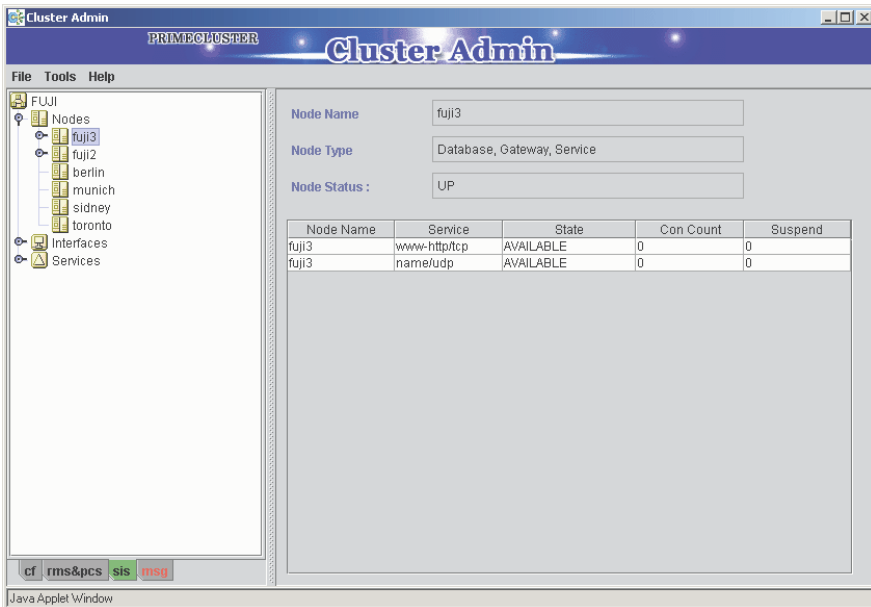


Figure 52: Node view showing node details

Click on a host name or IP address of a node interface to see the details of the interface. Click on an expansion icon in the left panel to see the services configured on that interface, including the node above. The right panel shows the following details (see Figure 53):

- *Interface Type*—VIP, PROXY, or PRIVATE
- *Address*—IP address of the interface
- *Mask*—The netmask of the interface
- *Preferred Gateway*—Only available if configured

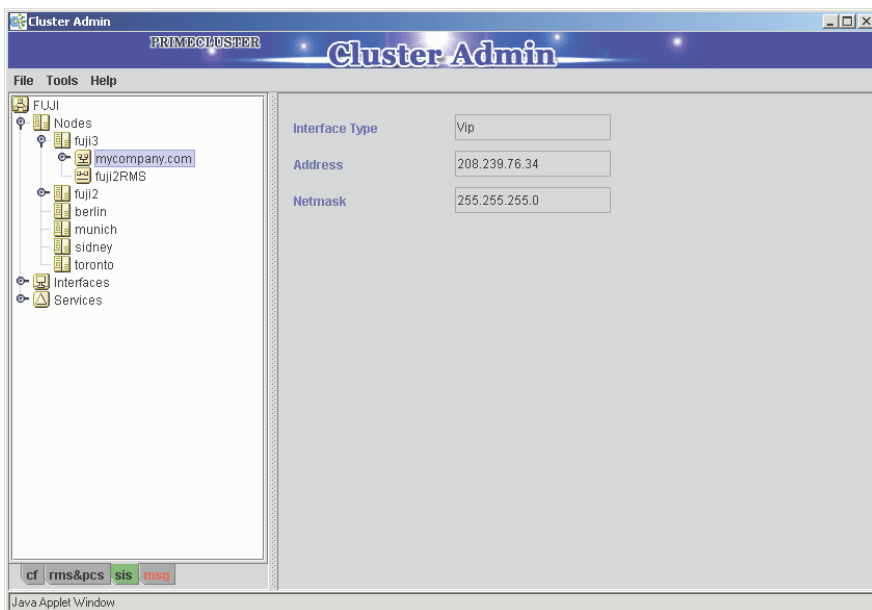


Figure 53: Node view showing interface details

Click on the service of a node interface in the left panel to see the details of the service. The right panel shows the following details (see Figure 54):

- *Service Name/Ports*—The name the service and the protocol, which is either tcp or udp.
- *Scheduling*—The scheduling algorithm for the service
- Details of the service

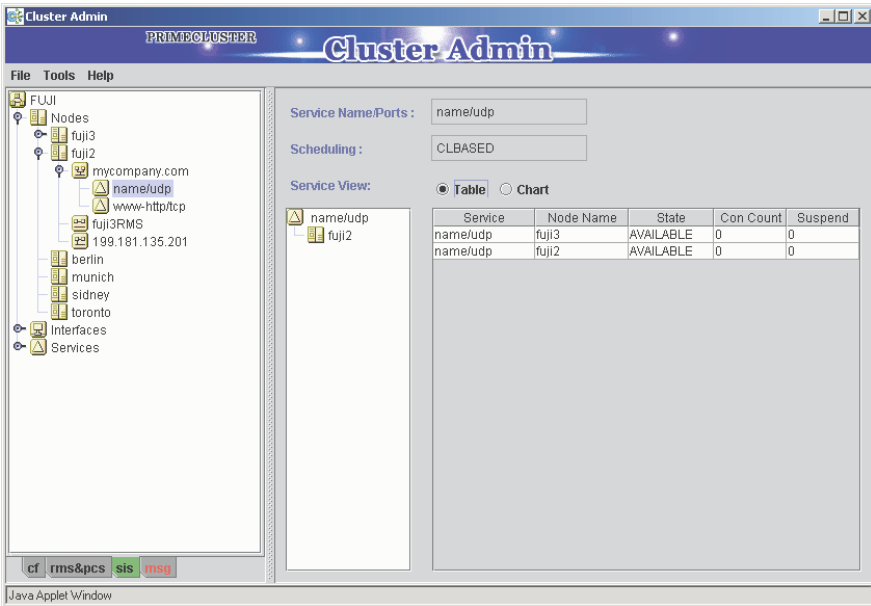


Figure 54: Node view showing interface details

Interfaces view

Click on the *Interfaces* branch on the SIS configuration tree on the left to see the configured interfaces (see Figure 55). This view shows a table of all of the interfaces with the following details:

- *Name*
- *Interface Type*
- *IP Address*
- *Mask (netmask)*

Name	Interface Type	IP Address	Netmask
fuji3RMS	Private	192.168.1.2	255.255.255.0
fuji2RMS	Private	192.168.1.1	255.255.255.0
199.181.135.201	Proxy	199.181.135.201	255.255.255.0
mycompany.com	Vip	208.239.76.34	255.255.255.0

Figure 55: Interfaces window

In the left panel, click on the expansion icon for to list the interfaces. Selecting an interface name or IP address reveals the details of that interface in the right panel:

- *Interface Type*—VIP, PROXY, or PRIVATE
- *Address*—IP address of the interface
- *Mask*—The netmask of the interface
- *Preferred Gateway*—Only available if configured

Further expanding of the tree reveals the nodes as well (see Figure 56).

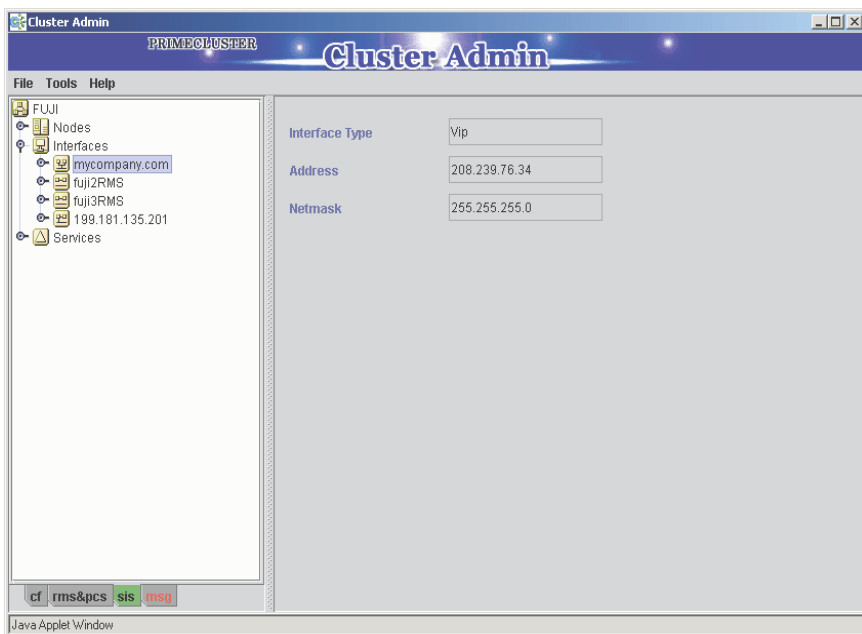


Figure 56: Interface window showing interface details

Further expanding the tree shows the services relative to each node (see Figure 57). Selecting a node shows the following node details in the right panel:

- *Node Name*
- *Node Type*
- *Node Status*
- List of services with their details

The screenshot shows the Cluster Admin interface. On the left is a tree view under 'FUJI' containing 'Nodes', 'Interfaces', and 'Services'. Under 'Nodes', there is a folder 'mycompany.com' containing nodes 'fuji3', 'fuji2', 'fuji2RMS', and 'fuji3RMS'. The 'fuji3' node is selected. The right panel displays details for 'fuji3':

- Node Name:** fuji3
- Node Type:** Database, Gateway, Service
- Node Status:** UP

Below the details is a table showing services for the selected node:

Node Name	Service	State	Con Count	Suspend
fuji3	www-http/tcp	AVAILABLE	0	0
fuji3	name/udp	AVAILABLE	0	0

The interface also shows a menu bar (File, Tools, Help) and a status bar at the bottom with 'Java Applet Window'.

Figure 57: Interface window with nodes and services

Click on a service to see its details (see Figure 58). The right panel shows the following:

- *Service Name/Ports*—The name the service and the protocol, which is either `tcp` or `udp`.
- *Scheduling*—The scheduling algorithm for the service
- Details of the service

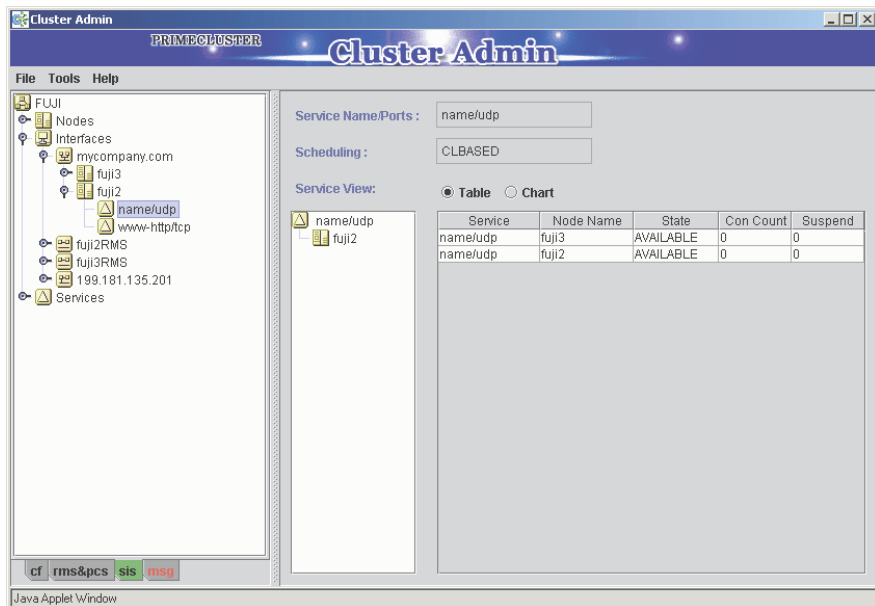
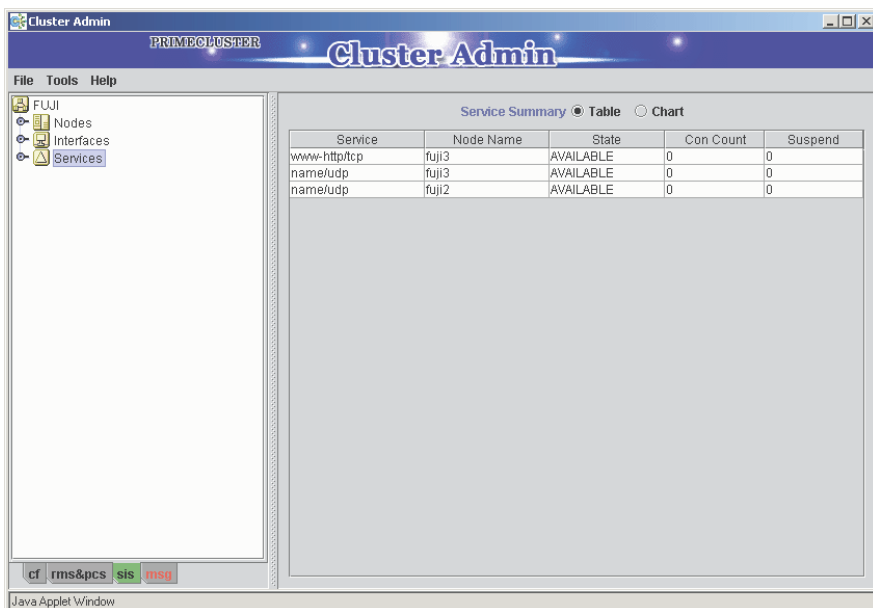


Figure 58: Interfaces window with service details

Services view

To view the services, click on *Services* in the SIS configuration tree in the left panel. This displays the services configured in the cluster. The right panel shows a summary table of all of the services as configured on the nodes with their status and statistics (see Figure 59).



The screenshot shows the Cluster Admin interface for a PRIMECLUSTER. The left sidebar contains a tree view with 'Services' selected. The main area displays a 'Service Summary' table with columns for Service, Node Name, State, Con Count, and Suspend. The table lists three services: www-http/tcp on fuji3, name/udp on fuji3, and name/udp on fuji2, all in an AVAILABLE state with 0 Con Count and 0 Suspend.

Service	Node Name	State	Con Count	Suspend
www-http/tcp	fuji3	AVAILABLE	0	0
name/udp	fuji3	AVAILABLE	0	0
name/udp	fuji2	AVAILABLE	0	0

Figure 59: Services window

Click on a service in the left panel to see its status in the right panel. The right panel shows the following service details:

- *Service*
- *Node Name*
- *State* (state of the service—Available, Suspended, or Unknown)
- *Con count* (connection count)
- *Suspend* (number of suspended states for each service)

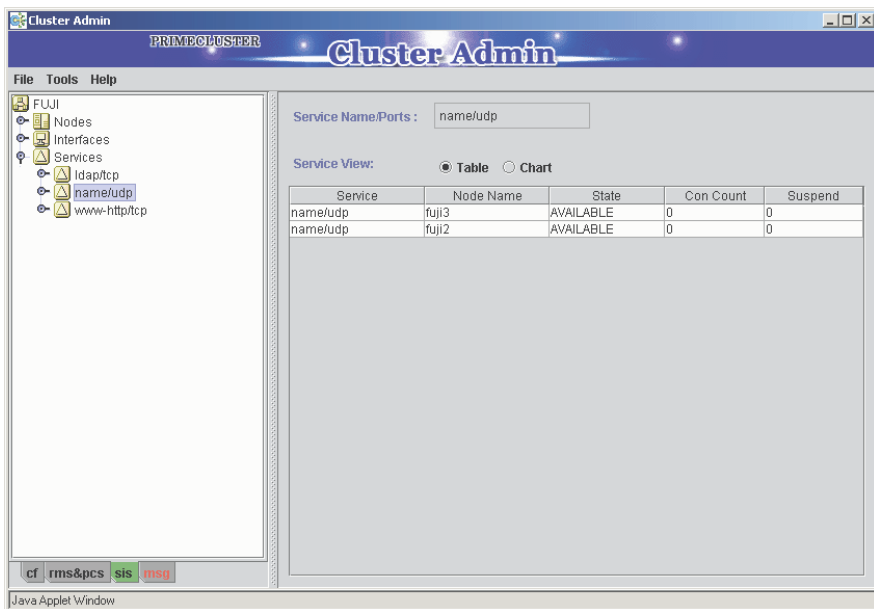


Figure 60: Service window showing details

Click on a host name or an interface IP address of a service in the left panel to see the interface details of the service. The interface details are shown in the right panel (see Figure 61). The interface details include the following:

- *Interface Type*—VIP, PROXY, or PRIVATE
- *Address*—IP address of the interface
- *Mask*—The netmask of the interface
- *Preferred Gateway*—Only available if configured

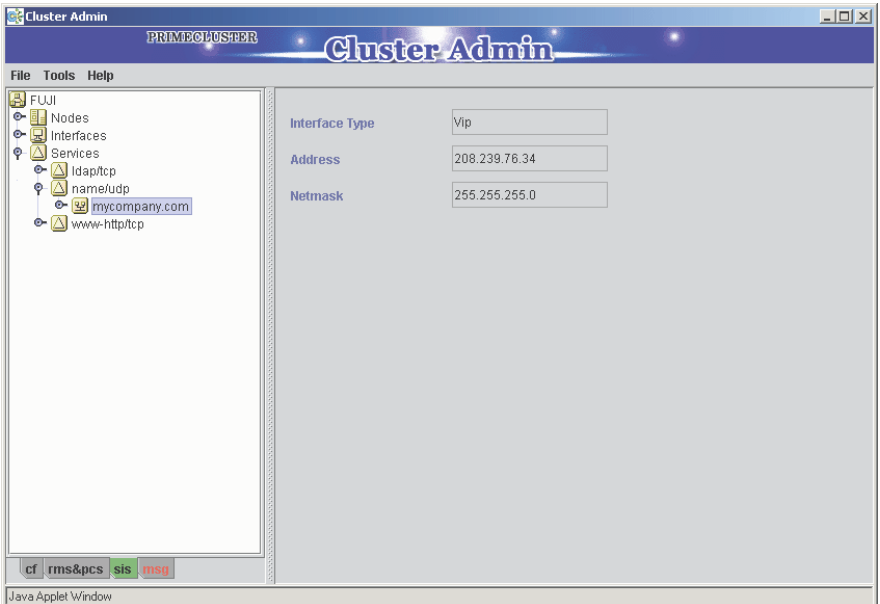


Figure 61: Service window showing interface details

Click on a node from the list on the left panel to see its details in the right panel (see Figure 62). The node details include the following:

- *Node Name*
- *Node Type*
- *Node Status*
- *Failover Nodes*
- List of services with their details, which are configured for that interface

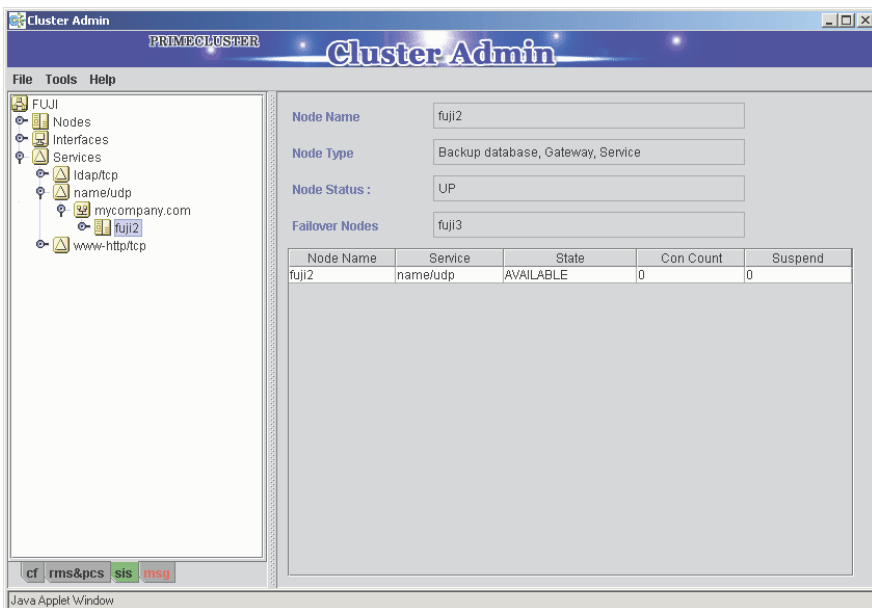


Figure 62: Service window showing node details

6.1.2 Using the menu bar

The menu bar of the SIS GUI offers some additional features as detailed in the following sections.

6.1.2.1 File

The *File* menu provides the *Exit* option.

Exit

Select *Exit* to leave Cluster Admin (see Figure 63).

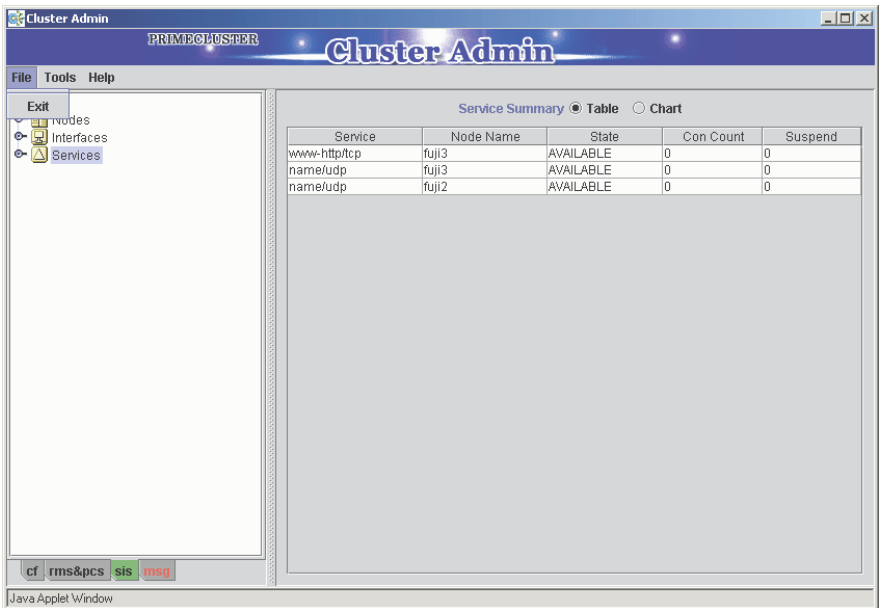


Figure 63: File menu

6.1.2.2 Tools

The *Tools* menu offers the following options (see Figure 64):

- *Start*
- *Stop*
- *Pause*
- *Resume*
- *Switch Gateway Node*
- *Add Backup Database Node*
- *Satellite node*
- *Wizard*

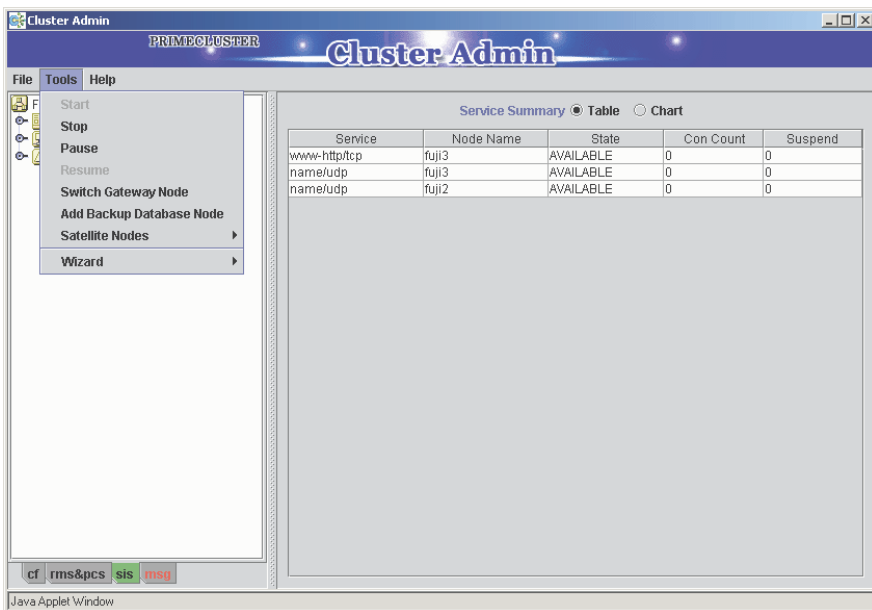


Figure 64: Tools menu

Stop, Start, Pause, and Resume options on all cluster nodes

If SIS is up and running, the *Start* and *Resume* options are not available. To stop SIS on every node in the cluster, select the *Stop* option from the menu, which brings up the confirmation dialog box (see Figure 65). Click on *Yes* to stop SIS. Once SIS is stopped in this manner, the SIS stopped window appears (see Figure 66). Click on *Ok*. The data displayed on the panels disappear.

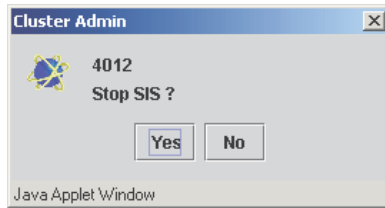


Figure 65: Stop SIS window

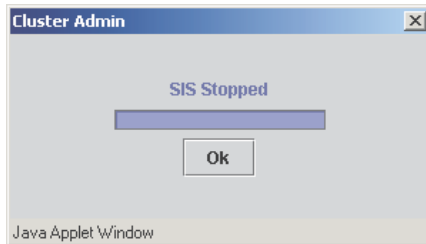


Figure 66: SIS stopped window

To start SIS on every node in the cluster, select the *Start* menu option from the *Tools* menu. This brings up the *SIS Startup Menu* window (see Figure 67). This window allows you to perform one of the following options:

- *Search all configuration files*—Searches all of the configuration files on all of the nodes of the SIS cluster.
- *Start configuration wizard*—Starts the configuration wizard.
- *Restore the last session*—Starts SIS with the previous configuration (the last configuration used before stopping SIS).

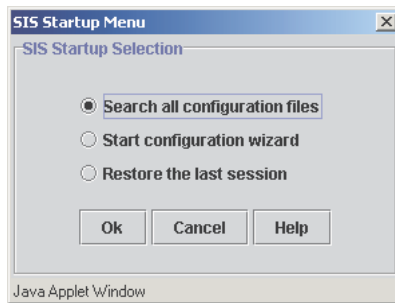


Figure 67: SIS Startup Menu window

If you select the *Search all configuration files* radio button and click *Ok*, it brings up the *SIS Clusterwide Startup* window (see Figure 68). This window shows the cluster tree in the left panel with the nodes and the configuration files on them (in the default directory `/etc/opt/SMaw/SMawdtcp`).

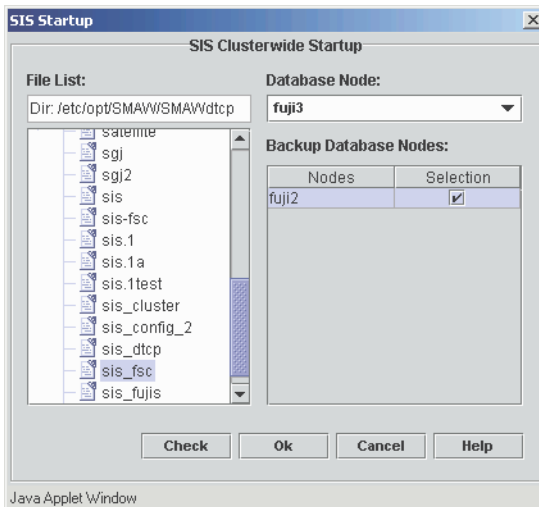


Figure 68: SIS Clusterwide Startup window

Use the right-mouse button on the items in the tree in the left panel to raise a small pop-up menu (see Figure 69). The menu items are as follows:

- *Reload*—Reloads all of the configuration files on the selected node.

- *View*—Opens a window that enables viewing the configuration file (see Figure 70). Activate this option by right-clicking on the configuration file name in the left-panel tree.
- *Edit*—Opens the configuration wizard for editing the selected configuration file. Activate this option by right-clicking on the configuration file name in the left-panel tree.

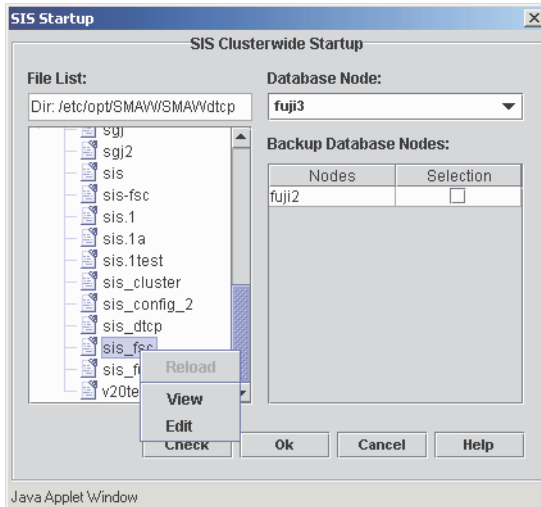


Figure 69: Startup options

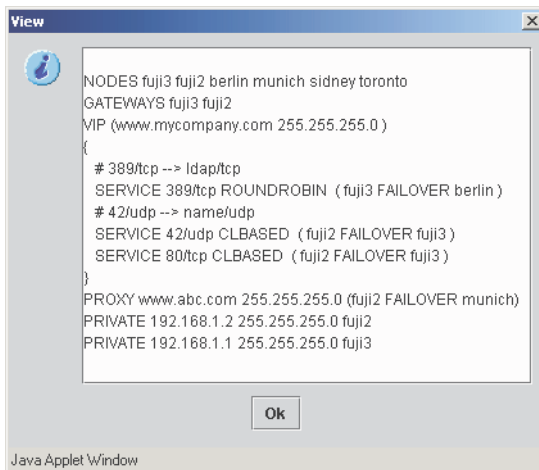


Figure 70: View configuration file window

Click on one of the nodes to select a predefined configuration file. You can also make selections for the primary and backup database nodes for the configuration on the right panel (see Figure 71).

After making all of the selections, the *Check* button becomes active. Click on the *Check* button to perform a syntax check on the selected configuration file.

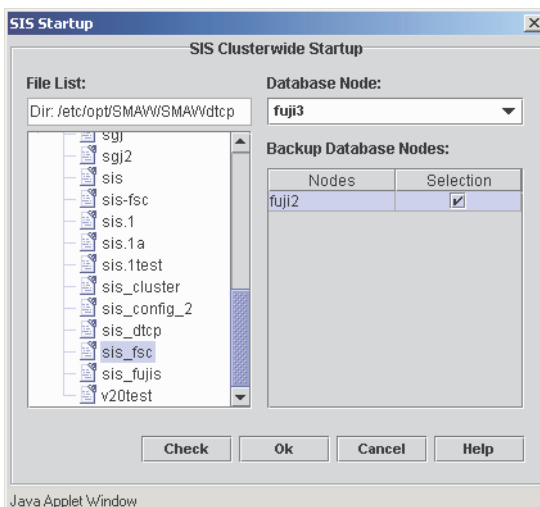


Figure 71: Selecting file and database node

The syntax check either returns with errors (see Figure 72), or it is successful (see Figure 73). If the syntax check is successful, then click on the *OK* button and SIS will start with the selected file. If the check was erroneous, select another file and repeat the check.



Figure 72: Syntax check error window

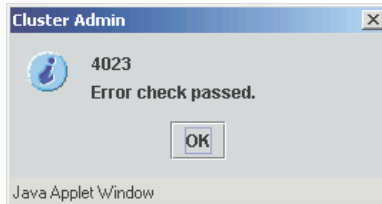


Figure 73: Syntax check OK window

To restore the configuration that was running before you stopped SIS, select the *Restore the last session* radio button on the *SIS Startup Menu* window and click *OK*. The GUI will automatically start SIS on all of the nodes with the previous configuration.

To start the wizard and create a new configuration file, select the *Start Wizard* radio button and click *OK*. This will start the SIS configuration wizard (as explained in the Section “Creating a new configuration file”).

For pausing SIS on every node in the cluster, select the *Pause* option from the *Tools* menu. This brings up a confirmation dialog box. Click *Yes* to pause (see Figure 74). After SIS has been paused, the *Pause* option becomes inactive and the *Resume* option becomes active.

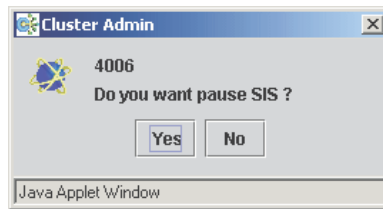


Figure 74: Pausing SIS

To resume SIS on every node in the cluster, select the *Resume* option from the *Tools* menu and click *Yes* on the confirmation dialog box (see Figure 75). This will inactivate the *Resume* option and activate the *Pause* option again.

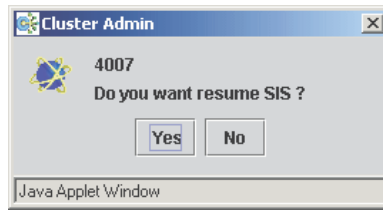


Figure 75: Resuming SIS

Starting, stopping operations on a single node

For operations on individual nodes, select the node from the SIS configuration tree on the left panel and click the right mouse button. This brings up a small menu list with the *Start* and *Stop* options (see Figure 76).

Select one of the active menu items to bring up a confirmation dialog box (see Figure 77), the same as clusterwide operation. Click *OK* to perform the operation. Errors are reported back. SIS indicates success by changing the status of the node in the right panel.

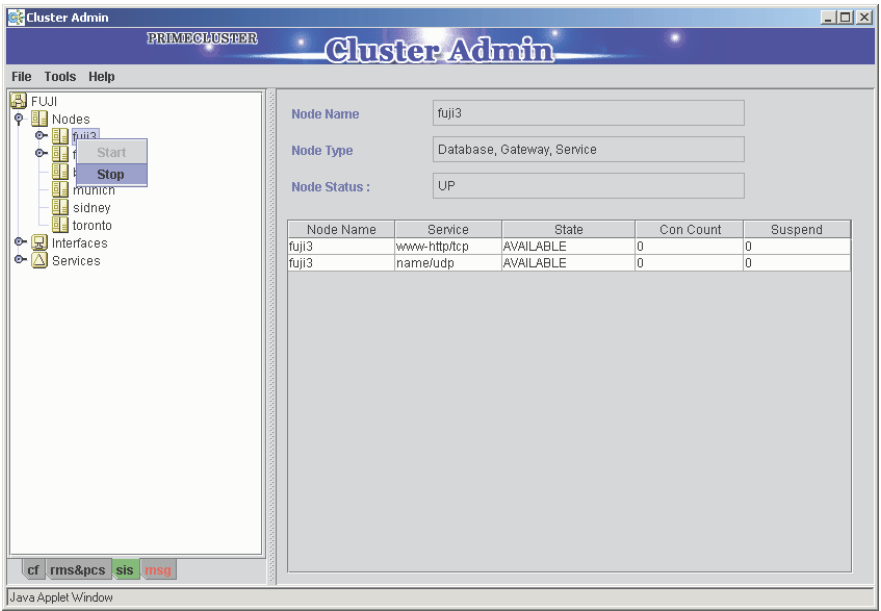


Figure 76: Start and stop on an individual node

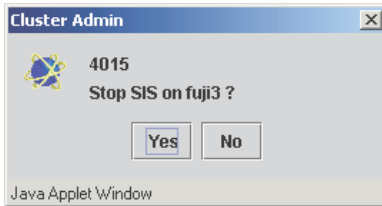


Figure 77: Stopping individual node

Switch gateway node

Choose the *Switch Gateway Node* option from the *Tools* menu to change a gateway node from one node to another. Selecting this option brings up the *Select Gateway Node* window (see Figure 78).

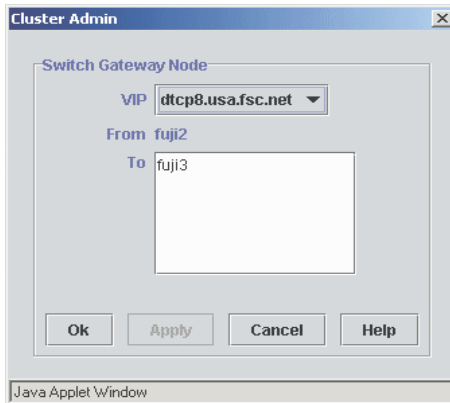


Figure 78: Select Gateway Node window

Select the gateway node to be switched to another node from the *VIP* pull-down list. Select the node that you want to switch. Click on *Apply*.

Add backup database node

Choose the *Add Backup Database Node* option to add a backup database node. This brings up the *Add Backup DBs* window (see Figure 79). Select a backup database from the *Candidates* window and click on *Apply*.

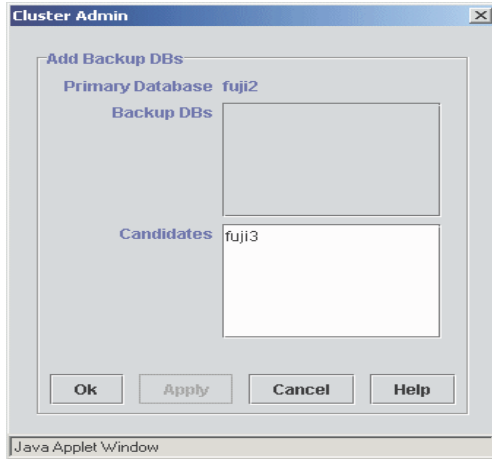


Figure 79: Add Backup DBs window

Satellite node

The Satellite node menu option offers the following choices:

- *Expel Satellite Nodes from the Cluster*—Removes satellite nodes from the SIS cluster.
- *Activate Satellite Nodes*—Joins expelled satellite nodes to the SIS cluster.

Select *Expel Satellite Nodes from the Cluster*. A window appears that lists the available satellite nodes in the cluster (see Figure 80).

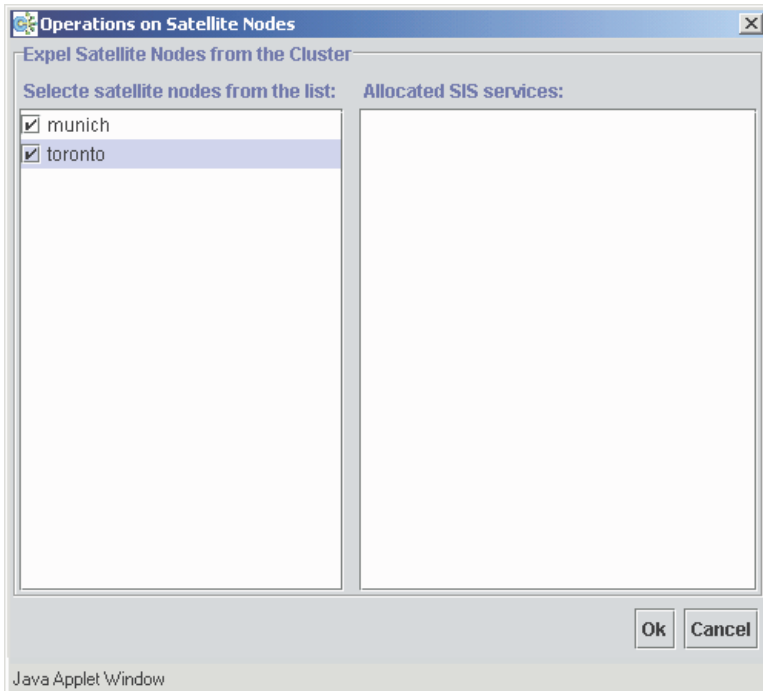


Figure 80: Expelling satellite nodes

Select the satellite nodes to be expelled from the left panel. The right panel displays all the services allocated on the selected node. Click on the *Ok* button to expel the selected satellite nodes.

Select *Activate Satellite Nodes*. A window appears that lists the expelled satellite nodes (see Figure 80).

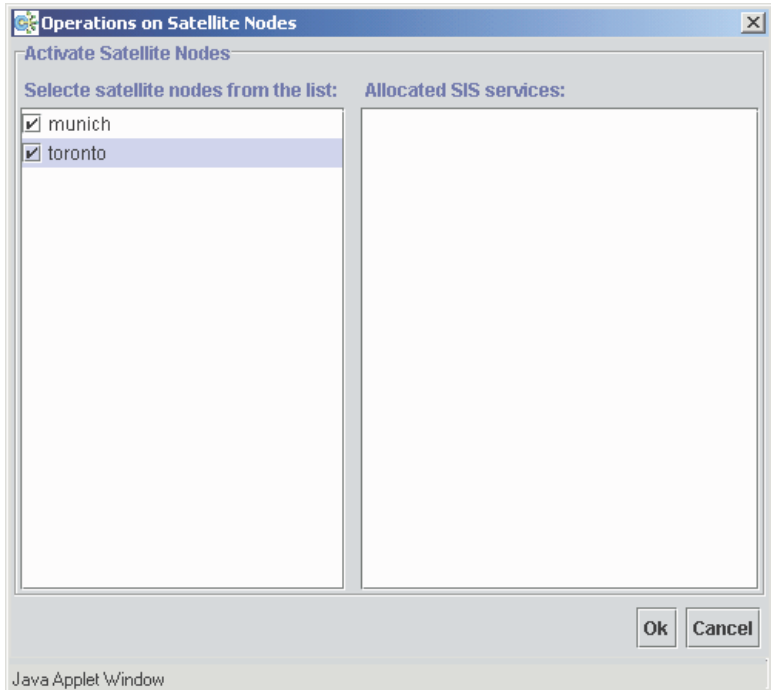


Figure 81: Expelling satellite nodes

Select the satellite nodes to be joined from the left panel. The right panel displays all the services allocated on the selected node. Click on the *Ok* button to activate the selected satellite nodes.

Wizard

The *Wizard* menu option offers the following choices:

- *New Configuration File*—Select to create a new file using the wizard (refer to the Section “Creating a new configuration file”).
- *Edit Configuration file*—Select to bring up the *Edit Wizard* window (see Figure 82). Find the file that you want to edit as follows:
 - a) Click on the node for the configuration file that you want to edit. This shows all of the SIS configuration files on that node.

- b) Select the file that you want to edit.
- c) The *View* and *Edit* buttons become active. To view the file, click on the *View* button (see Figure 70). Click the *Edit* button to open the configuration wizard for editing the selected configuration file using the wizard (refer to Section “Creating a new configuration file”).

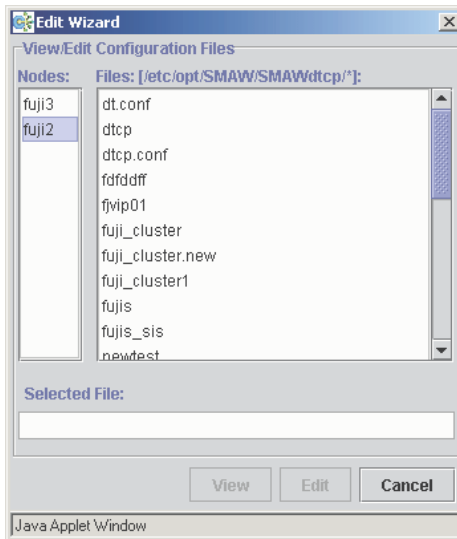


Figure 82: Edit Wizard window

6.1.2.3 Help

The *Help* menu option has help for all of the supported products in the GUI (see Figure 83). The menu items are as follows:

- *Content*
- *About*

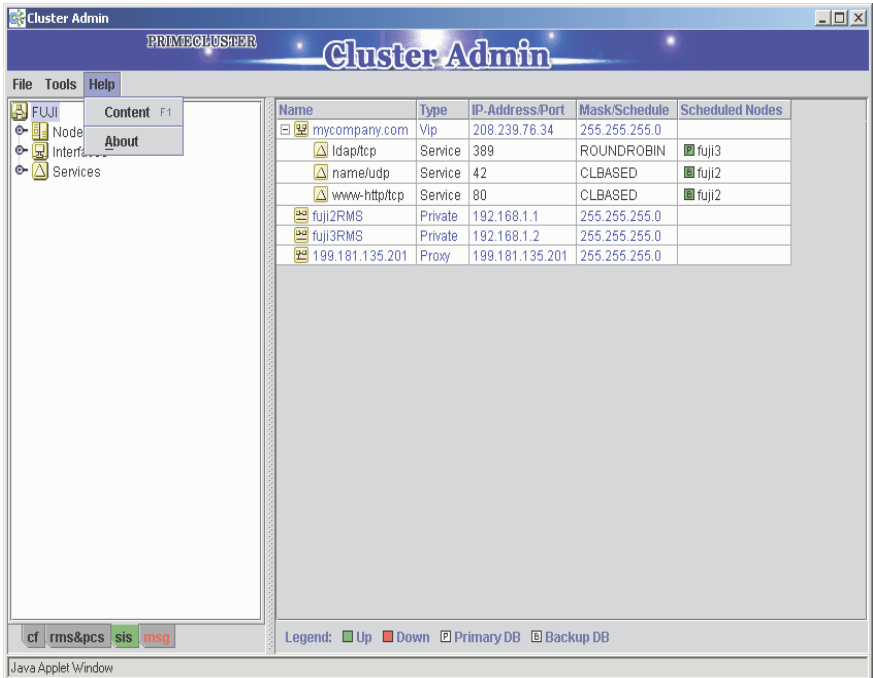


Figure 83: Help menu options

Selecting *Content* opens up another window (see Figure 84). This window lists all of the help topics per the products supported by the GUI and includes additional details of important procedures and menu items in the GUI.

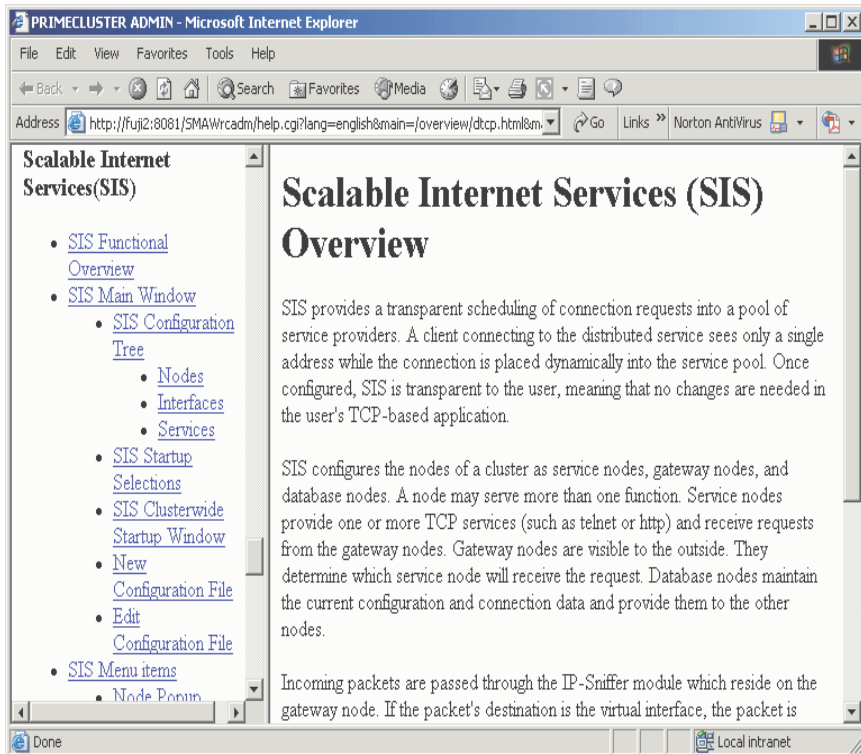


Figure 84: Help contents

Selecting *About* reveals the version details of all of the packages installed on the node, including Cluster Admin (see Figure 85).

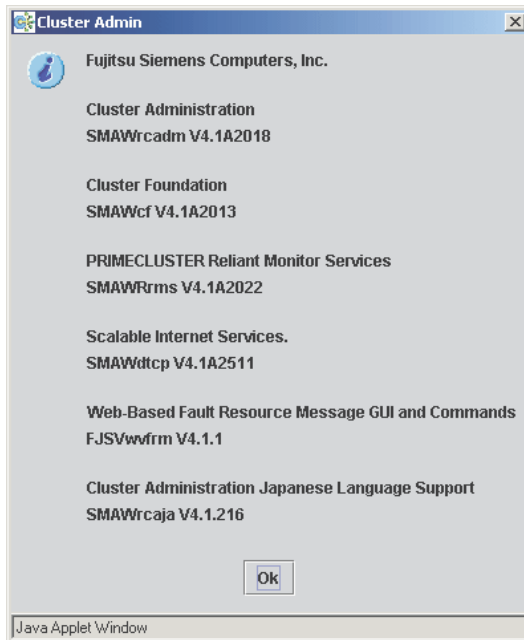


Figure 85: About option


6.2 Administering with the CLI

Use the `dtcpadmin` command to administer SIS from the command line (CLI).

The `dtcpadmin` command starts, stops, and reconfigures SIS as well as performing other administrative functions. The following list shows the `dtcpadmin` options and their meanings:

<code>dtcpadmin -s</code>	Starts SIS on the local node. Run this command on all nodes in the SIS cluster that are not database nodes.
<code>dtcpadmin -u</code>	Stops SIS on the local node. This command will fail on a database node if there are no backup database nodes and there are one or more non-database nodes still active. If the local node is inactive, then SIS will not report an error.
<code>dtcpadmin -r <i>config_file</i></code>	Reconfigures SIS. The new <i>config_file</i> is parsed and the configuration is sent to the designated database nodes. This command can be used on any active node.

<code>dtcpadmin -c <i>config_file</i></code>	<p>Makes the local node a database node. The node needs to be inactive. The first node called with this command reads the contents of the file, configures, and starts SIS on the local node. Additional nodes called with this command option need a valid <i>config_file</i> to prevent a syntax error; however, the configuration data are ignored and SIS is started on the local node as a backup database node.</p> <p>This command needs to be executed on at least one node.</p> <div style="border: 1px solid black; padding: 5px; margin-top: 10px; display: inline-block;"> <p>i To make SIS highly available, it is strongly recommended that you have at least one backup database node; however, at times it may be more convenient to use the <code>dtcpadmin -b</code> command since it does not require a syntactically correct configuration file.</p> </div>
<code>dtcpadmin -b <i>node</i></code>	<p>Sets the given active node as a backup database node. If no node name is given, it sets the local node as the backup database node. The node needs to be active. If the node is already a backup database node, no changes to the node are made.</p>
<code>dtcpadmin -g <i>gateway_node</i> <i>vip_address</i></code>	<p>Changes the gateway node for the VIP address <i>vip_address</i> to the gateway node <i>gateway_node</i>.</p>
<code>dtcpadmin -f</code>	<p>Pauses SIS on all nodes (freeze).</p> <p>When <code>dtcpadmin -f</code> is executed, all incoming packets are ignored. This can lead to dropped connections.</p>
<code>dtcpadmin -w</code>	<p>Resumes a paused SIS program (warm up).</p>
<code>dtcpadmin -n</code>	<p>Displays the SIS configuration of the local node.</p>
<code>dtcpadmin -p</code>	<p>Displays the current SIS configuration in a parsable format.</p>

<code>dtcpadmin -T <i>UDP_timeout</i></code>	<p>Sets the value of the UDP timeout. If entered without the <i>UDP_timeout</i>, the current value is printed.</p> <p>For UDP, SIS measures the interval between data packets from a client for the same service. Data packets that arrive within the timeout are considered to be part of the same UDP pseudo-connection. Data packets that arrive after this interval are treated as a new pseudo-connection.</p> <p>The default value is 5 seconds.</p>
<code>dtcpadmin -d <i>debug_level</i></code>	<p>Sets debug level (0 to 5). SIS displays the messages on the system console. Refer to the Section “SIS daemon” for redirection information. Debug levels are as follows:</p> <ul style="list-style-type: none"> ● 0 displays maximum debug information ● 1 through 4 display varying degrees of debug information ● 5 displays critical information (default) <p>This command can be used in combination with other options.</p> <p> Caution Do not use the <code>-d</code> option unless advised to do so by support personnel.</p>
<code>dtcpadmin -q</code>	When used in combination with other options, this command displays no output (quiet).
<code>dtcpadmin -t <i>config_file</i></code>	Tests <i>config_file</i> for syntax errors and displays configuration information.
<code>dtcpadmin -v</code>	Displays the version number of the <code>dtcpadmin</code> command.
<code>dtcpadmin -? -h</code>	Displays usage (help).

`dtcpadmin -k node1 node2 . . .` Expels the specified satellite node or nodes.

`dtcpadmin -S node1 node2 . . .` Readmits previously expelled satellite node or nodes.

`dtcpadmin -W -v <vip_address> -p <port> node1:weight1 node2:weight2 . . .` Changes the weights of nodes if the scheduling algorithm uses the weighted connection count algorithm for the *vip_address*. The node is specified in the same way as in the `NODES` section of the configuration. If a node does not exist in the configuration, it will be ignored and if it is not present in the list, their weights will be unchanged.

6.3 Displaying the status of SIS

Use the `dtcpstat` to display the status of SIS. The following list shows the usage for the `dtcpstat` command and its options:

<code>dtcpstat -? -h</code>	Shows a brief help message.
<code>dtcpstat -N</code>	Shows the status of each of the SIS nodes.
<code>dtcpstat -l</code>	Shows the status of all the configured services.
<code>dtcpstat -g [vip_address]</code>	Shows the gateway node associated with the VIP address <i>vip_address</i> . If no <i>vip_address</i> is specified, it shows the gateway nodes for all the configured VIP addresses.
<code>dtcpstat</code> (without any options)	Shows the status of all the connections.

The `-q` option can be used with any of the above options to suppress unnecessary headers.

The `-n` option can be used to show network addresses including host names and service names, such as IP addresses and port numbers respectively.

6.3.1 Status by node

The `dtcpstat -N` command displays the status of every node in the cluster configured for SIS. Entering the `dtcpstat -N` command results in output similar to the following:

Id	Name	Type	State
0	fuji1	DGS	UP
1	fuji2	LS	UP
2	fuji3	dGS	UP
3	fuji4	GS	DOWN

Type refers to the type of node. The meaning of the output for Type is as follows:

- D database node
- d Backup database node
- G Gateway node
- S Service node
- L Satellite node



State refers to the state of SIS on that node, not the state of the node.

6.3.2 Status by service

The `dtcpstat -l` command displays information about all available SIS services in the cluster. Specifying either of the options `node_name` or `service` limits the information to specific nodes or services. Only the services that are both configured and started are shown.

You can specify both `service` and `node_name`, but only one of each at a time. For instance, you can specify either `rlogin` or `telnet` for `service`, but you cannot specify them together. The following are examples of valid and invalid commands:

- Valid command:

```
dtcpstat -l telnet
```

- Invalid command:

```
dtcpstat -l telnet rlogin
```



Either a port number or a name can be entered for *service*.

The output contains the following information:

- Service
- Name of the node on which the service runs
- State of the service (Available, Suspended, or Unknown)
- Connection count (Con count)
- Number of Suspended states for each service

A service is put into the *Suspended* state if the maximum number of pending connections is reached (see description of the *backlog* parameter in the *listen(3N)* manual page).

Con count shows the number of currently established connections for the specified service.

Entering the `dtcpstat -l` command with any of the options results in output similar to the following:

```
$ dtcpstat -l
```

Service	Protocol	Node	Service State	Con count	Suspends
telnet	tcp	fuji1	Available	3	0
telnet	tcp	fuji2	Available	1	0
sunrpc	tcp	fuji1	Available	1	0
sunrpc	tcp	fuji2	Available	1	0
sunrpc	udp	fuji1	Available	0	0
sunrpc	udp	fuji2	Available	0	0
login	tcp	fuji1	Available	1	0
login	tcp	fuji2	Available	0	0

```
$ dtcpstat -l -q
```

telnet	tcp	fujil	Available	3	0
telnet	tcp	fujl2	Available	1	0
sunrpc	tcp	fujl1	Available	1	0
sunrpc	tcp	fujl2	Available	1	0
sunrpc	udp	fujl1	Available	0	0
sunrpc	udp	fujl2	Available	0	0
login	tcp	fujl1	Available	1	0
login	tcp	fujl2	Available	0	0

```
$ dtcpstat -l fujl1
```

Service	Protocol	Node	Service State	Con count	Suspends
telnet	tcp	fujl1	Available	3	0
sunrpc	tcp	fujl1	Available	1	0
sunrpc	udp	fujl1	Available	0	0
login	tcp	fujl1	Available	1	0

```
$ dtcpstat -l telnet
```

Service	Protocol	Node	Service State	Con count	Suspends
telnet	tcp	fujl1	Available	1	0
telnet	tcp	fujl2	Available	1	0

6.3.3 Status of SIS connections

Entering the `dtcpstat` command with no options displays information about all SIS connections in the cluster and results in output similar to the following:

```
$ dtcpstat
```

Node	Id	Local Address	Foreign Address	Gateways
fujl1	0	www.mycompany.com.login	client1.eng.pyra.965	fujl1

fujil	1	www.mycompany.com.telnet	client1.eng.pyra.1141	fujil
fujl2	0	www.mycompany.com.telnet	client2.eng.pyra.54108	fujil
fujl2	1	www.mycompany.com.login	client2.eng.pyra.1015	fujil

You can specify the following:

- One node or one service
- Both a node and a service

The service can be either a port number or a name. The output displays connections of the specified node or service. The `-n` option prints IP addresses in dot decimal notation.

Entering the `dtcpstat` command with options results in output similar to the following:

```
$ dtcpstat -n
```

Node	Id	Local Address	Foreign Address	Gateways
0	0	129.214.20.119.513	129.214.214.22.965	0
0	1	129.214.20.119.23	129.214.214.22.1141	0
1	0	129.214.20.119.23	129.214.214.28.54108	0
1	1	129.214.20.119.513	129.214.214.28.1015	0

```
$ dtcpstat fujil
```

Node	Id	Local Address	Foreign Address	Gateways
fujil	0	www.mycompany.com.login	client1.eng.pyra.965	fujil
fujil	1	www.mycompany.com.telnet	client1.eng.pyra.1141	fujil

```
$ dtcpstat 23
```

Node	Id	Local Address	Foreign Address	Gateways
fujil	1	www.mycompany.com.telnet	client1.eng.pyra.1141	fujil
fujl2	0	www.mycompany.com.telnet	client2.eng.pyra.54108	fujil

```
$ dtcpstat telnet
```

Node	Id	Local Address	Foreign Address	Gateways
fujil	1	www.mycompany.com.telnet	client1.eng.pyra.1141	fujil
fujl2	0	www.mycompany.com.telnet	client2.eng.pyra.54108	fujil

```
$ dtcpstat fuji1 telnet
```

Node	Id	Local Address	Foreign Address	Gateways
fuji1	1	www.mycompany.com.telnet	client1.eng.pyra.1141	fuji1

Domain names are truncated in the output.

6.3.4 Showing the gateway node

The `dtcpstat -g` command displays the gateway nodes for all the configured VIPs in the system. A VIP address can be specified after the `-g` option to show the gateway node only for that VIP address.

Some examples of the `dtcpstat -g` usage are as follows:

```
$dtcpstat -g
```

Vip	Gateway (Name)	Gateway (Id)
www.mycompany.com	fuji2	00
ldap.mycompany.com	fuji3	01

```
$dtcpstat -g www.mycompany.com
```

Vip	Gateway (Name)	Gateway (Id)
www.mycompany.com	fuji2	00

```
$dtcpstat -gn www.mycompany.com
```

Vip	Gateway (Name)	Gateway (Id)
192.168.17.1	fuji2	00

6.4 SIS daemon

The `dtcpd` command starts the SIS daemon. The daemon is responsible for receiving the VIP definitions from the configuration file and sending them to the TCP stack on the service nodes.

The command has the following syntax:

```
dtcpd [-d debug_file]
```

The `-d` option writes debug messages to the designated debug file instead of the system console.

6.5 Debug messages

The `dtcpdbg` program displays debugging information for SIS.

**Caution**

Do not use `dtcpdbg` unless advised to do so by support personnel. It slows down the system significantly.

See the Chapter “Debugging and troubleshooting” for more information on `dtcpdbg`.

7 Syntax rules

This chapter consists of the Backus Naur Form syntax rules to which the configuration file must conform.

```
[Config] ::= [NodeSection] [GatewaySection] [InterfaceSection] |
           [AssignmentSection] [NodeSection] [GatewaySection]
           [InterfaceSection]

[NodeSection] ::= [NodeDef] |
                 [NodeSection] [NodeDef]

[NodeDef] ::= NODES [List]

[AssignmentSection] ::= [Assignment] |
                       [AssignmentSection] [Assignment]

[Assignment] ::= STRING '=' [List] ';' |
                STRING '=' '(' [List] FAILOVER [List] ')' ';'

[CCNodeList] ::= [Blist] |
                 [CcNodeList] [Blist]

[ServNodeList] ::= [Alist] |
                  [Alist] [ServNodeList]

UdpServNodeList ::= [IPAddress] |
                   '(' [IPAddress] FAILOVER [List] ')'

[PrivateAlist] ::= [IPAddress]

[ProxyAlist] ::= [IPAddress] |
                '(' [IPAddress] FAILOVER [List] ')'

[Alist] ::= [IPAddress] |
           '(' [List] FAILOVER [List] ')'

[Bnum] ::= NUMBER

[Blist] ::= [Alist] |
           [Alist] ':' [Bnum]

[List] ::= [IPAddress] |
          [List] [IPAddress]

[InterfaceSection] ::= [InterfaceDef] |
                      [InterfaceSection] [InterfaceDef]

[InterfaceDef] ::= [VipDef] |
                  [ProxyDef] |
                  [PrivateDef]

[VipDef] ::= [VipStart] [VipList] '{' [VipBody] '}'

[VipStart] ::= VIP

[VipList] ::= [VipAddress] |
             [VipList] [VipAddress]

[FourBytes] ::= HEXNUMBER |
               DOTNUMBER
```

[IPAddress]	::=	[FourBytes] NUMBER STRING
[VipAddress]	::=	'(' [IPAddress] [IPAddress] ')' '(' [IPAddress] [IPAddress] [PGateway] ')'
[VipBody]	::=	[ServiceSection] [AssignmentSection] [ServiceSection]
[GatewaySection]	::=	[GatewayDef] [GatewaySection] [GatewayDef]
[GatewayDef]	::=	GATEWAYS [List]
[ServiceSection]	::=	[ServiceDef] [ServiceSection] [ServiceDef]
[ServiceDef]	::=	[SerStart] [PortList] [Schedule] [SerStart] [PortListDPO]
[SerStart]	::=	SERVICE
[PortList]	::=	[PortDef] [PortList] [PortDef]
[Pgateway]	::=	STRING
[PortDef]	::=	[PortValue] [PortValue] DCL
[PortListDPO]	::=	[PortDefDPO] [PortListDPO] [PortDefDPO]
[PortDefDPO]	::=	[PortValue] DPO [PortValue]
[PortValue]	::=	[PortRange] [PortVal] [PortRangeProto] [PortValProto]
[PortRange]	::=	[PortVal] ':' [PortVal]
[PortValProto]	::=	[PortVal] '/' [ProtoVal]
[ProtoVal]	::=	NUMBER STRING
[PortVal]	::=	NUMBER STRING
[Float]	::=	FLOAT
[Schedule]	::=	[KEEPLocal] [CLBASED] [ServNodeList] [SYSLoad] [ServNodeList] [ROUNDROBIN] [ServNodeList] [SPILOVER] [ServNodeList] AT [FLOAT] TO [ServNodeList] [CONCOUNT] [CcNodeList]
[ProxyDef]	::=	PROXY [IPAddress] [IPAddress] [ProxyAlist] PROXY [IPAddress] [IPAddress] [ProxyAlist] [Pgateway]

```
[PrivateDef] ::= PRIVATE [IPAddress] [IPAddress] [PrivateA1ist]
```

Non-terminal symbols are written in mixed case and are enclosed in brackets ([]). The following terminal symbols (upper case) are undefined: INTEGER, STRING, HEXNUMBER, and DOTNUMBER.

Uppercase strings *not* enclosed in brackets ([]) are keywords. In a configuration file, they must be entered literally.

Character literals are shown enclosed in single quotes (' '). In a configuration file, they are entered without quotes. If no whitespace is shown between a symbol and a literal, none is allowed.

If [PORT] is specified as a [STRING], it cannot take the value ftp-data, and if it is specified as a [NUMBER], it cannot take the value of 20.

8 Debugging and troubleshooting

This chapter contains information about the `dtcpdbg` command, which is used for debugging purposes. In addition, there is a section on troubleshooting to assist you with common configuration and administration issues.

This chapter discusses the following:

- The Section “`dtcpdbg`” describes the `dtcpdbg` command and provides an example of debug output.
- The Section “Troubleshooting” lists various debugging and troubleshooting solutions for SIS.

8.1 `dtcpdbg`



Caution

Do not use `dtcpdbg` unless directed to do so by support personnel. It slows down the system significantly.

The `dtcpdbg` command has the following syntax:

`dtcpdbg -?|-h` Displays usage

`dtcpdbg -G` Displays debugging output for all nodes

Enter the following commands after invoking `dtcpdbg`:

<code>help</code>	Print this list of commands.
<code>quit</code>	Quit <code>dtcpdbg</code> .
<code>stat</code>	Show current setting of <code>dtcpdbg</code> .
<code>global</code>	Start/stop global output (same as <code>-G</code> option).
<code>detail</code>	Start/stop displaying detailed output.
<code>tocon</code>	Stop/continue duplicating output to system console.
<code>Return</code>	Stop/continue debugging output to tty screen.
<code>redirect file_name</code>	Start/stop redirecting output to file.

<code>dup</code>	Duplicate output to file and screen.
<code>level [module_name all ALL]</code>	Change output level.
<code>debug_level</code>	

When you execute `dtcpdbg`, SIS displays the status (the same output as the `stat` command). SIS displays the output as follows:

node_ID: message_length DETAIL message

DETAIL is shown only if the `detail` flag has been set by the `detail` command. It has the following syntax:

debug_level (module_name, line_number)

The default *debug_level* is PANIC. The amount of output increases as the *debug_level* increases from PANIC to DEBUG. Debug levels are as follows:

- DEBUG
- TRACE
- LOG
- NOTICE
- WARNING
- PANIC

For example, entering the `dtcpdbg` command results in output similar to the following:

```
$ dtcpdbg
console flag: local tocon detail
(NU:DEBUG)(DB:DEBUG)(GW:DEBUG)(SV:DEBUG)(PS:DEBUG)(DM:DEBUG)(
VI:DEBUG)(DR:DEBUG)
0: (102) TRACE GW (gateway.c, 1202) dtcp_gw_process_arp: req
00:e0:b0:df:d5:
0: (102) TRACE S! (gw_sniffer.c, 365) dtcp_gw_send_arpmsg: to
00:e0:b0:dr_d5:
```

The amount of debugging information depends on the setting of *debug_level*. To set the level of output at DEBUG for module NU, perform the following steps:

1. After the system prompt, enter the following:

```
$ dtcpdbg
```

2. Enter:

```
level NU DEBUG
```



dtcpdbg has no prompt.

8.2 Troubleshooting

Use the following steps for debugging and troubleshooting SIS:

- Check if CF is functioning and verify that the nodes are up (refer to the CF installation guide for further information).

- Use the following command to check if dtcpd is running:

```
ps -ef | grep dtcpd
```

- Verify that the SIS configuration contains the correct nodes and services:

1. Use the `dtcpadmin -p` command to check the configuration file.

2. Enter the `dtcpadmin -r` command to reload the configuration.

- Use the following command to verify that the services scheduled are actually running:


```
ps | grep [httpd] <service>
```

- Use the following command to verify that the VIPs are configured and functioning:

```
netstat -ni.
```

The output for the `netstat -ni` command should look similar to the following:

```
$ netstat -in
Name      Mtu      Net/Dest  Address      [truncated]
lo0       8232     127.0.0.0  127.0.0.1
hme0     1500     192.168.21.0  192.168.21.14
hme1     1500     192.168.22.0  192.168.22.14
le0       1500     172.25.0.0   172.25.219.14
```


 The important information is the driver/interface name in the first column. Some of the lines in the above examples have been truncated.

- Check the system logs for any error messages:
 1. Use the `dmesg` command to display the most recent error messages.
 2. For additional error messages, look at the following file:

```
/var/adm/messages
```

- Check if the services are available to SIS:
 1. Use the following command:
 2. If the service is configured and appears in `dtcpadmin -p` but not in `dtcpstat -l <service>`, then try stopping (killing) and restarting the service.

Some services such as `telnet` and `ftp` are started by `inetd`. To restart these services, you need to restart the `inetd`.

 To kill the `inetd`, use a local console to log into the node.

9 Manual pages

This chapter lists the online manual pages for SIS.

To display a manual page, type the following command:

```
$ man man_page_name
```

1M. System administration

dtcpadmin(1M)
administer SIS

dtcpd(1M)
start the SIS daemon for configuring VIPs

dtcpdbg(1M)
display debug information about SIS

dtcpnfd(1M)
Netcard monitor daemon

dtcpstat(1M)
display status of connections within SIS

Index

A

- activating satellite nodes 111
- administering SIS
 - CLI 118
 - Cluster Admin 87
- assigning external connectivity 14
- Available 98, 123

B

- backup database nodes
 - See* database nodes
- blade servers 75

C

- checking syntax 107
- CLBASED 22
- client-based algorithm 13
- Cluster Admin 15
 - administering 87
 - configuring 28
 - main administration window 87
 - starting SIS 103
- Cluster Foundation 17
 - adding nodes 37
 - interconnect 18
 - with PRIVATE address 18
- cluster name
 - Linux satellite 82
 - Solaris satellite 82
 - Windows 2000 satellite 76
- Clusterwide Operations menu 103
- clusterwide summary table 87
- CONCOUNT 23
- configuration file
 - creating 113
 - editing 113
 - example 70
 - GATEWAYS 20
 - NODES 20
 - reloading 104
 - syntax rules 129

- variables 19
- viewing 105
- configuration wizard 105
- configuring
 - Cluster Admin 28
 - SIS Wizard 34
 - text editor 18
- connection count, weighted 23
- creating a configuration file 113

D

- daemon, SIS 126
- database nodes 67
 - backup 9, 17, 102
 - primary 9, 66
 - reconfigure 66
 - satellites 73
- DCL 22, 25, 55
- FTP 26
- GUI button 48
- DEBUG 134
- Debug menu 101
- debug messages, displaying 127
- debug_level 134
- debugging 135
- displaying
 - debug messages 127, 133
 - interfaces 90
 - SIS 121
 - status by node 122
 - status by service 122
- documentation 2
- dtcpadmin 118
 - command 118
 - options 120
- dtcpd 126
- dtcpdbg 127, 133
- dtcpstat 121, 126

E

- editing the configuration file 113

Index

expelling satellite nodes 111

F

FAILOVER 25

failover

defining scheduling 49

NIC 83

nodes 14, 56

PROXY 27

proxy addresses 14, 26

File menu 101

FTP 26, 55

G

gateway nodes

concepts 17

description 8

NIC failover 83

satellites 73

TCP 22

GATEWAYS 20

H

Help menu 115

I

interconnect 18

interface definitions 21

PRIVATE 27

VIP 21

VIP body 21

interfacedef 21

interfaces

details 99

node view 91, 92

Interfaces view 93

IP addresses 21

K

keep local algorithm 13

KEEPLOCAL 22

L

level 134

Linux NIC bonding 83

LOG 134

M

main administration window 87

manual pages list 137

menus

Clusterwide Operations 103

Debug 101

File 101

Help 115

Tools 102

Wizard 113

N

netmask 21

Network Interface Card 83

gateway node 83

service node 83

NIC

See Network Interface Card

NIC failover 83

module 83, 84

monitoring 84

restarting 85

starting 85

node_name 122

NODES 20

nodes

database 73

details 90, 95, 100

gateway 73

satellite 9, 73

view 89

Windows satellite 76

NOTICE 134

O

operations on individual nodes 108

P

PANIC 134

pausing SIS 107

prefgateway 21

- primary database nodes
 - See* database nodes
- PRIVATE 27
- private addresses 14, 27
- PROXY 26
- proxy addresses 14, 26
- R**
- reconfiguring SIS 118
- redirect 133
- reloading configuration files 104
- restoring the configuration 107
- resuming SIS 108
- round robin algorithm 13
- ROUNDROBIN 23
- S**
- satellite nodes 73
 - activating 111
 - differences from core nodes 9
 - expelling 111
 - hardware 74
 - menu options 111
 - overview 9
 - setting up 76
 - software 74
 - Windows 76
- scalability
 - network services 7
 - SIS 7
- scheduling 22
- scheduling algorithms
 - CLBASED 22
 - CONCOUNT 23
 - KEEPLOCAL 22
 - ROUNDROBIN 23
 - SPILLOVER 23
 - SYSLOAD 22
- service 122
- service nodes
 - client-based 22
 - concepts 17
 - description 8
 - NIC 83
 - VIPs 21
- Services view 97
- SIS
 - daemon 126
 - network services 7
 - starting 118
 - status 121
 - stopping 118
- SIS configuration tree 87, 88
- SIS Wizard
 - completed 64
 - node selection screen 41
- Solaris IP multipathing 83
- specifying
 - Linux satellite cluster name 82
 - Solaris satellite cluster name 82
 - Windows satellite cluster name 76
- spill over algorithm 13
- SPILLOVER 23
- starting
 - individual nodes 108
 - NIC failover module 85
 - SIS daemon 126
- starting SIS
 - CLI 118
 - Cluster Admin 103
 - clusterwide 103
 - individual node 108
- state of services
 - Available 98, 123
 - Suspended 98, 123
 - Unknown 98, 123
- status
 - by node 122
 - by service 122
 - displaying 121
- stopping individual nodes 108
- stopping SIS
 - CLI 118
 - clusterwide 103
 - individual node 108
 - Suspended 98, 123

Index

Synfinity Link/GLS 83
syntax
 check 107
 rules 129
syntax rules 129
SYSLOAD 22
system load algorithm 13

T

TCP 19, 24
 defining 46
 example 70
 VIP body 22
tocon 133
Tools menu 102
TRACE 134
troubleshooting 135

U

UDP 19, 22, 24
 example 70
 selecting 46
Unknown 98, 123
using the menu bar 101

V

variables 19
viewing the configuration file 105
Virtual Interface Providers
 body 21
 gateway node 83
 interface definition 21

W

WARNING 134
weighted connection count 13, 23
Windows
 cluster name 76
 satellite nodes 76
Windows 2000 9, 74
Wizard menu 113

Fujitsu Siemens Computers GmbH
User Documentation
33094 Paderborn
Germany

Comments
Suggestions
Corrections

Fax: (++49) 700 / 372 00001

email: manuals@fujitsu-siemens.com
<http://manuals.mchp.siemens.de>

Submitted by

Comments on PRIMECLUSTER™
Scalable Internet Services (SIS)



Fujitsu Siemens Computers GmbH
User Documentation
33094 Paderborn
Germany

Comments
Suggestions
Corrections

Fax: (++49) 700 / 372 00001

email: manuals@fujitsu-siemens.com
<http://manuals.mchp.siemens.de>

Submitted by

Comments on PRIMECLUSTER™
Scalable Internet Services (SIS)



