Sun Cluster 3.0 U1 Data Services Installation and Configuration Guide

Sun Microsystems, Inc.
901 San Antonio Road
Palo Alto, CA 94303-4900 U.S.A.
650-960-1300

Part No. 806-7071-10
July 2001, Revision A

Contents

Preface

The *Sun™ Cluster 3.0 U1 Data Services Installation and Configuration Guide* contains procedures to install and configure the Sun Cluster data services.

This document is intended for system administrators with extensive knowledge of Sun software and hardware. Do not use this document as a planning or presales guide. Before reading this document, you should have already determined your system requirements and purchased the appropriate equipment and software.

The instructions in this document assume knowledge of the Solaris™ operating environment and expertise with the volume manager software used with Sun Cluster.

UNIX Commands

This document contains information on commands specific to installing and configuring Sun Cluster data services. It might not contain information on basic UNIX® commands and procedures, such as shutting down the system, booting the system, and configuring devices. For that information, see one or more of the following:

- AnswerBook2™ online documentation for the Solaris software environment
- Solaris operating environment man pages
- Other software documentation that you received with your system

Typographic Conventions

Typeface or Symbol	Meaning	Examples
`AaBbCc123`	The names of commands, files, and directories; on-screen computer output	Edit your file. Use `ls -a` to list all files. `% You have mail`.

`AaBbCc123`	What you type, when contrasted with on-screen computer output	`%` **`su`**
		`Password:`

AaBbCc123	Book titles, new words or terms, words to be emphasized	Read Chapter 6 in the *User's Guide*. These are called *class* options. You *must* be superuser to do this.
	Command-line variable; replace with a real name or value	To delete a file, type `rm` *filename*.

Shell Prompts

Shell	Prompt
C shell	*machine_name*%
C shell superuser	*machine_name*#
Bourne shell and Korn shell	`$`
Bourne shell and Korn shell superuser	#

Related Documentation

Application	Title	Part Number
Installation	*Sun Cluster 3.0 U1 Installation Guide*	806-7069
Hardware	*Sun Cluster 3.0 U1 Hardware Guide*	806-7070
API development	*Sun Cluster 3.0 U1 Data Services Developers' Guide*	806-7399
Administration	*Sun Cluster 3.0 U1 System Administration Guide*	806-7073
Cluster concepts	*Sun Cluster 3.0 U1 Concepts*	806-7074
Release Notes	*Sun Cluster 3.0 U1 Release Notes*	806-7078

Sun Documentation Online

The `docs.sun.com`SM Web site enables you to access Sun technical documentation on the Web. You can browse the `docs.sun.com` archive or search for a specific book title or subject at `http://docs.sun.com`.

Help

If you have problems installing or using Sun Cluster, contact your service provider and provide the following information:

- Your name and E-mail address (if available)
- Your company name, address, and phone number
- The model and serial numbers of your systems
- The release number of the operating environment (for example, Solaris 7)
- The release number of Sun Cluster (for example, Sun Cluster 3.0)

Use the following commands to gather information about each node on your system for your service provider.

Command	Function
`prtconf -v`	Displays the size of the system memory and reports information about peripheral devices.
`psrinfo -v`	Displays information about processors.
`showrev –p`	Reports which patches are installed.
`prtdiag -v`	Displays system diagnostic information.
`scinstall -pv`	Displays Sun Cluster release and package version information.

Also have available the contents of the `/var/adm/messages` file.

CHAPTER 1

Planning for Sun Cluster Data Services

This chapter provides planning information and guidelines for installing and configuring Sun Cluster data services. This chapter contains the following sections.

See the *Sun Cluster 3.0 U1 Concepts* document for conceptual information about data services, resource types, resources, and resource groups.

If your applications are not currently offered as Sun Cluster data services, see the *Sun Cluster 3.0 U1 Data Services Developers' Guide* for information on how to develop other applications to become highly available data services.

Sun Cluster Data-Services Installation and Configuration Tasks

The following table lists the chapters that describe the installation and configuration of Sun Cluster data services.

TABLE 1-1 Task Map: Installing and Configuring Sun Cluster Data Services

Task	For Instructions, Go To ...
Install and configure the Sun Cluster HA for Oracle data service	Chapter 2
Install and configure the Sun Cluster HA for iPlanet™ Web Server data service	Chapter 3
Install and configure the Sun Cluster HA for Netscape LDAP data service	Chapter 4
Install and configure the Sun Cluster HA for Apache data service	Chapter 5
Install and configure the Sun Cluster HA for DNS data service	Chapter 6
Install and configure the Sun Cluster HA for NFS data service	Chapter 7
Install and configure the Sun Cluster HA for Oracle Parallel Server data service	Chapter 8
Install and configure the Sun Cluster HA for SAP data service	Chapter 9
Install and configure the Sun Cluster HA for Sybase ASE data service	Chapter 10
Administer data-service resources	Chapter 11

Configuration Guidelines for Sun Cluster Data Services

This section provides configuration guidelines for Sun Cluster data services.

Determining the Location of the Application Binaries

You can install the application software and application configuration files on one of the following locations.

- **The local disks of each cluster node –** The advantage to placing the software and configuration files on the individual cluster nodes is that if you want to upgrade the application software later, you can do so without shutting down the service. The disadvantage is that you then have several copies of the software and configuration files to maintain and administer.
- **The cluster file system –** If you put the application binaries on the cluster file system, you have only one copy to maintain and manage, but you must shut down the data service in the entire cluster to upgrade the application software. If you can spare a small amount of downtime for upgrades, put a single copy of the application and configuration files on the cluster file system.

  See the planning chapter of the *Sun Cluster 3.0 U1 Installation Guide* for information on creating cluster file systems.

Verifying the `nsswitch.conf` File Contents

The `nsswitch.conf` file is the configuration file for name-service lookups. This file determines the following information.

- which databases within the Solaris environment to use for name-service lookups
- in what order to consult the databases

Some data services require that you direct "group" lookups to "files" first. For these data services, change the "group" line in the `nsswitch.conf` file so that the "files" entry is listed first. See the chapter for the data service you are configuring to determine whether you need to change the "group" line.

See the planning chapter in the *Sun Cluster 3.0 U1 Installation Guide* for additional information on how to configure the `nsswitch.conf` file for the Sun Cluster environment.

Planning the Cluster File System Configuration

Depending on the data service, you might need to configure the cluster file system to meet Sun Cluster requirements. See the chapter for the data service that you plan to configure to determine whether any special considerations apply.

See the planning chapter of the *Sun Cluster 3.0 U1 Installation Guide* for information on creating cluster file systems.

Relationship Between Resource Groups and Disk Device Groups

Sun Cluster uses the concept of *node lists* for disk device groups and resource groups. Node lists are ordered lists of primary nodes, which are potential masters of the disk device group or resource group. Sun Cluster uses a *failback policy* to determine what happens when a node has been down and then rejoins the cluster, and the rejoining node appears earlier in the node list than the current primary node. If failback is set to `True`, the device group or resource group will be switched off of the current primary and switched onto the rejoining node, making the rejoining node the new primary.

To ensure high availability of a failover resource group, make the resource group's node list match the node list of associated disk device groups. For a scalable resource group, the resource group's node list cannot always match the device group's node list because, currently, a device group's node list must contain exactly two nodes. For a greater-than-two-node cluster, the node list for the scalable resource group can have more than two nodes.

For example, assume you have a disk device group `disk-group-1` that has nodes `phys-schost-1` and `phys-schost-2` in its node list, and the failback policy is set to `Enabled`. Assume you also have a failover resource group, `resource-group-1`, which uses `disk-group-1` to hold its application data. When you set up `resource-group-1`, also specify `phys-schost-1` and `phys-schost-2` for the resource group's node list and set the failback policy to `True`.

To ensure high availability of a scalable resource group, make the scalable resource group's node list a superset of the node list for the disk device group. Doing so ensures that the nodes that are directly connected to the disks are also nodes that can run the scalable resource group. The advantage is that, when at least one cluster node connected to the data is up, the scalable resource group runs on that same node, making the scalable services available also.

See the *Sun Cluster 3.0 U1 Installation Guide* for information on how to set up disk device groups. See the *Sun Cluster 3.0 U1 Concepts* document for more details on the relationship between disk device groups and resource groups.

SUNW.HAStorage Resource Type

The resource type SUNW.HAStorage serves the following purposes.

- coordinates the boot order of disk devices and resource groups by causing the START methods of the other resources in the same resource group containing the SUNW.HAStorage resource to wait until the disk device resources become available
- with AffinityOn set to True, enforces colocation of resource groups and disk device groups on the same node, thus enhancing the performance of disk-intensive data services

---

Note – If the device group is switched to another node while the SUNW.HAstorage resource is online, AffinityOn has no effect and the resource group does *not* migrate along with the device group. On the other hand, if the resource group is switched to another node, AffinityOn being set to True causes the device group to follow the resource group to the new node.

Recommendations

To determine whether to create SUNW.HAStorage resources within a data service resource group, consider the following criteria.

- In cases where a data-service resource group has a node list in which some of the nodes are not directly connected to the storage, you must configure SUNW.HAStorage resources in the resource group and set the dependency of the other data-service resources to the SUNW.HAStorage resource. This requirement coordinates the boot order between the storage and the data services.
- If your data service is disk intensive, such as the Sun Cluster HA for Oracle and Sun Cluster HA for NFS data services, add a SUNW.HAStorage resource to your data-service resource group, set the dependency of your data-service resources to the SUNW.HAStorage resource, and set AffinityOn to True. When you perform these steps, the resource groups and disk device groups are colocated on the same node.
- If your data service is *not* disk intensive—such as one that reads all its files at startup (for example, the Sun Cluster HA for DNS data service)—configuring the SUNW.HAStorage resource type is optional.
- If your cluster contains only two nodes, configuring the SUNW.HAStorage resource type is optional. However, if you plan to add nodes and run scalable services later on, you must configure the SUNW.HAStorage resource type when you perform these tasks. To prepare, you can set up the SUNW.HAStorage resource type now and add nodes to the node list later.

See the individual chapters on data services in this document for specific recommendations.

See "How to Set Up SUNW.HAStorage Resource Type for New Resources" on page 239 for information about the relationship between disk device groups and resource groups. Additional details are in the SUNW.HAStorage(5) man page.

Node List Properties

You can specify three node lists when configuring data services.

1. installed_nodes – A property of the resource type. This property is a list of the cluster node names on which the resource type is installed and enabled to run.

2. nodelist – A property of a resource group that specifies a list of cluster node names where the group can be brought online, in order of preference. These nodes are known as the potential primaries or masters of the resource group. For failover services, configure only one resource-group node list. For scalable services, configure two resource groups and thus two node lists. One resource group and its node list identifies the nodes on which the shared addresses are hosted. This list is a failover resource group on which the scalable resources depend. The other resource group and its list identifies nodes on which the application resources are hosted. The application resources depend on the shared addresses. Therefore, the node list for the resource group that contains the shared addresses must be a superset of the node list for the application resources.

3. auxnodelist – A property of a shared-address resource. This property is a list of physical node IDs that identify cluster nodes that can host the shared address but never serve as primary in the case of failover. These nodes are mutually exclusive with the nodes identified in the node list of the resource group. This list pertains to scalable services only. See the scrgadm(1M) man page for details.

Overview of the Installation and Configuration Process

Use the following procedures to install and configure data services.

- Install the data-service packages from the Sun Cluster Agents CD.
- Install and configure the application to run in the cluster environment.

- Configure the resources and resource groups that the data service uses. When you configure a data service, specify the resource types, resources, and resource groups that the Resource Group Manager (RGM) will manage. The chapters for the individual data services describe these procedures.

Before you install and configure data services, see the *Sun Cluster 3.0 U1 Installation Guide*, which includes procedures on how to install the data service software packages and how to configure Network Adapter Failover (NAFO) groups that the network resources use.

Note – You can use SunPlex Manager to install and configure the following data services: Sun Cluster HA for Oracle, Sun Cluster HA for iPlanet Web Server, Sun Cluster HA for Netscape Directory Server, Sun Cluster HA for Apache, Sun Cluster HA for DNS, and Sun Cluster HA for NFS. See the SunPlex Manager online help for more information.

Installation and Configuration Task Flow

The following table shows a task map of the procedures to install and configure a Sun Cluster failover data service.

TABLE 1-2 Task Map: Sun Cluster Data Service Installation and Configuration

Task	**For Instructions, Go to**
Install the Solaris and Sun Cluster software	*Sun Cluster 3.0 U1 Installation Guide*
Set up NAFO groups	*Sun Cluster 3.0 U1 Installation Guide*
Set up multihost disks	*Sun Cluster 3.0 U1 Installation Guide*
Plan resources and resource groups	*Sun Cluster 3.0 U1 Release Notes*
Decide the location for application binaries, and configure the `nsswitch.conf` file	Chapter 1
Install and configure the application software	The chapter for each data service in this book
Install the data-service software packages	*Sun Cluster 3.0 U1 Installation Guide* or the chapter for each data service in this book
Register and configure the data service	The chapter for each data service in this book

Example

The example in this section shows how you might set up the resource types, resources, and resource groups for an Oracle application that has been instrumented to be a highly available failover data service.

The main difference between this example and an example of a scalable data service is that, in addition to the failover resource group that contains the network resources, a scalable data service requires a separate resource group (called a scalable resource group) for the application resources.

The Oracle application has two components, a server and a listener. The Sun Cluster HA for Oracle data service is supplied by Sun, and therefore these components have already been mapped into Sun Cluster resource types. Both of these resource types are associated with resources and resource groups.

Because this example is a failover data service, the example uses logical-hostname network resources, which are the IP addresses that fail over from a primary node to a secondary node. Place the logical-hostname resources into a failover resource group, and then place the Oracle server resources and listener resources into the same resource group. This ordering enables all of the resources to fail over as a group.

To have the Sun Cluster HA for Oracle data service run on the cluster, you must define the following objects.

- `LogicalHostname` resource type – This resource type is built in, and therefore you need not explicitly register the resource type.
- Oracle resource types – The Sun Cluster HA for Oracle data service defines two Oracle resource types: a database server and a listener.
- Logical-hostname resources – These resources host the IP addresses that fail over in a node failure.
- Oracle resources – You must specify two resource instances for the Sun Cluster HA for Oracle data service: a server and a listener.
- Failover resource group – This container is composed of the Oracle server and listener and logical-hostname resources that will fail over as a group.

Tools for Data-Service Resource Administration

This section describes the tools you can use to perform installation and configuration tasks.

The SunPlex Manager Graphical User Interface (GUI)

SunPlex Manager is a web-based tool that enables you to perform the following tasks.

- Install a cluster.
- Administer a cluster.
- Create and configure resources and resource groups.
- Configure data services with the Sun Cluster software.

See the *Sun Cluster 3.0 U1 Installation Guide* for instructions on how to use SunPlex Manager to install cluster software. SunPlex Manager provides online help for most administrative tasks.

The Sun Cluster Module for the Sun Management Center GUI

The Sun Cluster module enables you to monitor clusters and to create and delete resources and resource groups from the Sun Management Center GUI. See the *Sun Cluster 3.0 U1 Installation Guide* for information about installation requirements and procedures for the Sun Cluster module. Go to `http://docs.sun.com` to access the Sun Management Center software documentation set, which provides additional information about Sun Management Center.

The `scsetup` Utility

The `scsetup`(1M) utility is a menu-driven interface that you can use for general Sun Cluster administration. You can also use this utility to configure data-service resources and resource groups. Select option 2 from the `scsetup` main menu to launch the Resource Group Manager submenu.

The `scrgadm` Command

You can use the `scrgadm` command to register and configure data-service resources. See the procedure on how to register and configure your data service in the applicable chapter of this book. If, for example, you use the Sun Cluster HA for Oracle data service, see "How to Register and Configure Sun Cluster HA for Oracle" on page 29. Chapter 11 also contains information on how to use the `scrgadm` command to administer data-service resources. Finally, see the `scrgadm`(1M) man page for additional information.

Data-Service Resource Administration Tasks

The following table lists which tool you can use in addition to the command line for different data-service resource administration tasks. See Chapter 11 for more information about these tasks and for details on how to use the command line to complete related procedures.

TABLE 1-3 Tools You Can Use for Data-Service Resource Administration Tasks

Task	SunPlex Manager	Sun Management Center	The `scsetup` Utility
Register a resource type	Yes	No	Yes
Create a resource group	Yes	Yes	Yes
Add a resource to a resource group	Yes	Yes	Yes
Bring a resource group online	Yes	Yes	No
Remove a resource group	Yes	Yes	No
Remove a resource	Yes	Yes	No
Switch the current primary of a resource group	Yes	No	No
Disable a resource	Yes	Yes	No
Move the resource group of a disabled resource into the unmanaged state	Yes	No	No
Display resource-type, resource-group, and resource configuration information	Yes	Yes	No
Change resource properties	Yes	No	No
Clear the `STOP_FAILED` error flag on resources	Yes	No	No
Add a node to a resource group	Yes	No	No

Sun Cluster Data-Service Fault Monitors

This section provides general information about data-service fault monitors. The Sun-supplied data services contain fault monitors that are built into the package. The fault monitor (or fault probe) is a process that probes the health of the data service.

Fault Monitor Invocation

The RGM invokes the fault monitor when you bring a resource group and its resources online. This invocation causes the RGM to internally call the `MONITOR_START` method for the data service.

The fault monitor performs the following two functions.

- monitors the abnormal exit of the data-service server process or processes
- checks the health of the data service

Monitoring of the Abnormal Exit of the Server Process

The Process Monitor Facility (PMF) monitors the data service processes.

The data service fault probe runs in an infinite loop and sleeps for an adjustable amount of time that the resource property `Thorough_probe_interval` sets. While sleeping, the probe checks with the PMF to see if the process has exited. If the process has exited, the probe updates the status of the data service as "Service daemon not running" and takes action. The action can involve restarting the data service locally or failing over the data service to a secondary cluster node. To decide whether to restart or to fail over the data service, the probe checks the value set in the resource properties `Retry_count` and `Retry_interval` for the data-service application resource.

Health Checks of the Data Service

Typically, communication between the probe and the data service occurs through a dedicated command or a successful connection to the specified data-service port.

The logic that the probe uses is roughly as follows.

1. Sleep (`Thorough_probe_interval`).

2. Perform health checks under a time-out property `Probe_timeout`. `Probe_timeout` is a resource extension property of each data service that you can set.

3. If Step 2 is a success, that is, the service is healthy, update the success/failure history. To update the success/failure history, purge any history records that are older than the value set for the resource property `Retry_interval`. The probe sets the status message for the resource as "Service is online" and returns to Step 1.

   If Step 2 resulted in a failure, the probe updates the failure history. The probe then computes the total number of times the health check failed.

   The result of the health check can range from a complete failure to success. The interpretation of the result depends on the specific data service. Consider a scenario where the probe can successfully connect to the server and send a handshake message to the server but receives only a partial response before timing out. This scenario is most likely a result of system overload. If some action is taken (such as restarting the service), the clients reconnect to the service again, thus further overloading the system. If this event occurs, a data service fault monitor can decide not to treat this "partial" failure as fatal. Instead, the monitor can track this failure as a nonfatal probe of the service. These partial failures are still accumulated over the interval that the `Retry_interval` property specifies.

   However, if the probe cannot connect to the server at all, the failure can be considered fatal. Partial failures lead to incrementing the failure count by a fractional amount. Every time the failure count reaches total failure (either by a fatal failure or by accumulation of partial failures), the probe either restarts or fails over the data service, attempting to correct the situation.

4. If the result of the computation in Step 3 (the number of failures in the history interval) is less than the value of the resource property `Retry_count`, the probe attempts to correct the situation locally (for example, by restarting the service). The probe sets the status message of the resource as "Service is degraded" and returns to Step 1.

5. If the number of failures in `Retry_interval` exceeds `Retry_count`, the probe calls `scha_control` with the "giveover" option. This option requests failover of the service. If this request succeeds, the fault probe stops on this node. The probe sets the status message for the resource as, "Service has failed."

6. The Sun Cluster framework can deny the `scha_control` request issued in the previous step for various reasons. The return code of `scha_control` identifies the reason. The probe checks the return code. If the `scha_control` is denied, the probe resets the failure/success history and starts afresh. This probe resets the history because the number of failures is already above `Retry_count`, and the fault probe would attempt to issue `scha_control` in each subsequent iteration (which would be denied again). This request would place additional load on the system and would increase the likelihood of further service failures if an overloaded system triggered service failures.

The probe then returns to Step 1.

CHAPTER **2**

Installing and Configuring Sun Cluster HA for Oracle

This chapter contains the following procedures.

- "How to Prepare the Nodes" on page 17
- "How to Install the Oracle Software" on page 19
- "How to Verify the Oracle Installation" on page 20
- "How to Configure Oracle Database Access With Solstice DiskSuite" on page 21
- "How to Configure Oracle Database Access With VERITAS Volume Manager" on page 21
- "How to Create an Oracle Database" on page 22
- "How to Set Up Oracle Database Permissions" on page 23
- "How to Install Sun Cluster HA for Oracle Packages" on page 28
- "How to Register and Configure Sun Cluster HA for Oracle" on page 29
- "How to Configure `SUNW.HAStorage` Resource Type" on page 33
- "How to Verify the Sun Cluster HA for Oracle Installation" on page 34

You must configure the Sun Cluster HA for Oracle data service as a failover service. See Chapter 1 and the *Sun Cluster 3.0 U1 Concepts* document for general information about data services, resource groups, resources, and other related topics.

Note – You can use SunPlex Manager to install and configure this data service. See the SunPlex Manager online help for details.

Installing and Configuring Sun Cluster HA for Oracle

The following table lists sections that describe the installation and configuration tasks.

TABLE 2-1 Task Map: Installing and Configuring HA for Oracle

Task	For Instructions, Go To
Prepare to install the Sun Cluster HA for Oracle data service	"Preparing to Install Sun Cluster HA for Oracle" on page 16
Install the Oracle application software	"Installing the Oracle Server Software" on page 17
Create an Oracle database	"Creating an Oracle Database" on page 20
Set up Oracle database permissions	"Setting Up Oracle Database Permissions" on page 23
Install the Sun Cluster HA for Oracle packages	"Installing Sun Cluster HA for Oracle Packages" on page 27
Register resource types and configure resource groups and resources	"Registering and Configuring Sun Cluster HA for Oracle" on page 28
Verify the Sun Cluster HA for Oracle installation	"Verifying the Sun Cluster HA for Oracle Installation" on page 34
Configure extension properties	"Configuring Sun Cluster HA for Oracle Extension Properties" on page 35
View fault-monitor information	"Sun Cluster HA for Oracle Fault Monitor" on page 37

Preparing to Install Sun Cluster HA for Oracle

Before you install the Sun Cluster HA for Oracle data service, select an install location for the following files.

- **Oracle application files –** These files include Oracle binaries, configuration files, and parameter files. You can install these files on either the local file system or on the cluster file system.

  See "Determining the Location of the Application Binaries" on page 3 for the advantages and disadvantages of placing the Oracle binaries on the local file system as opposed to the cluster file system.

- **Database-related files –** These files include the control file, redo logs, and data files. You must install these files on the cluster file system as either raw devices or regular files.

Installing the Oracle Server Software

Use the procedures in this section to complete the following tasks.

- Prepare the Sun Cluster nodes.
- Install the Oracle application software.
- Verify the Oracle installation.

Note – Before you configure the Sun Cluster HA for Oracle data service, follow the procedures in the *Sun Cluster 3.0 U1 Installation Guide* to configure the Sun Cluster software on each node.

▼ How to Prepare the Nodes

This procedure describes how to prepare the cluster nodes for installation of the Oracle application software.

Caution – Perform all the steps described in this section on all Sun Cluster nodes. If you do not perform all steps on all nodes, the Oracle installation will be incomplete, and the Sun Cluster HA for Oracle data service will fail during startup.

Note – Consult the Oracle documentation before you perform this procedure.

The following steps prepare Sun Cluster nodes and install the Oracle software.

1. **Become superuser on all the cluster members.**

2. **Configure the `/etc/nsswitch.conf` files as follows so that the data service starts and stops correctly if a switchover or failover occurs.**

   On each node that can master the logical host that runs the Sun Cluster HA for Oracle data service, include one of the following entries for `group` in the `/etc/nsswitch.conf` file.

   ```
   group:
   group:  files
   group:  files [NOTFOUND=return] nis
   group:  files [NOTFOUND=return] nisplus
   ```

   The Sun Cluster HA for Oracle data service uses the `su` *user* command to start and stop the database node. The network information name service might become unavailable when a cluster node's public network fails. Adding one of the preceding entries for group ensures that the `su`(1M) command does not refer to the NIS/NIS+ name services if the network information name service is unavailable.

3. **Configure the cluster file system for the Sun Cluster HA for Oracle data service.**

   If raw devices contain the databases, configure the global devices for raw-device access. See the *Sun Cluster 3.0 U1 Installation Guide* for information on how to configure global devices.

   When you use the Solstice™ DiskSuite volume manager, configure the Oracle software to use UNIX file system (UFS) logging or raw-mirrored meta devices. See the Solstice DiskSuite documentation for more information on how to configure raw-mirrored meta devices.

4. **Prepare the `$ORACLE_HOME` directory on a local or multihost disk.**

   Note – If you install the Oracle binaries on a local disk, use a separate disk if possible. Installing the Oracle binaries on a separate disk prevents the binaries from overwrites during operating environment reinstallation.

   ---

5. **On each node, create an entry for the database administrator group (DBA) in the `/etc/group` file, and add potential users to the group.**

   You typically name the DBA group `dba`. Verify that the `root` and *oracle_id* users are members of the `dba` group, and add entries as necessary for other DBA users. Ensure that the group IDs are the same on all the nodes that run the Sun Cluster HA for Oracle data service, as the following example illustrates.

   ```
   dba:*:520:root,oracle
   ```

   You can create group entries in a network name service (for example, NIS or NIS+). If you do so, add your entries to the local `/etc/inet/hosts` file to eliminate dependency on the network name service.

6. **On each node, create an entry for the Oracle user ID (*oracle_id*).**

   You typically name the Oracle user ID `oracle`. The following command updates the `/etc/passwd` and `/etc/shadow` files with an entry for the Oracle user ID.

   ```
   # useradd -u 120 -g dba -d /Oracle-home oracle
   ```

   Ensure that the *oracle_id* user entry is the same on all the nodes that run the Sun Cluster HA for Oracle data service.

▼ How to Install the Oracle Software

Perform the following steps to install the Oracle software.

1. **Become superuser on a cluster member.**

2. **Note the Oracle installation requirements.**

   Install Oracle binaries on one of the following locations.

   - Local disks of the cluster nodes
   - Cluster file system

   Note – Before you install the Oracle software on the cluster file system, start the Sun Cluster software and become the owner of the disk device group.

   See “Preparing to Install Sun Cluster HA for Oracle” on page 16 for more information about installation locations.

3. **Install the Oracle software.**

   Regardless of where you install the Oracle software, modify each node's `/etc/system` files as you would in standard Oracle installation procedures. Reboot afterward.

   Log in as *oracle_id* to ensure ownership of the entire directory before you perform this step. See the appropriate Oracle installation and configuration guides for instructions on how to install Oracle software.

▼ How to Verify the Oracle Installation

Perform the following steps to verify the Oracle installation.

1. **Verify that the *oracle_id* user and the `dba` group own the `$ORACLE_HOME/bin/oracle` directory.**

2. **Verify that the `$ORACLE_HOME/bin/oracle` permissions are set as follows.**

```
-rwsr-s--x
```

3. **Verify that the listener binaries exist in the `$ORACLE_HOME/bin` directory.**

Where to Go From Here

When you have completed the work in this section, go to "Creating an Oracle Database" on page 20.

Creating an Oracle Database

Complete the procedures in this section to configure and create the initial Oracle database in a Sun Cluster configuration. When creating and configuring additional databases, omit the procedure "How to Create an Oracle Database" on page 22.

▼ How to Configure Oracle Database Access With Solstice DiskSuite

If you use the Solstice DiskSuite volume manager, perform the following steps to configure Oracle database access with the Solstice DiskSuite volume manager.

1. **Configure the disk devices for the Solstice DiskSuite software to use.**

   See the *Sun Cluster 3.0 U1 Installation Guide* for information on how to configure the Solstice DiskSuite software.

2. **If you use raw devices to contain the databases, run the following commands to change each raw-mirrored metadevice's owner, group, and mode.**

   If you do not use raw devices, do not perform this step.

   a. **If you create raw devices, run the following commands for each device on each node that can master the Oracle resource group.**

```
# chown oracle_id /dev/md/metaset/rdsk/dn
# chgrp dba_id /dev/md/metaset/rdsk/dn
# chmod 600 /dev/md/metaset/rdsk/dn
```

metaset — Specifies the name of the diskset.

/rdsk/d*n* — Specifies the name of the raw disk device within the *metaset* diskset.

   b. **Verify that the changes are effective.**

```
# ls -lL /dev/md/metaset/rdsk/dn
```

▼ How to Configure Oracle Database Access With VERITAS Volume Manager

If you use VxVM software, perform the following steps to configure Oracle database access with the VxVM software.

1. **Configure the disk devices for the VxVM software to use.**

   See the *Sun Cluster 3.0 U1 Installation Guide* for information on how to configure VERITAS Volume Manager.

2. **If you use raw devices to contain the databases, run the following commands on the current disk-group primary to change each device's owner, group, and mode.**

   If you do not use raw devices, do not perform this step.

   a. **If you create raw devices, run the following command for each raw device.**

```
# vxedit -g diskgroup set user=oracle_id group=dba mode=600 volume
```

diskgroup Specifies the name of the disk group.

volume Specifies the name of the raw volume within the disk group.

   b. **Verify that the changes are effective.**

```
# ls -lL /dev/vx/rdsk/diskgroup/volume
```

   c. **Reregister the disk device group with the cluster to keep the VxVM namespace consistent throughout the cluster.**

```
# scconf -c -D name=diskgroup
```

▼ How to Create an Oracle Database

1. **Prepare database configuration files.**

   Place all database-related files (data files, redolog files, and control files) on either shared raw global devices or on the cluster file system. See "Preparing to Install Sun Cluster HA for Oracle" on page 16 for information on install locations.

   Within the `init$ORACLE_SID.ora` or `config$ORACLE_SID.ora` file, you might need to modify the assignments for `control_files` and `background_dump_dest` to specify the locations of the control files and alert files.

Note – If you use Solaris authentication for database logins, set the `remote_os_authent` variable in the `init$ORACLE_SID.ora` file to `True`.

---

2. **Create the database.**

   Start the Oracle installer and select the option to create a database. Alternatively, depending on your Oracle version, you can use the Oracle `svrmgrl`(1M) command to create the database.

   During creation, ensure that all database-related files are placed in the appropriate location, either on shared global devices or on the cluster file system.

3. **Verify that the file names of your control files match the file names in your configuration files.**

4. **Create the `v$sysstat` view.**

   Run the catalog scripts that create the `v$sysstat` view. The Sun Cluster fault monitoring scripts use this view.

Where to Go From Here

When you have completed the work in this section, go to "Setting Up Oracle Database Permissions" on page 23.

Setting Up Oracle Database Permissions

Use this procedure to set up Oracle database permissions.

▼ How to Set Up Oracle Database Permissions

When completing Step 1 of this procedure, select and configure either the Oracle authentication method or the Solaris authentication method for fault-monitoring access.

1. **Enable access for the user and password to be used for fault monitoring.**

   To complete this step, perform *one* of the following tasks.

- **To use the Oracle authentication method –** For all supported Oracle releases, enter the following script into the screen that the `srvmgrl` command displays to enable access.

```
# svrmgrl

    connect internal;
        grant connect, resource to user identified by passwd;
        alter user user default tablespace system quota 1m on
            system;
        grant select on v_$sysstat to user;
        grant create session to user;
        grant create table to user;
    disconnect;

   exit;
```

- **To use the Solaris authentication method –** Grant permission for the database to use Solaris authentication.

Note – The user for which you enable Solaris authentication is the user who owns the files under the `$ORACLE_HOME` directory. The following code sample shows that the user `oracle` owns these files.

```
# svrmgrl

    connect internal;
        create user ops$oracle identified by externally
            default tablespace system quota 1m on system;
        grant connect, resource to ops$oracle;
        grant select on v_$sysstat to ops$oracle;
        grant create session to ops$oracle;
        grant create table to ops$oracle;
    disconnect;

   exit;
```

2. **Configure NET8 for the Sun Cluster software.**

   The `listener.ora` and `tnsnames.ora` files must be accessible from all the nodes in the cluster. Place these files either under the cluster file system or in the local file system of each node that can potentially run the Oracle resources.

Note – If you place the `listener.ora` and `tnsnames.ora` files in a location other than the `/var/opt/oracle` directory or the `$ORACLE_HOME/network/admin` directory, then you must specify `TNS_ADMIN` or an equivalent Oracle variable (see the Oracle documentation for details) in a user-environment file. You must also run the `scrgadm`(1M) command to set the resource extension parameter `User_env`, which will source the user-environment file.

The Sun Cluster HA for Oracle data service imposes no restrictions on the listener name—it can be any valid Oracle listener name.

The following code sample identifies the lines in `listener.ora` that are updated.

```
LISTENER =
    (ADDRESS_LIST =
        (ADDRESS =
            (PROTOCOL = TCP)
            (HOST = logical-hostname) <- use logical hostname
            (PORT = 1527)
        )
    )
.
.
SID_LIST_LISTENER =
    .
        .
                (SID_NAME = SID) <- Database name, default is ORCL
```

The following code sample identifies the lines in `tnsnames.ora` that are updated on client machines.

```
service_name =
    .
        .
                (ADDRESS =
                    (PROTOCOL = TCP)
                    (HOST = logicalhostname)<- logical hostname
                    (PORT = 1527) <- must match port in LISTENER.ORA
                )
            )
            (CONNECT_DATA =
                (SID = <SID>)) <- database name, default is ORCL
```

The following example shows how to update the `listener.ora` and tnsnames.ora files given the following Oracle instances.

Instance	Logical Host	Listener
ora8	hadbms3	LISTENER-ora8
ora7	hadbms4	LISTENER-ora7

The corresponding listener.ora entries are the following entries.

```
LISTENER-ora7 =
    (ADDRESS_LIST =
        (ADDRESS =
            (PROTOCOL = TCP)
            (HOST = hadbms4)
            (PORT = 1530)
        )
    )
SID_LIST_LISTENER-ora7 =
    (SID_LIST =
        (SID_DESC =
            (SID_NAME = ora7)
        )
    )
LISTENER-ora8 =
  (ADDRESS_LIST =
    (ADDRESS= (PROTOCOL=TCP) (HOST=hadbms3)(PORT=1806))
  )
SID_LIST_LISTENER-ora8 =
  (SID_LIST =
     (SID_DESC =
        (SID_NAME = ora8)
     )
  )
```

The corresponding `tnsnames.ora` entries are the following entries.

```
ora8 =
(DESCRIPTION =
   (ADDRESS_LIST =
       (ADDRESS = (PROTOCOL = TCP)
       (HOST = hadbms3)
       (PORT = 1806))
    )
    (CONNECT_DATA = (SID = ora8))
)
ora7 =
(DESCRIPTION =
  (ADDRESS_LIST =
        (ADDRESS =
           (PROTOCOL = TCP)
           (HOST = hadbms4)
           (PORT = 1530))
  )
    (CONNECT_DATA = (SID = ora7))
)
```

3. **Verify that the Sun Cluster software is installed and running on all nodes.**

```
# scstat
```

Where to Go From Here

Go to "Installing Sun Cluster HA for Oracle Packages" on page 27 to register and configure the Sun Cluster HA for Oracle data service.

Installing Sun Cluster HA for Oracle Packages

Use the `scinstall`(1M) utility to install `SUNWscor`, the Sun Cluster HA for Oracle data-service package, on a cluster. Do not use the `-s` option to non-interactive `scinstall` to install all data-service packages.

If you installed the SUNWscor data-service package as part of your initial Sun Cluster installation, proceed to "Registering and Configuring Sun Cluster HA for Oracle" on page 28. Otherwise, use the following procedure to install the SUNWscor package.

▼ How to Install Sun Cluster HA for Oracle Packages

You need the Sun Cluster Agents CD to complete this procedure. Perform this procedure on all cluster nodes that run the Sun Cluster HA for Oracle data service.

1. **Load the Agents CD into the CD-ROM drive.**
2. **Run the `scinstall` utility with no options.**

   This step starts the `scinstall` utility in interactive mode.
3. **Select the Add Support for New Data Service to This Cluster Node menu option.**

   This option enables you to load software for any data services that exist on the CD.
4. **Exit the `scinstall` utility.**
5. **Unload the CD from the drive.**

Where to Go From Here

See "Registering and Configuring Sun Cluster HA for Oracle" on page 28 to register the Sun Cluster HA for Oracle data service and configure the cluster for the data service.

Registering and Configuring Sun Cluster HA for Oracle

Register and configure the Sun Cluster HA for Oracle data service as a failover data service. You must register the data service and configure resource groups and resources for the Oracle server and listener. See Chapter 1 and the *Sun Cluster 3.0 U1 Concepts* document for details on resources and resource groups.

▼ How to Register and Configure Sun Cluster HA for Oracle

This procedure describes how to use the scrgadm command to register and configure the Sun Cluster HA for Oracle data service.

Note – Other options also enable you to register and configure the data service. See "Tools for Data-Service Resource Administration" on page 8 for details about these options.

You must have the following information to perform this procedure.

- The names of the cluster nodes that master the data service.
- The logical hostname that clients use to access the data service. Normally, you set up this IP address when you install the cluster. See the section on how to set up logical hostnames in the *Sun Cluster 3.0 U1 Installation Guide* for details.
- The path to the Oracle application binaries for the resources that you plan to configure.

Note – Perform this procedure on any cluster member.

---

1. **Become superuser on a cluster member.**

2. **Run the scrgadm command to register the resource types for the data service.**

   For the Sun Cluster HA for Oracle data service, you register two resource types, SUNW.oracle_server and SUNW.oracle_listener, as follows.

```
# scrgadm -a -t SUNW.oracle_server
# scrgadm -a -t SUNW.oracle_listener
```

-a	Adds the data service resource type.
-t SUNW.oracle_*type*	Specifies the predefined resource type name for your data service.

3. **Create a failover resource group to hold the network and application resources.**

   You can optionally select the set of nodes on which the data service can run with the -h option, as follows.

```
# scrgadm -a -g resource-group [-h nodelist]
```

`-g` *resource-group*
Specifies the name of the resource group. This name can be your choice but must be unique for resource groups within the cluster.

`-h` *nodelist*
Specifies an optional comma-separated list of physical node names or IDs that identify potential masters. The order here determines the order in which the nodes are considered as primary during failover.

Note – Use the `-h` option to specify the order of the node list. If all the nodes in the cluster are potential masters, you do not need to use the `-h` option.

---

4. **Verify that all logical hostnames that you use have been added to your name-service database.**

   You should have performed this verification during the Sun Cluster installation.

Note – Ensure that all logical hostnames are present in the server's and client's `/etc/hosts` file to avoid any failures because of name-service lookup.

---

5. **Add a logical hostname to the failover resource group.**

```
# scrgadm -a -L -g resource-group -l logical-hostname \
[-j resource] [-n netiflist]
```

`-l` *logical-hostname*
Specifies a logical hostname.

`-j` *resource*
An optional name for the logical-hostname resource. If a name is not specified, the default resource name is the first name to appear after the `-l` option.

`-n` *netiflist*
An optional comma-separated list that identifies the NAFO groups on each node. All nodes in *nodelist* of the resource group must be represented in the *netiflist*. If you do not specify this option, `scrgadm` attempts to discover a net adapter on the subnet that the *hostname* list identifies for each node in *nodelist*.

6. **Create Oracle application resources in the failover resource group.**

```
# scrgadm -a -j resource -g resource-group \
-t SUNW.oracle_server \
-x Connect_string=user/passwd \
-x ORACLE_SID=instance \
-x ORACLE_HOME=Oracle-home \
-x Alert_log_file=path-to-log
```

```
# scrgadm -a -j resource -g resource-group \
-t SUNW.oracle_listener \
-x LISTENER_NAME=listener \
-x ORACLE_HOME=Oracle-home
```

`-j` *resource*

Specifies the name of the resource to add.

`-g` *resource-group*

Specifies the name of the resource group into which the resources are to be placed.

`-t SUNW.oracle_server/listener`

Specifies the type of the resource to add.

`-x Alert_log_file=`*path-to-log*

Sets the path under `$ORACLE_HOME` for the server message log.

`-x Connect_string=`*user/passwd*

The user and password that the fault monitor uses to connect to the database. These settings must agree with the permissions that you set up in "How to Set Up Oracle Database Permissions" on page 23. If you use Solaris authorization, type a slash (`/`) instead of the user name and password.

`-x ORACLE_SID=`*instance*

Sets the Oracle system identifier.

`-x LISTENER_NAME=`*listener*

Sets the name of the Oracle listener instance. This name must match the corresponding entry in `listener.ora`.

`-x ORACLE_HOME=`*Oracle-home*

Sets the path to the Oracle home directory.

Tip – When a fault occurs in an Oracle server resource that causes a restart, the whole resource group is restarted. Any other resources (such as Apache or DNS) in the resource group are restarted, even if they did not have a fault. To prevent other resources from being restarted along with an Oracle server resource, put them in a separate resource group.

Note – Optionally, you can set additional extension properties that belong to the Oracle data service to override the default value. See "Configuring Sun Cluster HA for Oracle Extension Properties" on page 35 for a list of extension properties.

7. **Run the `scswitch` command to complete the following tasks.**

- Enable the resource and fault monitoring.
- Move the resource group into a managed state.
- Bring the resource group online.

```
# scswitch -Z -g resource-group
```

`-Z`
Enables the resource and monitor, moves the resource group to the managed state, and brings it online.

`-g` *resource-group*
Specifies the name of the resource group.

Example – Registering Sun Cluster HA for Oracle

The following example shows how to register the Sun Cluster HA for Oracle data service on a two-node cluster.

```
Cluster Information
Node names: phys-schost-1, phys-schost-2
Logical Hostname: schost-1
Resource group: resource-group-1 (failover resource group)
Oracle Resources: oracle-server-1, oracle-listener-1
Oracle Instances: ora-lsnr (listener), ora-srvr (server)

(Add the failover resource group to contain all the resources.)
# scrgadm -a -g resource-group-1

(Add the logical hostname resource to the resource group.)
# scrgadm -a -L -g resource-group-1 -l schost-1

(Register the Oracle resource types)
# scrgadm -a -t SUNW.oracle_server
# scrgadm -a -t SUNW.oracle_listener

(Add the Oracle application resources to the resource group.)
# scrgadm -a -j oracle-server-1 -g resource-group-1 \
-t SUNW.oracle_server -x ORACLE_HOME=/global/oracle \
-x Alert_log_file=/global/oracle/message-log \
-x ORACLE_SID=ora-srvr -x Connect_string=scott/tiger

# scrgadm -a -j oracle-listener-1 -g resource-group-1 \
-t SUNW.oracle_listener -x ORACLE_HOME=/global/oracle \
-x LISTENER_NAME=ora-lsnr

(Bring the resource group online.)
# scswitch -Z -g resource-group-1
```

▼ How to Configure `SUNW.HAStorage` Resource Type

The `SUNW.HAStorage` resource type synchronizes actions between HA storage and the data service. The Sun Cluster HA for Oracle data service is disk-intensive, and therefore you should configure the `SUNW.HAStorage` resource type.

See the SUNW.HAStorage(5) man page and "Relationship Between Resource Groups and Disk Device Groups" on page 4 for background information. See "How to Set Up SUNW.HAStorage Resource Type for New Resources" on page 239 for the procedure.

Where to Go From Here

Go to "Verifying the Sun Cluster HA for Oracle Installation" on page 34 after you register and configure the Sun Cluster HA for Oracle data service.

Verifying the Sun Cluster HA for Oracle Installation

Perform the following verification tests to make sure that you have correctly installed the Sun Cluster HA for Oracle data service.

These sanity checks ensure that all the nodes that run the Sun Cluster HA for Oracle data service can start the Oracle instance and that the other nodes in the configuration can access the Oracle instance. Perform these sanity checks to isolate any problems starting the Oracle software from the Sun Cluster HA for Oracle data service.

How to Verify the Sun Cluster HA for Oracle Installation

1. **Log in as *oracle_id* to the node that currently masters the Oracle resource group.**

2. **Set the environment variables ORACLE_SID and ORACLE_HOME.**

3. **Confirm that you can start the Oracle instance from this node.**

4. **Confirm that you can connect to the Oracle instance.**

   Use the sqlplus command with the tns_service variable defined in the tnsnames.ora file.

```
# sqlplus user/passwd@tns_service
```

5. **Shut down the Oracle instance.**

   The Sun Cluster software will restart the Oracle instance because the Oracle instance is under Sun Cluster control.

6. **Switch the resource group that contains the Oracle database resource to another cluster member.**

   The following example shows how to complete this step.

   ```
   # scswitch -z -g resource-group -h node
   ```

7. **Log in as *oracle_id* to the node that now contains the resource group.**

8. **Repeat Step 3 and Step 4 to confirm interactions with the Oracle instance.**

Oracle Clients

Clients must always refer to the database by using the logical hostname (an IP address that can move between physical nodes during failover), not the physical hostname (a machine name).

For example, in the `tnsnames.ora` file, you must specify the logical hostname as the host on which the database instance is running. See "How to Set Up Oracle Database Permissions" on page 23.

Note – Oracle client-server connections cannot survive a Sun Cluster HA for Oracle switchover. The client application must be prepared to handle disconnection and reconnection or recovery as appropriate. A transaction monitor might simplify the application. Further, Sun Cluster HA for Oracle node recovery time is application dependent.

Configuring Sun Cluster HA for Oracle Extension Properties

This section describes the Sun Cluster HA for Oracle extension properties. Typically, you use the command line `scrgadm -x` *parameter=value* to configure the extension properties when you create the Oracle server and listener resources. You can also use the procedures described in Chapter 11 to configure them later. See Appendix A for details on all Sun Cluster properties.

TABLE 2-2 describes the extension properties that you can set for the Oracle listener resource. The required extension property for creating an Oracle listener resource is the ORACLE_HOME property. You can update some extension properties dynamically. You can update others, however, only when you create the resource. The Tunable column of the following table indicates when you can update each property.

TABLE 2-2 Sun Cluster HA for Oracle Listener Extension Properties

Name/Data Type	Default	Range	Tunable	Description
LISTENER_NAME (string)	LISTENER	None	When disabled	The name of the Oracle listener.
ORACLE_HOME (string)	None	Minimum = 1	When disabled	The path to the Oracle home directory.
User_env (string)	“”	None	Any time	A file that contains environment variables to be set before listener startup and shutdown.

TABLE 2-3 describes the extension properties that you can set for the Oracle server. The only extension properties that you are required to set for the Oracle server are the ORACLE_HOME, ORACLE_SID, Alert_log_file, and Connect_string properties.

TABLE 2-3 Sun Cluster HA for Oracle Server Extension Properties

Name/Data Type	Default	Range	Tunable	Description
Alert_log_file (string)	None	Minimum = 1	Any time	Oracle alert log file.
Connect_cycle (integer)	5	0 – 99,999	Any time	The number of fault monitor probe cycles before disconnecting from the database.
Connect_string (string)	None	Minimum = 1	Any time	The Oracle user and password that the fault monitor uses to connect to the database.
ORACLE_HOME (string)	None	Minimum = 1	When disabled	The path to the Oracle home directory.
ORACLE_SID (string)	None	Minimum = 1	When disabled	The Oracle system identifier.
Parameter_file (string)	“”	Minimum = 0	Any time	The Oracle parameter file. If the Oracle parameter file is not specified, this property defaults to Oracle’s default.

TABLE 2-3 Sun Cluster HA for Oracle Server Extension Properties

Name/Data Type	Default	Range	Tunable	Description
`Probe_timeout` (integer)	`60`	`0 – 99,999`	Any time	The time-out value (in seconds) that the fault monitor uses to probe an Oracle server instance.
`User_env` (string)	“”	None	Any time	A file that contains environment variables to be set before listener startup and shutdown.
`Wait_for_online` (Boolean)	`True`	None	Any time	Wait in the `START` method until the database is online.

Sun Cluster HA for Oracle Fault Monitor

The two fault monitors for the Sun Cluster HA for Oracle data service are a server and a listener monitor.

Oracle Server Fault Monitor

The fault monitor for the Oracle server uses a request to the server to query the health of the server.

The server fault monitor consists of the following two processes.

- a main fault-monitor process, which performs error lookup and `scha_control` actions
- a database client fault probe, which performs database transactions

All database connections from the probe are performed as user `oracle`. The main fault monitor determines that the operation is successful if the database is online and no errors are returned during the transaction.

If the database transaction fails, the main process checks the internal action table for an action to be performed and performs the predetermined action. If the action executes an external program, it is executed as a separate process in the background. Possible actions include the following.

- Switchover
- Stopping the server
- Restarting the server
- Stopping the resource group
- Restarting the resource group

The probe uses the time-out value that is set in the resource property Probe_timeout to determine how much time to spend to successfully probe Oracle.

The server fault monitor also scans Oracle's alert_log_file and takes action based on any errors that the fault monitor finds.

The server fault monitor is started through pmfadm to make the monitor highly available. If the monitor is killed for any reason, the Process Monitor Facility (PMF) automatically restarts the monitor.

Oracle Listener Fault Monitor

The Oracle listener fault monitor checks the status of an Oracle listener.

If the listener is running, the Oracle listener fault monitor considers a probe successful. If the fault monitor detects an error, the listener is restarted.

The listener probe is started through pmfadm to make the probe highly available. If the probe is killed, PMF automatically restarts the probe.

If a problem occurs with the listener during a probe, the probe tries to restart the listener. The value set in the resource property Retry_count determines the maximum number of times the probe attempts the restart. If, after trying for the maximum number of times, the probe is still unsuccessful, the probe stops the fault monitor and does not switch over the resource group.

CHAPTER 3

Installing and Configuring Sun Cluster HA for iPlanet Web Server

This chapter provides the procedures to install and configure the Sun Cluster HA for iPlanet Web Server data service. This data service was formerly known as Sun Cluster HA for Netscape™ HTTP. Some error messages from the application might still use the name Netscape, but the messages refer to iPlanet Web Server.

This chapter contains the following procedures.

- "How to Install an iPlanet Web Server" on page 43
- "How to Configure an iPlanet Web Server" on page 44
- "How to Install Sun Cluster HA for iPlanet Web Server Packages" on page 49
- "How to Register and Configure Sun Cluster HA for iPlanet Web Server" on page 50
- "How to Configure `SUNW.HAStorage` Resource Type" on page 59

You can configure the Sun Cluster HA for iPlanet Web Server data service as a failover or scalable service. See Chapter 1 and the *Sun Cluster 3.0 U1 Concepts* document for general information about data services, resource groups, resources, and other related topics.

Note – You can use SunPlex Manager to install and configure this data service. See the SunPlex Manager online help for details.

Note – If you run multiple data services in your Sun Cluster configuration, you can set up the data services in any order, with the following exception. If the Sun Cluster HA for iPlanet Web Server data service depends on the Sun Cluster HA for DNS data service, you must set up DNS first. See Chapter 6 for details.

The Solaris operating environment includes the DNS software. If the cluster is to obtain the DNS service from another server, then configure the cluster to be a DNS client first.

Note – After installation, do not manually start and stop the iPlanet Web server except by using the cluster administration command `scswitch`(1M). See the man page for details. After the iPlanet Web Server is started, the Sun Cluster software controls it.

Planning the Installation and Configuration

Use the following section in conjunction with your configuration worksheets as a checklist before you install and configure the Sun Cluster HA for iPlanet Web Server data service.

Consider the following questions before you start your installation.

- Will you be running the Sun Cluster HA for iPlanet Web Server data service as a failover or as a scalable service? See the *Sun Cluster 3.0 U1 Concepts* document for information on the two types of services. For scalable services, consider the following questions.
  - What nodes will host the scalable service? In most cases, you will want to scale across all nodes. You can, however, limit the set of nodes that host the service.
  - Will your iPlanet Web Server instances require sticky IP? Sticky IP is a resource property setting, `Load_balancing_policy`, which stores the client state in memory so return traffic from the same node always goes to the same cluster node. You can choose from several load balancing policies, as described in the table on resource properties in Appendix A.

    Exercise caution when changing `Load_balancing_weights` for an online scalable service that has `Load_balancing_policy` set to `LB_STICKY` or `LB_STICKY_WILD`. Changing those properties while the service is online can cause existing client affinities to be reset, and hence a different node might service a subsequent client request even if another cluster member had previously serviced the client.

Similarly, when a new instance of the service is started on a cluster, existing client affinities might be reset.

- Where will the Web server root reside?
- Does the Web server serve data for another highly available application? If so, resource dependencies might exist between the resources so that one starts or stops before the other. See Appendix A for a description of the resource property `Resource_dependencies` that sets these dependencies.
- Determine the resource groups to use for network addresses and application resources and the dependencies between them. See Appendix A for a description of the resource group property `RG_dependencies` that sets these dependencies.
- Provide the logical hostname (for failover services) or shared address (for scalable services) for clients to use to access the data service.
- Because you can configure iPlanet Web Server to bind to `INADDR_ANY`, if you plan to run multiple instances of the iPlanet Web Server data service or multiple data services on the same node, each instance must bind to a unique network address and port number.
- Determine the entries for the `Confdir_list` and `Port_list` properties. For failover services, both of these properties can have only one entry. For scalable services, they can have multiple entries. The number of entries, however, must be the same and must map to each other in the order specified. See "How to Register and Configure Sun Cluster HA for iPlanet Web Server" on page 50 for details.
- Determine where to place logs, error files, and the PID file on the local file system.
- Determine where to place the contents on the cluster file system.

Installing and Configuring Sun Cluster HA for iPlanet Web Server

The following table lists the sections that describe the installation and configuration tasks.

TABLE 3-1 Task Map: Installing and Configuring Sun Cluster HA for iPlanet Web Server

Task	For Instructions, Go To
Install iPlanet Web Server	"Installing and Configuring an iPlanet Web Server" on page 42
Install the Sun Cluster HA for iPlanet Web Server data-service packages	"Installing Sun Cluster HA for iPlanet Web Server Packages" on page 48
Configure the Sun Cluster HA for iPlanet Web Server data service	"Registering and Configuring Sun Cluster HA for iPlanet Web Server" on page 49
Configure resource extension properties	"Configuring Sun Cluster HA for iPlanet Web Server Extension Properties" on page 59
View fault-monitor information	"Sun Cluster HA for iPlanet Web Server Fault Monitor" on page 61

Installing and Configuring an iPlanet Web Server

This section describes the steps to use the `setup` command to perform the following tasks.

- Install the iPlanet Web Server.
- Enable the iPlanet Web Server to run as the Sun Cluster HA for iPlanet Web Server data service.

Note – You must follow certain conventions when you configure URL mappings for the Web server. For example, to preserve availability when setting the CGI directory, you must locate the mapped directories on the cluster file system. In this example, you map your CGI directory to `/global/`*pathname*`/cgi-bin`.

Note – In situations where the CGI programs access "back-end" servers, such as an RDBMS, ensure that the Sun Cluster software also controls the "back-end" server. If the server is an RDBMS that the Sun Cluster software supports, use one of the highly available RDBMS packages. Alternatively, you can use the APIs documented in the *Sun Cluster 3.0 U1 Data Services Developers' Guide* to put the server under Sun Cluster control.

▼ How to Install an iPlanet Web Server

To perform this procedure, you need the following information about your configuration.

- The server root directory (the path to the application binaries). You can install the binaries on the local disks or on the cluster file system. For a discussion of the advantages and disadvantages of each location, see "Determining the Location of the Application Binaries" on page 3.
- The logical hostname (for failover services) or shared address (for scalable services) that clients use to access the data service. You must configure these addresses, and they must be online.

Note – If you run the Sun Cluster HA for iPlanet Web Server service and another HTTP server and they use the same network resources, configure them to listen on different ports. Otherwise, a port conflict might occur between the two servers.

1. **Become superuser on a cluster member.**

2. **Run the `setup` command from the iPlanet install directory on the CD.**

3. **When prompted, enter the location where the iPlanet server binaries will be installed.**

   You can specify a location on the cluster file system or on local disks for the location of the install. If you choose to install on local disks, run the `setup` command on all the cluster nodes that are potential primaries of the network resource (logical hostname or shared address) specified in the next step.

4. **When prompted for a machine name, enter the logical hostname on which the iPlanet server depends and the appropriate DNS domain name.**

   A full logical hostname is of the format *network-resource.domainname*, such as `schost-1.sun.com`.

Note – For the Sun Cluster HA for iPlanet Web Server data service to fail over correctly, you must use either the logical hostname or shared-address resource name (rather than the physical hostname) here and everywhere else that you are asked.

5. **Select Run Admin Server as Root when you are asked.**

   Note the port number that the iPlanet install script selects for the administration server if you want to use this default value later when configuring an instance of the iPlanet Web server. Otherwise, you can specify a different port number when you configure the iPlanet server instance.

6. **Type a Server Administrator ID and a chosen password when you are asked.**

   Follow the guidelines for your system.

   When a message displays that the admin server will be started, your installation is ready for configuration.

Where to Go From Here

To configure the Web server, see the next section, "How to Configure an iPlanet Web Server" on page 44.

▼ How to Configure an iPlanet Web Server

This procedure describes how to configure an instance of the iPlanet Web server to be highly available. Use the Netscape browser to interact with this procedure.

Consider the following points before you perform this procedure.

- Before you start, ensure that you have installed the browser on a machine that can access the network on which the cluster resides. You can install the browser on a cluster node or on the administrative workstation for the cluster.
- Your configuration files can reside on either a local file system or on the cluster file system.
- Any certificates that are installed for the secure instances must be installed from all cluster nodes. This installation involves running the admin console on each node. Thus, if a cluster has nodes `n1`, `n2`, `n3`, and `n4`, the installation steps are as follows.

1. Run the admin server on node `n1`.

2. From your Web browser, connect to the admin server as `http://n1.`*domain*:*port*—for example, `http://n1.eng.sun.com:8888`—or whatever you specified as the admin server port. The port is typically `8888`.

3. Install the certificate.

4. Stop the admin server on node `n1` and run the admin server from node `n2`.

5. From the Web browser, connect to the new admin server as `http://n2.`*domain*`:`*port*, for example, `http://n2.eng.sun.com:8888`.

6. Repeat these steps for nodes `n3` and `n4`.

   After you have considered the preceding points, complete the following steps.

1. **From the administrative workstation or a cluster node, start the Netscape browser.**

2. **On one of the cluster nodes, go to the directory `https-admserv`, then start the iPlanet admin server.**

```
# cd https-admserv
# ./start
```

3. **Enter the URL of the iPlanet admin server in the Netscape browser.**

   The URL consists of the physical hostname and port number that the iPlanet installation script established in Step 4 of the server installation procedure, for example, `n1.eng.sun.com:8888`. When you perform Step 2 of this procedure, the `./start` command displays the admin URL.

   When prompted, use the user ID and password you specified in Step 6 of the server installation procedure to log in to the iPlanet administration server interface.

4. **Begin to administer the iPlanet Web Server instance that was created.**

   If you need another instance, create a new one.

   The administration graphical interface provides a form with details of the iPlanet server configuration. You can accept the defaults on the form, with the following exceptions.

   - Verify that the server name is correct.
   - Verify that the server user is set as superuser.
   - Change the bind address field to one of the following addresses.
     - A logical hostname or shared address if you use DNS as your name service
     - The IP address associated with the logical hostname or shared address if you use NIS as your name service

5. **Create a directory on the local disk of all the nodes to hold the logs, error files, and PID file that iPlanet Web Server manages.**

   For iPlanet to work correctly, these files must be located on each node of the cluster, not on the cluster file system.

   Choose a location on the local disk that is the same for all the nodes in the cluster. Use the `mkdir -p` command to create the directory. Make `nobody` the owner of this directory.

   The following example shows how to complete this step.

```
phys-schost-1# mkdir -p /var/pathname/http-instance/logs/
```

Note – If you anticipate large error logs and PID files, do not put them in a directory under `/var` because they will overwhelm this directory. Rather, create a directory in a partition with adequate space to handle large files.

---

6. **Edit the `ErrorLog` and `PidLog` entries in the `magnus.conf` file to reflect the directory created in the previous step, and synchronize the changes from the administrator's interface.**

   The `magnus.conf` file specifies the locations for the error files and PID files. Edit this file to change the error and PID file locations to the directory that you created in Step 5. The `magnus.conf` file is located in the `config` directory of the iPlanet server instance. If the instance directory is located on the local file system, you must modify the `magnus.conf` file on each of the nodes.

   Change the entries as follows.

```
# Current ErrorLog and PidLog entries
ErrorLog /global/data/netscape/https-schost-1/logs/error
PidLog /global/data/netscape/https-insecure-schost-1/logs/pid

# New entries
ErrorLog /var/pathname/http-instance/logs/error
PidLog /var/pathname/http-instance/logs/pid
```

   As soon as the administrator's interface detects your changes, the interface displays a warning message, as follows.

```
Warning: Manual edits not loaded
Some configuration files have been edited by hand. Use the Apply
button on the upper right side of the screen to load the latest
configuration files.
```

   Click Apply as prompted.

   The administrator's interface then displays the following warning.

```
Configuration files have been edited by hand. Use this button to
load the latest configuration files.
```

   Click Load Configuration Files as prompted.

7. **Use the administrator's interface to set the location of the access log file.**

   From the administration graphical interface, click the Preferences tab and then Logging Options on the side bar. A form is then displayed for configuring the Access Log parameter.

   Change the location of the log file to the directory that you created in Step 5.

   For example, make the following changes to the log file.

```
Log File: /var/pathname/http-instance/logs/access
```

8. **Click Save to save your changes.**

   Do *not* click Save and Apply—doing so starts iPlanet Web Server.

Where to Go From Here

If you have not installed the Sun Cluster HA for iPlanet Web Server data-service packages from the Sun Cluster Agents CD, go to "Installing Sun Cluster HA for iPlanet Web Server Packages" on page 48. Otherwise, go to "Registering and Configuring Sun Cluster HA for iPlanet Web Server" on page 49.

Installing Sun Cluster HA for iPlanet Web Server Packages

You can use the `scinstall`(1M) utility to install `SUNWschtt`, the Sun Cluster HA for iPlanet Web Server data-service package, on a cluster. Do not use the `-s` option to non-interactive `scinstall` to install all data-service packages on the CD.

If you installed the data-service packages during your initial Sun Cluster installation, proceed to "Registering and Configuring Sun Cluster HA for iPlanet Web Server" on page 49. Otherwise, use the following procedure to install the `SUNWschtt` package.

▼ How to Install Sun Cluster HA for iPlanet Web Server Packages

You need the Sun Cluster Agents CD to complete this procedure. Run this procedure on all the cluster nodes that will run the Sun Cluster HA for iPlanet Web Server data service.

1. **Load the Agents CD into the CD-ROM drive.**
2. **Run the `scinstall` utility with no options.**

   This step starts the `scinstall` utility in interactive mode.
3. **Select the Add Support for New Data Service to This Cluster Node menu option.**

   This option enables you to load software for any data services that exist on the CD.
4. **Exit the `scinstall` utility.**
5. **Unload the CD from the drive.**

Where to Go From Here

See "Registering and Configuring Sun Cluster HA for iPlanet Web Server" on page 49 to register the Sun Cluster HA for iPlanet Web Server data service and configure the cluster for the data service.

Registering and Configuring Sun Cluster HA for iPlanet Web Server

You can configure the Sun Cluster HA for iPlanet Web Server data service as a failover service or as a scalable service. You must include some additional steps to configure iPlanet as a scalable service. In the first procedure in this section, these additional steps begin with a notation that they are required for scalable services only. Individual examples of a failover service and a scalable service follow the procedure.

▼ How to Register and Configure Sun Cluster HA for iPlanet Web Server

This procedure describes how to use the scrgadm(1M) command to register and configure the Sun Cluster HA for iPlanet Web Server data service.

Note – Other options also enable you to register and configure the data service. See "Tools for Data-Service Resource Administration" on page 8 for details about these options.

To perform this procedure, you must have the following information.

- The name of the resource type for the Sun Cluster HA for iPlanet Web Server data service. This name is SUNW.iws.
- The names of the cluster nodes that master the data service. For a failover service, only one node can master a data service at a time.
- The logical hostname (for failover services) or shared address (for scalable services) that clients use to access the data service.
- The path to the iPlanet binaries. You can install the binaries on the local disks or the cluster file system. See "Determining the Location of the Application Binaries" on page 3 for a discussion of the advantages and disadvantages of each location.

Note – The Network_resources_used setting on the iPlanet application resource determines the set of IP addresses that iPlanet Web Server uses. The Port_list setting on the resource determines the list of port numbers that iPlanet Web Server uses. The fault monitor assumes that the iPlanet Web Server daemon is listening on all combinations of IP and port. If you have customized your magnus.conf file for the iPlanet Web Server to listen on different port numbers (in addition to port 80), your resultant magnus.conf file must contain all possible combinations of IP address and ports. The fault monitor attempts to probe all such combinations and starts to fail if the iPlanet Web Server is not listening on a particular IP address-port combination. If the iPlanet Web Server does not serve all IP address-port combinations, you must break the iPlanet Web Server into separate instances that do.

Note – Perform this procedure on any cluster member.

---

1. **Become superuser on a cluster member.**

2. **Register the resource type for the Sun Cluster HA for iPlanet Web Server data service.**

```
# scrgadm -a -t SUNW.iws
```

`-a` Adds the data-service resource type.

`-t SUNW.iws` Specifies the predefined resource-type name for your data service.

3. **Create a failover resource group to hold the network and application resources.**

   For failover services, this resource group also holds the application resources.

   You can optionally select the set of nodes on which the data service can run with the `-h` option.

```
# scrgadm -a -g resource-group [-h nodelist]
```

`-g` *resource-group* Specifies the name of the failover resource group. This name can be your choice but must be unique for resource groups within the cluster.

`-h` *nodelist* An optional comma-separated list of physical node names or IDs that identify potential masters. The order here determines the order in which the nodes are considered as primary during failover.

Note – Use `-h` to specify the order of the node list. If all the nodes in the cluster are potential masters, you need not use the `-h` option.

4. **Verify that all network addresses that you use have been added to your name-service database.**

   You should have performed this verification during the Sun Cluster installation. See the planning chapter in the *Sun Cluster 3.0 U1 Installation Guide* for details.

Note – To avoid any failures because of name-service lookup, ensure that all logical hostnames and shared addresses are present in the server's and client's `/etc/hosts` file. Configure name-service mapping in `/etc/nsswitch.conf` on the servers to first check the local files before trying to access NIS or NIS+.

5. **Add a network resource (logical hostname or shared address) to the failover resource group.**

```
# scrgadm -a {-S | -L} -g resource-group \
-l network-resource,… [-j resource] \
[-X auxnodelist=node, …] [-n netiflist]
```

`-S` | `-L`	You use `-S` for shared-address resources or `-L` for logical-hostname resources.
`-g` *resource-group*	Specifies the name of the failover resource group.
`-l` *network-resource, …*	Specifies a comma-separated list of network resources to add. You can use the `-j` option to specify a name for the resources. If you do not do so, the network resources have the name of the first entry on the list.
`-j` *resource*	Specifies an optional resource name. If you do not supply this name, the name of the network resource defaults to the first name specified after the `-l` option.
`-X` *auxnodelist=node, …*	Specifies an optional comma-separated list of physical node IDs that identify cluster nodes that can host the shared address but never serve as a primary if failover occurs. These nodes are mutually exclusive with the nodes identified in *nodelist* for the resource group, if specified.
`-n` *netiflist*	Specifies an optional comma-separated list that identifies the NAFO groups on each node. All nodes in *nodelist* of the resource group must be represented in *netiflist*. If you do not specify this option, `scrgadm` attempts to discover a net adapter on the subnet that the *hostname* list identifies for each node in *nodelist*.

6. **For scalable services only – Create a scalable resource group to run on all desired nodes of the cluster.**

   If you run the Sun Cluster HA for iPlanet Web Server data service as a failover data service, do not perform this step—go to Step 8.

   Create a resource group to hold a data-service application resource. You must specify the maximum and desired number of primary nodes, as well as a dependency between this resource group and the failover resource group that you created in Step 3. This dependency ensures that in the event of failover, the resource manager starts the network resource before starting any data services that depend on the network resource.

```
# scrgadm -a -g resource-group \
-y Maximum_primaries=m -y Desired_primaries=n \
-y RG_dependencies=resource-group
```

-y Maximum_primaries=*m*
Specifies the maximum number of active primary nodes allowed for this resource group. If you do not assign a value to this property, the default is 1.

-y Desired_primaries=*n*
Specifies the desired number of active primary nodes allowed for this resource group. If you do not assign a value to this property, the default is 1.

-y RG_dependencies=*resource-group*
Identifies the resource group that contains the shared-address resource on which the resource group being created depends.

7. **For scalable services only – Create an application resource in the scalable resource group.**

   If you run the Sun Cluster HA for iPlanet Web Server data service as a failover data service, do not perform this step—go to Step 8.

   You can repeat this step to add multiple application resources (such as secure and insecure versions) to the same resource group.

   You might also want to set load balancing for the data service. To do so, use the two standard resource properties Load_balancing_policy and Load_balancing_weights. See Appendix A for a description of these properties. Additionally, see the examples that follow this section.

```
# scrgadm -a -j resource -g resource-group \
-t resource-type -y Network_resources_used=network-resource, … \
-y Port_list=port-number/protocol, … -y Scalable=True \
-x Confdir_list=config-directory, …
```

`-j` *resource*
Specifies the name of the resource to add.

`-g` *resource-group*
Specifies the name of the scalable resource group into which the resources are to be placed.

`-t` *resource-type*
Specifies the type of the resource to add.

`-y Network_resources_used=`*network-resource, …*
Specifies a comma-separated list of network resources that identify the shared addresses that the data service uses.

`-y Port_list=`*port-number/protocol, …*
Specifies a comma-separated list of port numbers and protocol to be used, for example, `80/tcp,81/tcp`.

`-y Scalable=True`
Specifies a Boolean that is required for scalable services.

`-x Confdir_list=`*config-directory, …*
Specifies a comma-separated list of the locations of the iPlanet configuration files. The Sun Cluster HA for iPlanet Web Server data service requires this extension property.

Note – A one-to-one mapping applies for `Confdir_List` and `Port_List`, that is, each of the values in one list must correspond to the values in the other list in the order specified.

---

8. **For failover services only – Create an application resource in the failover resource group.**

   Perform this step only if you run the Sun Cluster HA for iPlanet Web Server data service as a failover data service. If you run the Sun Cluster HA for iPlanet Web Server data service as a scalable service, you must have performed Step 6 and Step 7 previously and must now go to Step 10.

   You can repeat this step to add multiple application resources (such as secure and insecure versions) to the same resource group.

```
# scrgadm -a -j resource -g resource-group \
-t resource-type -y Network_resources_used=logical-hostname-list \
-y Port_list=port-number/protocol \
-x Confdir_list=config-directory
```

`-j` *resource*	Specifies the name of the resource to add.
`-g` *resource-group*	Specifies the name of the failover resource group into which the resources are to be placed.
`-t` *resource-type*	Specifies the type of the resource to add.
`-y Network_resources_used=`*network-resource, …*	Specifies a comma-separated list of network resources that identify the logical hosts that the data service uses.
`-y Port_list=`*port-number/protocol*	Specifies the port number and protocol to use, for example, `80/tcp`. `Port_list` for failover services must have exactly one entry only because of the one-to-one mapping rule between `Port_list` and `Confdir_list`.
`-x Confdir_list=`*config-directory*	Specifies the location of the iPlanet configuration files. The `Confdir_list` file for failover services must have exactly one entry only. The *config-directory* must contain a directory called `config`. The Sun Cluster HA for iPlanet Web Server data service requires this extension property.

Note – Optionally, you can set additional extension properties that belong to the iPlanet data service to override the default value. See TABLE 3-2 for a list of these properties.

---

9. **Bring the failover resource group online.**

```
# scswitch -Z -g resource-group
```

`-Z`	Enables the network resource and fault monitoring, switches the resource group into a managed state, and brings the resource group online.
`-g` *resource-group*	Specifies the name of the failover resource group.

10. **For scalable services only – Bring the scalable resource group online.**

```
# scswitch -Z -g resource-group
```

`-Z` Enables the resource and monitor, moves the resource group to the managed state, and brings the resource group online.

`-g` *resource-group* Specifies the name of the scalable resource group.

Example – Registering Scalable Sun Cluster HA for iPlanet Web Server

The following example shows how to register a scalable iPlanet service.

```
Cluster Information
Node names: phys-schost-1, phys-schost-2
Shared address: schost-1
Resource groups: sa-resource-group-1 (for shared addresses),
    iws-resource-group-1 (for scalable iPlanet application resources)
Resources: schost-1 (shared address), iplanet-insecure-1 (insecure iPlanet
    application resource), iplanet-secure-1 (secure iPlanet application
    resource)

(Add a failover resource group to contain shared addresses.)
# scrgadm -a -g sa-resource-group-1

(Add the shared address resource to the failover resource group.)
# scrgadm -a -S -g sa-resource-group-1 -l schost-1

(Add a scalable resource group.)
# scrgadm -a -g iws-resource-group-1 -y Maximum_primaries=2 \
-y Desired_primaries=2 -y RG_dependencies=sa-resource-group-1

(Register the iPlanet resource type.)
# scrgadm -a -t SUNW.iws

(Add an insecure iPlanet instance with default load balancing.)
# scrgadm -a -j iplanet-insecure-1 -g iws-resource-group-1 -t SUNW.iws \
-x Confdir_List=/opt/iplanet/https-iplanet-insecure-1 \
-y Scalable=True -y Network_resources_used=schost-1 -y Port_list=80/tcp

(Add a secure iPlanet instance with sticky IP load balancing.)
# scrgadm -a -j iplanet-secure-1 -g iws-resource-group-1 -t SUNW.iws \
-x Confdir_List=/opt/iplanet/https-iplanet-secure-1 \
-y Scalable=True -y Network_resources_used=schost-1 \
-y Port_list=443/tcp -y Load_balancing_policy=LB_STICKY \
-y Load_balancing_weight=40@1,60@2

(Bring the failover resource group online.)
# scswitch -Z -g sa-resource-group-1

(Bring the scalable resource group online.)
# scswitch -Z -g iws-resource-group-1
```

Example – Registering Failover Sun Cluster HA for iPlanet Web Server

The following example shows how to register a failover iPlanet service on a two-node cluster.

```
Cluster Information
Node names: phys-schost-1, phys-schost-2
Logical hostname: schost-1
Resource group: resource-group-1 (for all resources)
Resources: schost-1 (logical hostname), iplanet-insecure-1 (insecure iPlanet
    application resource), iplanet-secure-1 (secure iPlanet application
    resource)

(Add the resource group to contain all resources.)
# scrgadm -a -g resource-group-1

(Add the logical hostname resource to the resource group.)
# scrgadm -a -L -g resource-group-1 -l schost-1

(Register the iPlanet resource type.)
# scrgadm -a -t SUNW.iws

(Add an insecure iPlanet application resource instance.)
# scrgadm -a -j iplanet-insecure-1 -g resource-group-1 -t SUNW.iws \
-x Confdir_list=/opt/iplanet/conf -y Scalable=False \
-y Network_resources_used=schost-1 -y Port_list=80/tcp\

(Add a secure iPlanet application resource instance.)
# scrgadm -a -j iplanet-secure-1 -g resource-group-1 -t SUNW.iws \
-x Confdir_List=/opt/iplanet/https-iplanet-secure-1 -y Scalable=False \
-y Network_resources_used=schost-1 -y Port_list=443/tcp \

(Bring the failover resource group online.)
# scswitch -Z -g resource-group-1
```

Where to Go From Here

To configure the `SUNW.HAStorage` resource type, see "How to Configure `SUNW.HAStorage` Resource Type" on page 59.

▼ How to Configure SUNW.HAStorage Resource Type

The SUNW.HAStorage resource type synchronizes actions between HA storage and the data service. The Sun Cluster HA for iPlanet Web Server data service is scalable, and therefore you should configure the SUNW.HAStorage resource type.

See the SUNW.HAStorage(5) man page and "Relationship Between Resource Groups and Disk Device Groups" on page 4 for background information. See "How to Set Up SUNW.HAStorage Resource Type for New Resources" on page 239 for the procedure.

Configuring Sun Cluster HA for iPlanet Web Server Extension Properties

This section describes the Sun Cluster HA for iPlanet Web Server extension properties. For failover, the data service enforces that the size of Confdir_list is one. If you want multiple configuration files (instances), make multiple failover resources, each with one Confdir_list entry.

Typically, you use the command line scrgadm -x *parameter=value* to configure extension properties when you create the iPlanet Web Server resource. You can also use the procedures described in Chapter 11 to configure them later. See Appendix A for details on all Sun Cluster properties.

TABLE 3-2 describes extension properties that you can configure for the iPlanet server. The only required extension property for creating an iPlanet server resource is the Confdir_list property. You can update some extension properties dynamically. You can update others, however, only when you create the resource. The Tunable column of the following table indicates when you can update each property

TABLE 3-2 Sun Cluster HA for iPlanet Web Server Extension Properties

Name/Data Type	Default	Range	Tunable	Description
Confdir_list (string array)	None	None	At creation	A pointer to the server root directory for a particular iPlanet Web server instance. If the Netscape Directory Server is in secure mode, the path name must contain a file named keypass, which contains the secure key password needed to start this instance.
Monitor_retry_count (integer)	4	0 – 2,147,483,641 -1 indicates an infinite number of retry attempts.	Any time	The number of times the process monitor facility (PMF) restarts the fault monitor during the time window that the Monitor_retry_interval property specifies. Note that this property refers to restarts of the fault monitor itself rather than to the resource. The system-defined properties Retry_interval and Retry_count control restarts of the resource.
Monitor_retry_interval (integer)	2	0 – 2,147,483,641 -1 indicates an infinite retry interval.	Any time	The time (in minutes) over which failures of the fault monitor are counted. If the number of times the fault monitor fails exceeds the value specified in the extension property Monitor_retry_count within this period, the PMF does not restart the fault monitor.
Probe_timeout (integer)	30	0 – 2,147,483,641	Any time	The time-out value (in seconds) that the fault monitor uses to probe an iPlanet Web Server instance.

Sun Cluster HA for iPlanet Web Server Fault Monitor

The probe for the Sun Cluster HA for iPlanet Web Server (iWS) data service uses a request to the server to query the health of that server. Before the probe actually queries the server, a check is made to confirm that network resources are configured for this Web server resource. If no network resources are configured, an error message (`No network resources found for resource`) is logged, and the probe exits with failure.

The probe must address the following two configurations of iWS.

- the secure instance
- the insecure instance

If the Web server is in secure mode and if the probe cannot get the secure ports from the configuration file, an error message (`Unable to parse configuration file`) is logged, and the probe exits with failure. The secure and insecure instance probes involve common steps.

The probe uses the time-out value that the resource property `Probe_timeout` specifies to limit the time spent trying to successfully probe iWS. See Appendix A for details on this resource property.

The `Network_resources_used` resource-property setting on the iWS resource determines the set of IP addresses that the Web server uses. The `Port_list` resource-property setting determines the list of port numbers that iWS uses. The fault monitor assumes that the Web server is listening on all combinations of IP and port. If you customize your Web server configuration to listen on different port numbers (in addition to port `80`), ensure that your resultant configuration (`magnus.conf`) file contains all possible combinations of IP addresses and ports. The fault monitor attempts to probe all such combinations and might fail if the Web server is not listening on a particular IP address and port combination.

The probe executes the following steps.

1. The probe uses the specified IP address and port combination to connect to the Web server. If the connection is unsuccessful, the probe concludes that a complete failure has occurred. The probe then records the failure and takes appropriate action.

2. If the probe successfully connects, the probe checks if the Web server is run in a secure mode. If so, the probe disconnects and returns with a success status. No further checks are performed for a secure iWS server.

However, if the Web server is running in insecure mode, the probe sends an HTTP 1.0 HEAD request to the Web server and waits for the response. The request can be unsuccessful for various reasons, including heavy network traffic, heavy system load, and misconfiguration.

Misconfiguration can occur when the Web server is not configured to listen on all IP address and port combinations that are being probed. The Web server should service every port for every IP address specified for this resource.

Misconfigurations can also result if the `Network_resources_used` and `Port_list` resource properties are not set correctly while you create the resource.

If the reply to the query is not received within the `Probe_timeout` resource proper limit, the probe considers this a failure of the Sun Cluster HA for iPlanet Web Server data service. The failure is recorded in the probe's history.

A probe failure can be a complete or partial failure. The following probe failures are considered complete failures.

- Failure to connect to the server, as the following error message flags, with `%s` indicating the host name and `%d` the port number.

```
Failed to connect to %s port %d
```

- Running out of time (exceeding the resource-property timeout `Probe_timeout`) after trying to connect to the server.
- Failure to successfully send the probe string to the server, as the following error message flags, with the first `%s` indicating the host name and `%d` the port number. The second `%s` indicates further details about the error.

```
Failed to communicate with server %s port %d: %s
```

Two such partial failures within the resource-property interval `Retry_interval` are accumulated by the monitor and are counted as one.

The following probe failures are considered partial failures.

- Running out of time (exceeding the resource-property timeout `Probe_timeout`) while trying to read the reply from the server to the probe's query.
- Failing to read data from the server for other reasons, as the following error message flags, with the first `%s` indicating the host name and `%d` the port number. The second `%s` indicates further details about the error.

```
Failed to communicate with server %s port %d: %s
```

3. Based on the history of failures, a failure can cause either a local restart or a failover of the data service. This action is further described in "Health Checks of the Data Service" on page 11.

CHAPTER 4

Installing and Configuring Sun Cluster HA for Netscape Directory Server

This chapter describes the procedures for installing and configuring the Sun Cluster HA for Netscape Directory Server data service. This data service was formerly known as Sun Cluster HA for Netscape LDAP. Some error messages from the application might still use the name Netscape LDAP, but they refer to Netscape Directory Server (NDS).

This chapter contains the following procedures.

- "How to Configure and Activate Network Resources" on page 68
- "How to Install Netscape Directory Server" on page 71
- "How to Configure Netscape Directory Server" on page 71
- "How to Install Sun Cluster HA for Netscape Directory Server Packages" on page 72
- "How to Complete the Sun Cluster HA for Netscape Directory Server Configuration" on page 74
- "How to Configure `SUNW.HAStorage` Resource Type" on page 77

You must configure the Sun Cluster HA for Netscape Directory Server data service as a failover service. See Chapter 1 and the *Sun Cluster 3.0 U1 Concepts* document for general information about data services, resource groups, resources, and other related topics.

Note – You can use SunPlex Manager to install and configure this data service. See the SunPlex Manager online help for details.

Planning the Installation and Configuration

Use this section in conjunction with the worksheets in the *Sun Cluster 3.0 U1 Release Notes* as a checklist before installation and configuration.

Consider the following points prior to starting your installation.

- Where will the server root reside?

  You can store files and data that do not change on the local file system of each cluster node. However, place dynamic data on the cluster file system so that you can view or update the data from any cluster node.

- If you plan to use multiple NDS instances on a node, you must set the `listenhost` directive in the `slapd.conf` file with the appropriate network resource as the IP address (a logical hostname). This setting is necessary because the default NDS behavior is for the instance to bind to all IP addresses on the node.

  For example, to set up a particular instance to use the logical hostname `nds-1`, add the following entry to the instance's `slapd.conf` file: `listenhost nds-1`. This setting causes the instance to bind to the logical hostname `nds-1` only, rather than to all the IP addresses on the node.

- The LDAP administrative server is case-sensitive in its consideration of hostnames. Therefore, all hostnames specified in the LDAP configuration for the administrative server must match their case with the LDAP specification in the name service in use on the cluster node. If DNS is the name service in use, this case-matching is particularly important because the DNS domain name must also match the host-name specification in the LDAP configuration.

  Be sure that the case of the fully qualified domain name of the machine for LDAP matches the case of the domain name that the resolver returns. For example, if the DNS resolver returns `Eng.Sun.COM` as the domain name (note the mixed case), you must spell that name exactly the same way when you configure the LDAP administrative server.

Installing and Configuring Sun Cluster HA for Netscape Directory Server

The following table lists the sections that describe the installation and configuration tasks.

TABLE 4-1 Task Map: Installing and Configuring Sun Cluster HA for Netscape Directory Server

Task	For Instructions, Go To
Configure and activate network resources	"How to Configure and Activate Network Resources" on page 68
Install and configure Netscape Directory Server	"Installing and Configuring Netscape Directory Server" on page 70
Install the Sun Cluster HA for Netscape Directory Server-data service packages	"Installing Sun Cluster HA for Netscape Directory Server Packages" on page 72
Configure application resources and start the Sun Cluster HA for Netscape Directory Server data service	"Completing the Sun Cluster HA for Netscape Directory Server Configuration" on page 73
Configure resource extension properties	"Configuring Sun Cluster HA for Netscape Directory Server Extension Properties" on page 77

Note – If you are running multiple data services in your Sun Cluster configuration, you can set up the data services in any order, with the following exception. If you use the Sun Cluster HA for DNS data service, you must set up the Sun Cluster HA for DNS data service before you set up Netscape Directory Server. See Chapter 6 for details.

DNS software is included in the Solaris operating environment. If the cluster is to obtain the DNS service from another server, configure the cluster to be a DNS client first.

Note – After installation, use only the cluster administration command `scswitch`(1M) to manually start and stop Netscape Directory Server. See the man page for details. After Netscape Directory Server is started, the Sun Cluster software controls it.

Configuring and Activating Network Resources

Before you install and configure Netscape Directory Server, set up the network resources that the server will attempt to use after the server has been installed and configured. To configure and activate the network resources, use the following command-line procedure.

▼ How to Configure and Activate Network Resources

To perform this procedure, you need the following information about your configuration.

- The names of the cluster nodes that can master the data service.
- The logical hostname that clients use to access the Sun Cluster HA for Netscape Directory Server data service. Normally, you set up this hostname when you install the cluster. See the section in the *Sun Cluster 3.0 U1 Installation Guide* on how to set up logical hostnames for details.

Note – Perform this procedure on any cluster member.

---

1. **Become superuser on a cluster member.**
2. **Verify that all network addresses that you use have been added to your name-service database.**

   You should have performed this verification during the Sun Cluster installation. See the planning chapter in the *Sun Cluster 3.0 U1 Installation Guide* for details.

Note – To avoid any failures because of name-service lookup, ensure that all logical hostnames and shared addresses are present in the `/etc/hosts` file on all cluster nodes. Configure name-service mapping in the `/etc/nsswitch.conf` file on the servers to first check the local files before trying to access NIS, NIS+, or DNS.

---

3. **Create a failover resource group to hold the network and application resources.**

```
# scrgadm -a -g resource-group [-h nodelist]
```

`-g` *resource-group*
Specifies the name of the resource group. This name can be your choice.

`-h` *nodelist*
Specifies an optional comma-separated list of physical node names or IDs that identify potential masters. The order here determines the order in which the nodes are considered as primary during failover.

Note – Use the `-h` option to specify the order of the node list. If all the nodes in the cluster are potential masters, you need not use the `-h` option.

---

4. **Add logical-hostname resources to the resource group.**

```
# scrgadm -a -L -g resource-group -l hostname, …[-n netiflist]
```

`-L`
Specifies that a logical-hostname resource is being added.

`-g` *resource-group*
Specifies the name of the resource group.

`-l` *hostname, …*
Specifies a comma-separated list of logical hostnames.

`-n` *netiflist*
Specifies an optional comma-separated list that identifies the NAFO groups on each node. All the nodes in *nodelist* of the resource group must be represented in *netiflist*. If you do not specify this option, `scrgadm`(1M) attempts to discover a net adapter on the subnet that the *hostname* list identifies for each node in *nodelist*.

5. **Verify that all logical hostnames that you use have been added to your name-service database.**

   You should have performed this verification during the Sun Cluster installation. See the planning chapter in the *Sun Cluster 3.0 U1 Installation Guide* for details.

6. **Run the `scswitch` command to enable the resource group and bring the resource group online.**

```
# scswitch -Z -g resource-group
```

`-Z` Moves the resource group to the managed state, and brings the resource group online.

`-g` *resource-group* Specifies the name of the resource group.

Where to Go From Here

After you configure and activate the network resources, go to "Installing and Configuring Netscape Directory Server" on page 70.

Installing and Configuring Netscape Directory Server

The Sun Cluster HA for Netscape Directory Server data service is the Netscape Directory Server that uses Netscape Lightweight Directory Access Protocol (LDAP) and runs under the control of the Sun Cluster software. This section describes the steps to install Netscape Directory Server (using the `setup` command) and enable Netscape Directory Server to run as the Sun Cluster HA for Netscape Directory Server data service.

Netscape Directory Server requires some variation from the default installation parameters. When you install and configure Netscape Directory Server, consider the following points.

- For the service to fail over correctly, when prompted for the name of Netscape Directory Server, instead of specifying a physical machine, you must specify a logical hostname (IP address) that can fail over between nodes. This requirement means that before you begin the installation, you must set up the logical hostname in your name services. You normally perform this step, which the *Sun Cluster 3.0 U1 Installation Guide* describes, as part of the Sun Cluster installation.
- Do not use the default server root disk path when prompted. Place your files on the cluster file system.

Note – Do not remove or relocate any of the installed files or directories that the Netscape Directory Server installation places on the cluster file system. For example, do not relocate any of the client binaries, such as `ldapsearch`, that are installed along with the rest of the Netscape Directory Server software.

▼ How to Install Netscape Directory Server

This procedure describes the interaction with the Netscape `setup` command. Only the sections that are specific to the Sun Cluster HA for Netscape Directory Server data service are included here. For the other sections, choose or change the default values as appropriate. This procedure includes only basic steps. See the Netscape LDAP documentation for details.

1. **Become superuser on a cluster member.**

2. **Run the `setup` command from the install directory on the Netscape CD.**

3. **From `setup`, choose the menu items to install a Netscape Server with a Custom Installation.**

   Supply the logical hostname when the `setup` command prompts you for the full server name.

4. **For the install location, select a location on the global file system, for example, `/global/nsldap`.**

   Supply the logical hostname when the `setup` command prompts you for the full server name. This step is required for failover to work correctly.

Note – The logical host that you specify must be online on the node from which you run the Netscape Directory Server installation. This state is necessary because at the end of the Netscape Directory Server installation, Netscape Directory Server automatically starts and will fail if the logical host is offline on that node.

---

5. **Select the logical hostname along with your domain for the computer name, for example, `schost-1.eng.sun.com`.**

6. **When prompted for the IP address to be used as the LDAP Administrative Server, specify an IP address for one of the cluster nodes.**

   As part of the installation, you set up an LDAP Administrative Server. The IP address that you specify for this server must be that of a physical cluster node, not the name of the logical host that will fail over.

▼ How to Configure Netscape Directory Server

- Use the Netscape Administration Server to configure and test Netscape Directory Server.

  See your Netscape documentation for details.

After completing the configuration, Netscape Directory Server starts automatically. Before you proceed to the next part of the installation and configuration process, you must use stop-slapd to stop the server.

Where to Go From Here

If you have not installed the data-service packages for Netscape Directory Server from the Sun Cluster Agents CD, go to "Installing Sun Cluster HA for Netscape Directory Server Packages" on page 72. If you have installed the packages, go to "Completing the Sun Cluster HA for Netscape Directory Server Configuration" on page 73.

Installing Sun Cluster HA for Netscape Directory Server Packages

You can use the scinstall(1M) utility to install SUNWscnsl, the Sun Cluster HA for Netscape Directory Server data-service package, on a cluster. Do not use the -s option to non-interactive scinstall to install all data service packages on the CD.

If you installed the data-service packages during your initial Sun Cluster installation, proceed to "Completing the Sun Cluster HA for Netscape Directory Server Configuration" on page 73. Otherwise, use the following procedure to install the SUNWscnsl package now.

▼ How to Install Sun Cluster HA for Netscape Directory Server Packages

You need the Sun Cluster Agents CD to complete this procedure. Run this procedure on all cluster members that can master the Sun Cluster HA for Netscape Directory Server data service.

1. **Load the Agents CD into the CD-ROM drive.**
2. **Run the scinstall utility with no options.**

   This step starts the scinstall utility in interactive mode.
3. **Select the Add Support for New Data Service to This Cluster Node menu option.**

   This option enables you to load software for any data services that exist on the CD.

4. **Exit the `scinstall` utility.**

5. **Unload the CD from the drive.**

Where to Go From Here

See "Completing the Sun Cluster HA for Netscape Directory Server Configuration" on page 73 to register the Sun Cluster HA for Netscape Directory Server data service and to configure the cluster for the data service.

Completing the Sun Cluster HA for Netscape Directory Server Configuration

This procedure describes how to use the `scrgadm` command to register and configure the Sun Cluster HA for Netscape Directory Server data service.

Note – Other options also enable you to register and configure the data service. See "Tools for Data-Service Resource Administration" on page 8 for details about these options.

To perform this procedure, you need the following information about your configuration.

- The name of the resource type for the Sun Cluster HA for Netscape Directory Server data service. This name is `SUNW.nsldap`.
- The names of the cluster nodes that can master the data service.
- The logical hostname that clients use to access the Sun Cluster HA for Netscape Directory Server data service. Normally, you set up this logical hostname when you install the cluster. See the section on how to set up logical hostnames in the *Sun Cluster 3.0 U1 Installation Guide* for details.
- The path to the Netscape Directory Server application binaries that are the resources for the Sun Cluster HA for Netscape Directory Server data service. You can install the binaries on the local disks or the cluster file system. See Chapter 1 for a discussion of the advantages and disadvantages of each location.
- The port where Netscape Directory Server listens. For non-secure instances, the `Port_list` standard resource property for the Netscape Directory Server resource defaults to `389/tcp`, and the value for the secure port is `636/tcp`. If

you set the port to a number other than `389`, you must specify that value when you configure the `Port_list` property. See Chapter 11 for instructions on how to set resource properties.

Note – Perform this procedure on any cluster member.

How to Complete the Sun Cluster HA for Netscape Directory Server Configuration

Perform the following steps to complete your configuration.

1. **Become superuser on a cluster member.**
2. **Register the resource type for the data service.**

```
# scrgadm -a -t SUNW.nsldap
```

`-a`	Adds the data-service resource type.
`-t SUNW.nsldap`	Specifies the predefined resource-type name.

3. **Add the Netscape Directory Server application resource t the failover resource group that you created for your network resources.**

   The resource group that contains the application resources is the same resource group that you created for your network resources in "How to Configure and Activate Network Resources" on page 68.

```
# scrgadm -a -j resource -g resource-group \
-t resource-type [-y Network_resources_used=network-resource, …] \
-y Port_list=port-number/protocol -x Confdir_list=pathname
```

`-j` *resource*	Specifies the LDAP application resource name.
`-y Network_resources_used=`*network-resource*	Specifies a comma-separated list of network resources (logical hostnames or shared addresses) in *resource-group*, which the LDAP application resource must use.

`-t` *resource-type*
Specifies the resource type to which the resource belongs, for example, `SUNW.iws`.

`-y Port_list=` *port-number/protocol*
Specifies a port number and the protocol to be used, for example, `389/tcp`. The `Port_list` property must have exactly one entry.

`-x Confdir_list=` *pathname*
Specifies a path for your LDAP configuration directory. The `Confdir_list` extension property is required. The `Confdir_list` property must have exactly one entry.

4. **Enable the resource and its monitor.**

```
# scswitch -e -j resource
```

`-e`
Enables the resource and its monitor.

`-g` *resource*
Specifies the name of the application resource being enabled.

Example–Registering and Configuring Sun Cluster HA for Netscape Directory Server

This example shows how to register the Sun Cluster HA for Netscape Directory Server data service.

```
Cluster Information
Node names: phys-schost-1, phys-schost-2
Logical hostname: schost-1
Resource group: resource-group-1 (for all resources)
Resources: schost-1 (logical hostname),
    nsldap-1 (LDAP application resource)

(Create a failover resource group.)
# scrgadm -a -g resource-group-1 -h phys-schost-1,phys-schost-2

(Add a logical hostname resource to the resource group.)
# scrgadm -a -L -g resource-group-1 -l schost-1

(Bring the resource group online.)
# scswitch -Z -g resource-group-1

(Install and configure Netscape Directory Server.)

(Stop the LDAP server.)

(Register the SUNW.nsldap resource type.)
# scrgadm -a -t SUNW.nsldap

(Create an LDAP resource and add it to the resource group.)
# scrgadm -a -j nsldap-1 -g resource-group-1 \
-t SUNW.nsldap -y Network_resources_used=schost-1 \
-y Port_list=389/tcp \
-x Confdir_list=/global/nsldap/slapd-schost-1

(Enable the application resources.)
# scswitch -e -j nsldap-1
```

▼ How to Configure SUNW.HAStorage Resource Type

The SUNW.HAStorage resource type synchronizes actions between HA storage and the data service. The Sun Cluster HA for Netscape Directory Server data service is not disk-intensive and not scalable, and therefore configuring the SUNW.HAStorage resource type is optional.

See the SUNW.HAStorage(5) man page and "Relationship Between Resource Groups and Disk Device Groups" on page 4 for background details. See "How to Set Up SUNW.HAStorage Resource Type for New Resources" on page 239 for information about the procedure.

Configuring Sun Cluster HA for Netscape Directory Server Extension Properties

This section describes how to configure the Sun Cluster HA for Netscape Directory Server extension properties. Typically, you use the command line scrgadm -x *parameter=value* to configure extension properties when you create the Netscape Directory Server resource. You can also use the procedures that Chapter 11 describes to configure them later.

See Appendix A for details on all Sun Cluster properties.

TABLE 4-2 describes the extension properties that you can configure for Netscape Directory Server. The only required extension property for creating a Netscape Directory Server resource is the Confdir_list property, which specifies a directory in which the Netscape Directory Server configuration files reside. You can update some extension properties dynamically. You can update others, however, only when you create the resource. The Tunable column of the following table indicates when you can update each property.

TABLE 4-2 Sun Cluster HA for Netscape Directory Server Extension Properties

Name/Data Type	Default	Range	Tunable	Description
`Confdir_list` (string array)	None	None	At creation	A path name that points to the server root, including the `slapd`-*hostname* subdirectory where the `start-slapd` and `stop-slapd` scripts reside. The Sun Cluster HA for Netscape Directory Server data service requires this extension property, and the property must have one entry only. If Netscape Directory Server is in secure mode, then the path name must also contain a file named `keypass`, which contains the secure key password needed to start this instance.
`Monitor_retry_count` (integer)	4	`0 – 2,147,483,641` `–1` indicates an infinite number of retry attempts.	Any time	The number of times the process monitor facility (PMF) restarts the fault monitor during the time window that the `Monitor_retry_interval` property specifies. Note that this property refers to restarts of the fault monitor itself rather than to the resource. The system-defined properties `Retry_interval` and `Retry_count` control restarts of the resource.
`Monitor_retry_interval` (integer)	2	`0 – 2,147,483,641` `–1` indicates an infinite retry interval.	Any time	The time (in minutes) over which failures of the fault monitor are counted. If the number of times the fault monitor fails exceeds the value specified in the extension property `Monitor_retry_count` within this period, the PMF cannot restart the fault monitor.
`Probe_timeout` (integer)	30	`0 – 2,147,483,641`	Any time	The time-out value (in seconds) that the fault monitor uses to probe a Netscape Directory Server instance.

Sun Cluster HA for Netscape Directory Server Fault Monitor

The probe for the Sun Cluster HA for Netscape Directory Server data service accesses particular IP addresses and port numbers. The IP addresses are from network resources that the `Network_resources_used` property lists. The `Port_list` resource property lists the port. See Appendix A for descriptions of these properties.

The fault monitor determines whether the Sun Cluster HA for Netscape Directory Server instance is secure or non-secure. The monitor probes secure and non-secure directory servers differently. If the keyword "security" is not found in the configuration file (`slapd.conf`), or the setting `security off` is found, then the instance is determined to be non-secure. Otherwise, the instance is determined to be secure.

The probe for a secure instance consists of a simple TCP connect. If the connect succeeds, the probe is successful. Secure connect failure or timeout is interpreted as complete failure.

The probe for an insecure instance depends on running the `ldapsearch` executable provided with the Sun Cluster HA for Netscape Directory Server data service. The search filter that is used is intended to always find something. The probe detects partial and complete failures. The following conditions are considered partial failures. All other conditions are interpreted as complete failures.

- `Probe_timeout` duration is exceeded while the set of IP addresses is probed for the port. The following list identifies potential causes of this problem.
- System load.
- Network-traffic load.
- Directory-server load.
- `Probe_timeout` is set too low for the typical load or the number of directory-server instances (that is, IP address and port combinations) that are being monitored.
- A problem other than timeout occurs while `ldapsearch` is invoked. Note that this scenario does not apply to the situation where `ldapsearch` is invoked successfully but returns an error.

CHAPTER 5

Installing and Configuring Sun Cluster HA for Apache

This chapter describes the steps to install and configure the Sun Cluster HA for Apache data service on your Sun Cluster servers.

This chapter contains the following procedures.

- "How to Install and Configure the Apache Application Software from the Apache Web Site" on page 90
- "How to Install Sun Cluster HA for Apache Packages" on page 92
- "How to Register and Configure Sun Cluster HA for Apache" on page 93
- "How to Configure `SUNW.HAStorage` Resource Type" on page 101
- "How to Verify Data-Service Installation and Configuration" on page 102

You can configure the Sun Cluster HA for Apache data service as either a failover service or a scalable service. See Chapter 1 and the *Sun Cluster 3.0 U1 Concepts* document for an overview of failover and scalable data services.

Note – You can use SunPlex Manager to install and configure this data service. See the SunPlex Manager online help for details.

Planning the Installation and Configuration

Before you install the Sun Cluster HA for Apache data service, update the following information in the Apache configuration file `httpd.conf`.

- **The `ServerName` directive that contains the hostname –** For the Sun Cluster HA for Apache data service to be highly available, you must set this directive to the name of the network address (logical hostname or shared address) that is used to access the server. The logical hostname or shared address should have been set up when the cluster was installed. If not, see the *Sun Cluster 3.0 U1 Installation Guide* for information on how to set up logical hostnames and shared addresses, and set one up now.
- **The `BindAddress` directive, which you must set to the logical host or shared address –** Because you can configure the Apache software to bind to `INADDR_ANY`, if you plan to run multiple instances of the Apache data service or multiple data services on the same node, each instance must bind to a unique network resource and port number.
- **The `ServerType` directive –** This directive must be set to `standalone`, the default.
- **The `ServerRoot` directive that specifies the top of the directory tree under which the server's `conf` and `log` subdirectories are typically located –** This directive has no default.

  If you use a cluster file system as the location for the server root, you need only install the Apache software on that single file system to make the software accessible to all the nodes that can run the data service. See the discussion on placement of the binary files in "Determining the Location of the Application Binaries" on page 3.

You might have multiple instances that use a single Apache binary. The location of the configuration file is specified according to the `Confdir_list` resource property. The following example shows the location of the configuration file.

```
(Location of the Apache binaries – also the value of the Bin_dir
property)
/global/apache/bin

(Location of configuration directories – Confdir_list property)
/global/websites/dev/conf
/global/websites/sqa/conf

(Location of httpd.conf files)
/global/websites/dev/conf/httpd.conf
/global/websites/sqa/conf/httpd.conf
```

Run the following commands to start up the instances by hand, as you might when you verify your setup. Also, when instructed by the Resource Group Manager (RGM), the data service in effect issues the following commands to start the instances.

```
# /global/apache/bin/httpd \
-f /global/websites/dev/conf/httpd.conf
# /global/apache/bin/httpd \
-f /global/websites/sqa/conf/httpd.conf
```

- **The `DocumentRoot` directive that specifies the location of the documentation root directory –** This directive is a pointer to a location on the cluster file system, where the HTML documents are installed.
- **The `ScriptAlias` directive that contains the location on a cluster file system of the `cgi-bin` directory –** This directive is a pointer to a location on the cluster file system, where the `cgi-bin` files are installed.

Note – You must follow certain conventions when you configure URL mappings for the Web server. For example, when setting the CGI directory, locate the CGI directory on the cluster file system to preserve availability. For example, you might map your CGI directory to /global/*diskgroup*/*ServerRoot*/cgi-bin, where *diskgroup* is the disk device group that contains the Apache software.

In situations where the CGI programs access "back-end" servers, such as an RDBMS, ensure that the Sun Cluster software controls the "back-end" server. If the server is an RDBMS that the Sun Cluster software supports, use one of the highly available RDBMS packages. Alternatively, you can use the APIs that the *Sun Cluster 3.0 U1 Data Services Developers' Guide* documents to put the server under Sun Cluster control.

- **If you use a lock file –** Set the value of the LockFile directive in your httpd.conf file to a local file.
- **Use a PidFile directive –** Point this directive to a local file, as in the following example.

```
PidFile /usr/local/apache/log/httpd.pid
```

- **The Port directive setting that the server port or ports access –** The defaults are set in each node's httpd.conf file. The Port_list resource property must include all the ports specified in the httpd.conf files.

  The Port_list property assumes that the Web server serves all combinations of ports and IP addresses from the network resources as defined in the Network_resources_used property.

```
Port_list="80/tcp,443/tcp,8080/tcp"
```

The preceding Port_list configuration, for example, probes the following IP-port combinations.

Host	Port	Protocol
node1	80	tcp
node1	443	tcp
node1	8080	tcp
node2	80	tcp
node2	443	tcp
node2	8080	tcp

However, if *node1* serves ports 80 and 443 only and *node2* serves ports 80 and 8080 only, you can configure the Port_list property for Apache as follows.

```
Port_list=node1/80/tcp,node1/443/tcp,node2/80/tcp,node2/8080/tcp
```

Consider the following rules.

- You must specify hostnames or IP addresses (not network resource names) for *node1* and *node2*.
- If Apache serves *nodeN*/*port* for every *nodeN* in the Network_resources_used property, you can use a short form to replace the combination of *node1*/*port1*, *node2*/*port2*, and so on. See the following examples.

  Example One

```
Port_list=”80/tcp,node1/443/tcp,node2/8080/tcp”
Network_resources_used=node1,node2
```

This example probes the following IP-port combinations.

Host	Port	Protocol
node1	80	tcp
node1	443	tcp
node2	80	tcp
node2	8080	tcp

Example Two

```
Port_list="node1/80/tcp,node2/80/tcp"
Network_resources_used=net-1,net-2
#net-1 contains node1.
#net-2 contains node2 and node3.
```

This example probes the following IP-port combinations.

Host	Port	Protocol
node1	80	tcp
node2	80	tcp

- All hostnames (IP addresses) that the `Port_list` property specifies must not belong to a network resource that is specified in any other scalable resource's `Network_resources_used` property. Otherwise, as soon as a scalable service detects that another scalable resource already uses an IP address, creation of the Apache resource fails.

Note – If you are running the Sun Cluster HA for Apache data service and another HTTP server, configure the HTTP servers to listen on different ports. Otherwise, a port conflict can occur between the two servers.

To register and configure the Sun Cluster HA for Apache data service, you must consider or provide information on the following points.

- Decide whether to run the Sun Cluster HA for Apache data service as a failover or a scalable service.
- Decide which fault monitoring resource properties (such as the `Thorough_probe_interval` or `Probe_timeout` properties) to set. In most cases, the default values suffice. See "Configuring Sun Cluster HA for Apache Extension Properties" on page 102 for information about these properties.
- Provide the name of the resource type for the Sun Cluster HA for Apache data service. This name is `SUNW.apache`.
- Provide the names of the cluster nodes that will master the data service.
- Provide the logical hostname (failover services) or shared address (scalable services) that clients use to access the data service. You typically set up this IP address when you install the cluster. See the *Sun Cluster 3.0 U1 Installation Guide* for details on how to set up network addresses.

- Provide the path to the application binaries. You can install the binaries on the local disks or on the cluster file system. See “Determining the Location of the Application Binaries” on page 3 for a discussion of the advantages and disadvantages of each location.
- Provide the path to the `conf` directory.
- Exercise caution when changing the `Load_balancing_weights` property for an online scalable service that has the `Load_balancing_policy` property set to `LB_STICKY` or `LB_STICKY_WILD`. Changing these properties while the service is online can cause existing client affinities to be reset, hence a different node might service a subsequent client request even if another cluster member previously serviced the client.

  Similarly, when a new instance of the service is started on a cluster, existing client affinities might be reset.

Note – If a scalable proxy is serving a scalable Web resource with the `LB_STICKY` policy, you must also set up an `LB_STICKY` policy for the proxy.

- Determine the entries for the `Confdir_list` and `Port_list` properties. For failover services, the `Confdir_list` property can have only one entry, but the `Port_list` property can have multiple entries. For scalable services, both properties can have multiple entries. See “How to Register and Configure Sun Cluster HA for Apache” on page 93 for details.

Installing and Configuring Sun Cluster HA for Apache

TABLE 5-1 lists the sections that describe the installation and configuration tasks.

TABLE 5-1 Task Map: Installing and Configuring Sun Cluster HA for Apache

Task	For Instructions, Go To
Install the Apache software	"How to Install and Configure the Apache Application Software from the Apache Web Site" on page 90
Install the Sun Cluster HA for Apache data-service packages	"How to Install Sun Cluster HA for Apache Packages" on page 92
Configure and start the Sun Cluster HA for Apache data service	"How to Register and Configure Sun Cluster HA for Apache" on page 93
Configure resource extension properties	"Configuring Sun Cluster HA for Apache Extension Properties" on page 102
View fault-monitor information	"Sun Cluster HA for Apache Fault Monitor" on page 104

Installing and Configuring Apache

This section describes the steps to install the Apache server—either from the Solaris 8 operating-environment CD-ROM or from the Apache Web site—and to enable the server to run as the Sun Cluster HA for Apache data service.

The Sun Cluster HA for Apache data service works with the Apache software configured as either a Web server or a proxy server.

See Apache documentation at `http://www.apache.org` for standard installation instructions. See the *Sun Cluster 3.0 U1 Release Notes* for a list of Apache releases supported for use with the Sun Cluster software.

▼ How to Install and Configure the Apache Application Software from the Solaris 8 CD-ROM

The Apache binaries are included in three packages—`SUNWapchr`, `SUNWapchu`, and `SUNWapchd`—which form the `SUNWCapache` package metacluster. You must install the `SUNWapchr` package before you install the `SUNWapchu` package.

Place the Web server binaries on the local file system on each of your cluster nodes or on a cluster file system.

1. **Run the `pkginfo(1)` command to determine if the Apache packages `SUNWapchr`, `SUNWapchu`, and `SUNWapchd` have been installed.**

   If not, install as follows.

```
# pkgadd -d Solaris 8 Product directory SUNWapchr SUNWapchu SUNWapchd
...
Installing Apache Web Server (root) as SUNWapchr
...
[ verifying class initd ]
/etc/rc0.d/K16apache linked pathname
/etc/rc1.d/K16apache linked pathname
/etc/rc2.d/K16apache linked pathname
/etc/rc3.d/S50apache linked pathname
/etc/rcS.d/K16apache linked pathname
...
```

2. **Disable the `START` and `STOP` run control scripts that were just installed as part of the `SUNWapchr` package.**

   This step is necessary because the Sun Cluster HA for Apache data service starts and stops the Apache application after you have configured the data service. Perform the following steps.

   1. List the Apache run control scripts.

   2. Rename the Apache run control scripts.

   3. Verify that all the Apache-related scripts have been renamed.

Note – The following example changes the first letter in the name of the run control script from uppercase to lowercase. However, you can rename the scripts to be consistent with your normal administration practices.

```
# ls -1 /etc/rc?.d/*apache
/etc/rc0.d/K16apache
/etc/rc1.d/K16apache
/etc/rc2.d/K16apache
/etc/rc3.d/S50apache
/etc/rcS.d/K16apache

# mv /etc/rc0.d/K16apache  /etc/rc0.d/k16apache
# mv /etc/rc1.d/K16apache  /etc/rc1.d/k16apache
# mv /etc/rc2.d/K16apache  /etc/rc2.d/k16apachc
# mv /etc/rc3.d/S50apache  /etc/rc3.d/s50apache
# mv /etc/rcS.d/K16apache  /etc/rcS.d/k16apache

# ls -1 /etc/rc?.d/*apache
/etc/rc0.d/k16apache
/etc/rc1.d/k16apache
/etc/rc2.d/k16apache
/etc/rc3.d/s50apache
/etc/rcS.d/k16apache
```

▼ How to Install and Configure the Apache Application Software from the Apache Web Site

1. **Become superuser on a cluster member.**

2. **Use the steps that the Apache documentation describes to install the Apache software.**

   See the documentation that you received with your Apache software or the Apache Web site at `http://www.apache.org`.

3. **Update the `httpd.conf` configuration file.**

   - Set the `ServerName` directive.
   - Set the `BindAddress` directive (optional).
   - Set the `ServerType`, `ServerRoot`, `DocumentRoot`, `ScriptAlias`, and `LockFile` directives.
   - Set the `Port` directive to the same number as the `Port_list` standard resource property. See Step 4 for more information.

- Make changes to run as a proxy server if you choose to run the Apache software as a proxy server. See the Apache documentation for more information. If you will run the Apache software as a proxy server, the `CacheRoot` setting must point to a location on the cluster file system.

4. **Verify that the port number or numbers in the `httpd.conf` file match those of the `Port_list` standard resource property.**

   You can edit the `httpd.conf` configuration file to change its port number or numbers to match the standard Sun Cluster resource property default (port `80`). Alternatively, while you configure the Sun Cluster HA for Apache data service, you can set the `Port_list` standard property to match the setting in the `httpd.conf` file.

5. **(Optional) If you will use the Apache start/stop script** *Bin_dir*`/apachectl`**, update the paths in the script file.**

   You must change the paths from the Apache defaults to match your Apache directory structure.

6. **Perform the following tasks to verify your configuration changes.**

   a. **Run `apachectl configtest` to check the Apache `httpd.conf` file for correct syntax.**

   b. **Ensure that any logical hostnames or shared addresses that Apache uses are configured and online.**

   c. **Issue `apachectl start` to start up your Apache server by hand. If Apache does not start up correctly, correct the problem.**

   d. **After Apache has started, stop it before moving to the next procedure.**

Where to Go From Here

If the Apache data-service packages have not been installed from the Sun Cluster Agents CD, go to "Installing Sun Cluster HA for Apache Packages" on page 92. Otherwise, go to "Registering and Configuring Sun Cluster HA for Apache" on page 93.

Installing Sun Cluster HA for Apache Packages

You can use the scinstall(1M) utility to install SUNWscapc, the Sun Cluster HA for Apache data-service package, on a cluster. Do not use the -s option to noninteractive scinstall to install all data-service packages.

If you installed the data-service packages during your initial Sun Cluster installation, proceed to "Registering and Configuring Sun Cluster HA for Apache" on page 93. Otherwise, use the following procedure to install the SUNWscapc package now.

▼ How to Install Sun Cluster HA for Apache Packages

You need the Sun Cluster Agents CD to complete this procedure. Run this procedure on all cluster members that can master the Sun Cluster HA for Apache data service.

1. **Load the Agents CD into the CD-ROM drive.**
2. **Run the scinstall utility with no options.**

   This step starts the scinstall utility in interactive mode.
3. **Select the Add Support for New Data Service to This Cluster Node menu option.**

   This option enables you to load software for any data services that exist on the CD.
4. **Exit the scinstall utility.**
5. **Unload the CD from the drive.**

Where to Go From Here

See "How to Register and Configure Sun Cluster HA for Apache" on page 93 to register the Sun Cluster HA for Apache data service and to configure the cluster for the data service.

Registering and Configuring Sun Cluster HA for Apache

This procedure describes how to use the `scrgadm`(1M) command to register and configure the Sun Cluster HA for Apache data service.

Apache can be configured as a failover service or as a scalable service, as follows.

- When you configure Apache as a failover service, you place the Apache application resources and the network resources in a single resource group.
- When you configure Apache as a scalable service, you create a scalable resource group for the Apache application resources and a failover resource group for the network resources.

The scalable resource group depends on the failover resource group. Additional steps are required to configure Apache as a scalable service. The leading text "For scalable services only" in the following procedure identifies these steps. If you are not configuring Apache as a scalable service, skip the steps marked "For scalable services only."

▼ How to Register and Configure Sun Cluster HA for Apache

Note – Run this procedure on any cluster member.

---

1. **Become superuser on a cluster member.**

2. **Register the resource type for the data service.**

```
# scrgadm -a -t SUNW.apache
```

`-a`	Adds the data-service resource type.
`-t SUNW.apache`	Specifies the predefined resource-type name for your data service.

3. **Create a failover resource group to hold the network and application resources.**

   This resource group is required for both failover and scalable services. For failover services, the resource group contains both network and failover application resources. For scalable services, the resource group contains network resources only. A dependency is created between this group and the resource group that contains the application resources.

   Optionally, you can select the set of nodes on which the data service can run with the `-h` option.

```
# scrgadm -a -g resource-group [-h nodelist]
```

`-a`
Adds a new configuration.

`-g` *resource-group*
Specifies the name of the failover resource group to add. This name can be your choice but must be unique for the resource groups within the cluster.

`-h` *nodelist*
An optional comma-separated list of physical node names or IDs that identify potential masters. The order specified here determines the order in which the nodes are considered as primary during failover.

Note – Use `-h` to specify the order of the node list. If all the nodes in the cluster are potential masters, you need not use the `-h` option.

---

4. **Verify that all network addresses that you use have been added to your name-service database.**

   You should have performed this verification during your initial Sun Cluster installation. See the planning chapter in the *Sun Cluster 3.0 U1 Installation Guide* for details.

Note – To avoid failures because of name-service lookup, verify that all network addresses are present in the `/etc/hosts` file on all cluster nodes. Configure name-service mapping in the `/etc/nsswitch.conf` file on the servers to first check the local files prior to accessing NIS, NIS+, or DNS.

---

5. **Add a network resource (logical hostname or shared address) to the failover resource group that you created in Step 3.**

```
# scrgadm -a {-S | -L} -g resource-group \
-l hostname, … [-j resource] \
[-X auxnodelist] [-n netiflist]
```

`-S` | `-L`	The `-S` option specifies shared-address resources. The `-L` option specifies logical-hostname resources.
`-l` *hostname, …*	Specifies a comma-separated list of network resources to add. You can use the `-j` option to specify a name for the resources. If you do not do so, the network resources have the name of the first entry on the list.
`-g` *resource-group*	Specifies the name of the failover resource group that you created in Step 3.
`-j` *resource*	Specifies a resource name. If you do not supply your choice for a resource name, the name of the network resource defaults to the first name that is specified after the `-l` option.
`-X` *auxnodelist*	Specifies a comma-separated list of physical node names or node IDs that identify cluster nodes that can host the shared address but never serve as primary in the case of failover. These nodes are mutually exclusive with the nodes identified in *nodelist* for the resource group, if specified.
`-n` *netiflist*	Specifies an optional comma-separated list that identifies the NAFO groups on each node. All nodes in *nodelist* of the resource group must be represented in *netiflist*. If you do not specify this option, `scrgadm` attempts to discover a net adapter on the subnet that the *hostname* list identifies for each node in *nodelist*.

6. **For scalable services only – Create a scalable resource group to run on all desired nodes of the cluster.**

   If you run the Sun Cluster HA for Apache data service as a failover data service, proceed to Step 8.

   Create a resource group to hold a data-service application resource. You must specify the maximum and desired number of primary nodes.

Note – If only a subset of nodes can be primaries for this resource group, you must specify the names of these potential primaries using the `-h` option when you create the resource group.

You must also specify any dependency between this resource group and the failover resource group that you created in Step 3. This dependency ensures that when failover occurs, if the two resource groups are being brought online on the same node, the Resource Group Manager (RGM) starts up the network resource before any data services that depend on the network resource.

```
# scrgadm -a -g resource-group \
-y Maximum_primaries=m -y Desired_primaries=n \
-y RG_dependencies=resource-group \
[-h nodelist]
```

`-g` *resource-group*

Specifies the name of the scalable-service resource group to add.

`-y Maximum_primaries=`*m*

Specifies the maximum number of active primary nodes allowed for this resource group. If you do not assign a value to this property, the default is `1`.

`-y Desired_primaries=`*n*

Specifies the desired number of active primary nodes allowed for this resource group. If you do not assign a value to this property, the default is `1`.

`-y RG_dependencies=` *resource-group*

Identifies the resource group that contains the shared-address resource on which the resource group being created depends, that is, the name of the failover resource group created in Step 3.

`-h` *nodelist*

An optional list of nodes that can be primaries for this resource group. You only need to specify this list if some nodes cannot act as primaries for this resource group.

7. **For scalable services only – Create an application resource in the scalable resource group.**

   If you run the Sun Cluster HA for Apache data service as a failover data service, proceed to Step 8.

```
# scrgadm -a -j resource -g resource-group \
-t resource-type -y Network_resources_used=network-resource, … \
-y Port_list=port-number/protocol[, …] -y Scalable=True \
-x Confdir_list=config-directory -x Bin_dir=bin-directory
```

`-j` *resource*

Specifies your choice for the name of the resource to add.

`-g` *resource-group*

Specifies the name of the scalable resource group into which the resources are to be placed.

`-t` *resource-type*

Specifies the type of the resource to add.

`-y Network_resources_used=`*network-resource, …*

Specifies a comma-separated list of network-resource names that identify the shared addresses that the data service uses.

`-y Port_list=`*port-number/protocol, …*

Specifies a comma-separated list of port numbers and protocol to be used, for example, `80/tcp,81/tcp`.

`-y Scalable=`

Specifies a required parameter for scalable services. Must be set to `True`.

`-x Confdir_list=`*config-directory, …*

Specifies a comma-separated list of the locations of the Apache configuration files. The Sun Cluster HA for Apache data service requires this extension property.

`-x Bin_dir=`*bin-directory*

Specifies the location where the Apache binaries are installed. The Sun Cluster HA for Apache data service requires this extension property.

Note – Optionally, you can set additional extension properties that belong to the Apache data service to override the default value. See TABLE 5-2 for a list of extension properties.

---

8. **For failover services only – Create an application resource in the failover resource group.**

   Perform this step only if you run the Sun Cluster HA for Apache data service as a failover data service. If you run the Sun Cluster HA for Apache data service as a scalable service, you should have performed Step 6 and Step 7 and should now proceed to Step 10.

```
# scrgadm -a -j resource -g resource-group \
-t resource-type -y Network_resources_used=network-resource, … \
-y Port_list=port-number/protocol[, …] -y Scalable=False \
-x Confdir_list=config-directory -x Bin_dir=bin-directory
```

`-j` *resource*

Specifies your choice for the name of the resource to add.

`-g` *resource-group*

Specifies the name of the resource group into which the resources are to be placed, created in Step 3.

`-t` *resource-type*

Specifies the type of the resource to add.

`-y Network_resources_used=`*network-resource, …*

Specifies a comma-separated list of network resources that identify the shared addresses that the data service uses.

`-y Port_list=`*port-number/protocol, …*

Specifies a comma-separated list of port numbers and protocol to be used, for example, `80/tcp,81/tcp`.

`-y Scalable=` This property is required for scalable services only. Here the value is set to `False` or can be omitted.

`-x Confdir_list=`*config-directory* Specifies the location of the Apache configuration file. The Sun Cluster HA for Apache data service requires this extension property, and the property must have exactly one entry.

`-x Bin_dir=`*bin-directory* Specifies the location where the Apache binaries are installed. The Sun Cluster HA for Apache data service requires this extension property.

9. **Bring the failover resource group online.**

```
# scswitch -Z -g resource-group
```

`-Z` Enables the shared-address resource and fault monitoring, switches the resource group into a managed state, and brings the resource group online.

`-g` *resource-group* Specifies the name of the failover resource group.

10. **For scalable services only – Bring the scalable resource group online.**

```
# scswitch -Z -g resource-group
```

`-Z` Enables the resource and monitor, moves the resource group to the managed state, and brings the resource group online.

`-g` *resource-group* Specifies the name of the scalable resource group.

Example – Registering Scalable Sun Cluster HA for Apache

For scalable services, you create the following resource groups.

- a failover resource group that contains the network resources
- a scalable resource group that contains the application resources

The following example shows how to register a scalable Apache service on a two-node cluster.

```
Cluster Information
Node names: phys-schost-1, phys-schost-2
Shared address: schost-1
Resource groups: resource-group-1 (for shared addresses),
    resource-group-2 (for scalable Apache application
    resources)
Resources: schost-1 (shared address), apache-1 (Apache application
    resource)

(Add a failover resource group to contain shared addresses.)
# scrgadm -a -g resource-group-1

(Add the shared address resource to the failover resource group.)
# scrgadm -a -S -g resource-group-1 -l schost-1

(Register the Apache resource type.)
# scrgadm -a -t SUNW.apache

(Add a scalable resource group.)
# scrgadm -a -g resource-group-2 -y Maximum_primaries=2 \
-y Desired_primaries=2 -y RG_dependencies=resource-group-1

(Add Apache application resources to the scalable resource group.)
# scrgadm -a -j apache-1 -g resource-group-2 \
-t SUNW.apache -y Network_resources_used=schost-1 \
-y Scalable=True -y Port_list=80/tcp \
-x Bin_dir=/opt/apache/bin -x Confdir_list=/opt/apache/conf

(Bring the failover resource group online.)
# scswitch -Z -g resource-group-1

(Bring the scalable resource group online on both nodes.)
# scswitch -Z -g resource-group-2
```

Example – Registering Failover Sun Cluster HA for Apache

The following example shows how to register a failover Apache service on a two-node cluster.

```
Cluster Information
Node names: phys-schost-1, phys-schost-2
Logical hostname: schost-1
Resource group: resource-group-1 (for all resources)
Resources: schost-1 (logical hostname),
    apache-1 (Apache application resource)

(Add a failover resource group to contain all resources.)
# scrgadm -a -g resource-group-1

(Add the logical hostname resource to the failover resource group.)
# scrgadm -a -L -g resource-group-1 -l schost-1

(Register the Apache resource type.)
# scrgadm -a -t SUNW.apache

(Add Apache application resources to the failover resource group.)
# scrgadm -a -j apache-1 -g resource-group-1 \
-t SUNW.apache -y Network_resources_used=schost-1 \
-y Scalable=False -y Port_list=80/tcp \
-x Bin_dir=/opt/apache/bin -x Confdir_list=/opt/apache/conf

(Bring the failover resource group online.)
# scswitch -Z -g resource-group-1
```

Where to Go From Here

Use the information in “How to Verify Data-Service Installation and Configuration” on page 102 to verify the installation. See “Configuring Sun Cluster HA for Apache Extension Properties” on page 102 to set or modify resource extension properties.

How to Configure `SUNW.HAStorage` Resource Type

The `SUNW.HAStorage` resource type synchronizes actions between HA storage and the data service. The Sun Cluster HA for Apache data service is scalable, and therefore you should configure the `SUNW.HAStorage` resource type.

See the SUNW.HAStorage(5) man page and "Relationship Between Resource Groups and Disk Device Groups" on page 4 for background information. See "How to Set Up SUNW.HAStorage Resource Type for New Resources" on page 239 for the procedure.

▼ How to Verify Data-Service Installation and Configuration

After you configure the Sun Cluster HA for Apache data service, verify that you can open a Web page with the network resources (logical hostnames or shared addresses) and port number from a Web browser. Perform a switchover with the scswitch(1M) command to verify that the service continues to run on a secondary node and can be switched back to the original primary.

Configuring Sun Cluster HA for Apache Extension Properties

The only required extension properties for creating an Apache server resource are the Confdir_list and Bin_dir properties. The Confdir_list property specifies a directory that contains a subdirectory named conf, in which the Apache configuration properties (httpd.conf) reside.

Typically, you use the command-line scrgadm -x *parameter=value* to configure the extension properties when you create the Apache server resource. You can also follow the procedures described in Chapter 11 to configure the properties later.

See Appendix A for details on all Sun Cluster properties.

You can update some extension properties dynamically. You can update others, however, only when you create the Apache server resource. The following table describes extension properties that you can configure for the Apache server. The Tunable column indicates when you can update the property.

TABLE 5-2 Sun Cluster HA for Apache Extension Properties

Name/Data Type	Default	Range	Tunable	Description
`Bin_dir` (string)	None	None	At creation	The path to the Apache binaries. The Sun Cluster HA for Apache data service requires this extension property.
`Confdir_list` (string array)	None	None	At creation	The directory that contains a subdirectory called `conf`, which contains the `httpd.conf` configuration file. The Sun Cluster HA for Apache data service requires this extension property.
`Monitor_retry_count` (integer)	`4`	`0` – `2,147,483,641` –1 indicates an infinite number of retry attempts.	At creation	Controls restarts of the fault monitor and indicates the number of times that the process monitor facility (PMF) restarts the fault monitor during the time window that the `Monitor_retry_interval` property specifies. This property refers to restarts of the fault monitor itself rather than to the resource. The system-defined properties `Retry_interval` and `Retry_count` control resource restarts.
`Monitor_retry_interval` (integer)	`2`	`0` – `2,147,483,641` `–1` indicates an infinite retry interval.	At creation	The time (in minutes) over which failures of the fault monitor are counted. If the number of times that the fault monitor fails exceeds the value that is specified in the extension property `Monitor_retry_count` within this period, the PMF does not restart the fault monitor.
`Probe_timeout` (integer)	`30`	`0` – `2,147,483,641`	At creation	The time-out value (in seconds) that the fault monitor uses to probe an Apache instance.

Sun Cluster HA for Apache Fault Monitor

The Sun Cluster HA for Apache probe sends a request to the server to query the health of the Apache server. Before the probe actually queries the Apache server, the probe checks to confirm that network resources are configured for this Apache resource. If no network resources are configured, an error message (`No network resources found for resource`) is logged, and the probe exits with failure.

The probe executes the following steps.

1. Uses the time-out value that the resource property `Probe_timeout` sets to limit the time spent trying to successfully probe the Apache server.

2. Connects to the Apache server and performs an HTTP 1.0 HEAD check by sending the HTTP request and receiving a response. In turn, the probe connects to the Apache server on each IP address/port combination.

   The result of this query can be either a failure or a success. If the probe successfully receives a reply from the Apache server, the probe returns to its infinite loop and continues the next cycle of probing and sleeping.

   The query can fail for various reasons, such as heavy network traffic, heavy system load, and misconfiguration. Misconfiguration can occur if the Apache server is not configured to be listening on all IP address/port combinations that are being probed. The Apache server should service every port for every IP address specified for this resource. If the reply to the query is not received within the `Probe_timeout` limit (specified in Step 1 previously), the probe considers this scenario a failure on the part of the Apache data service and records the failure in its history. An Apache probe failure can be a complete failure or a partial failure.

   The following probe failures are considered complete failures.

- Failure to connect to the server, as the following error message flags, with `%s` indicating the host name and `%d` the port number.

```
Failed to connect to %s port %d %s
```

- Running out of time (exceeding the resource property time-out `Probe_timeout`) after trying to connect to the server.

- Failure to successfully send the probe string to the server, as the following error message flags, with the first %s indicating the host name, %d the port number, and the second %s indicating further details about the error.

```
Failed to communicate with server %s port %d: %s
```

  The monitor accumulates two such partial failures within the resource property interval Retry_interval and counts them as one.

  The following probe failures are considered partial failures.

- Running out of time (exceeding the resource property timeout Probe_timeout) while trying to read the reply from the server to the probe's query.
- Failing to read data from the server for other reasons, as the following error message flags, with the first %s indicating the host name and %d the port number. The second %s indicates further details about the error.

```
Failed to communicate with server %s port %d: %s
```

3. Based on the history of failures, a failure can cause either a local restart or a failover of the data service. "Health Checks of the Data Service" on page 11 further describes this action.

CHAPTER 6

Installing and Configuring Sun Cluster HA for Domain Name Service (DNS)

This chapter describes the steps to install and configure the Sun Cluster HA for Domain Name Service (DNS) data service on your Sun Cluster servers.

This chapter contains the following procedures.

You must configure the Sun Cluster HA for DNS data service as a failover service. See Chapter 1 and the *Sun Cluster 3.0 U1 Concepts* document for general information on data services, resource groups, resources, and other related topics.

Note – You can use SunPlex Manager to install and configure this data service. See the SunPlex Manager online help for details.

Installing and Configuring Sun Cluster HA for DNS

The following table lists the sections that describe the installation and configuration tasks.

TABLE 6-1 Task Map: Installing and Configuring Sun Cluster HA for NFS

Task	For Instructions, Go To …
Install DNS	"Installing DNS" on page 108
Install Sun Cluster HA for DNS packages	"Installing Sun Cluster HA for DNS Packages" on page 111
Configure and start Sun Cluster HA for DNS data service	"Registering and Configuring Sun Cluster HA for DNS" on page 112
Configure resource extension properties	"Configuring Sun Cluster HA for DNS Extension Properties" on page 117
View fault-monitor information	"Sun Cluster HA for DNS Fault Monitor" on page 119

Installing DNS

This section describes the steps to install DNS and to enable DNS to run as the Sun Cluster HA for DNS data service.

The Sun Cluster HA for DNS data service uses the Internet Domain Name Server (`in.named`) software that is bundled with the Solaris 8 operating environment. See the `in.named`(1M) man page for information on how to set up DNS. The Sun Cluster configuration involves the following differences.

- The DNS database is located on the cluster file system, not a local file system.
- A logical hostname (relocatable IP address), not the name of a physical host, identifies the name of a DNS server.

▼ How to Install DNS

1. **Become superuser on a cluster member.**

2. **Decide on the logical hostname that will provide the DNS service.**

   This name should be a hostname that is set up when you install the Sun Cluster software. See the *Sun Cluster 3.0 U1 Installation Guide* for details on how to set up hostnames.

3. **Ensure that the DNS executable (**`in.named`**) is in the directory** `/usr/sbin`**.**

   The DNS executable is bundled with the Solaris 8 operating environment and is located in the `/usr/sbin` directory before you begin the installation.

4. **Create a directory structure on the cluster file system to hold the DNS configuration and database files.**

   Note – Create a `dns` directory and a `named` directory underneath the `dns` directory on a cluster file system, for example, `/global/dns/named`. See the *Sun Cluster 3.0 U1 Installation Guide* for information on how to set up cluster file systems.

   ```
   # mkdir -p /global/dns/named
   ```

5. **Place the configuration file for DNS,** `named.conf` **or** `named.boot`**, under** `/global/dns`**.**

   If DNS is already installed, you can copy the existing `named.conf` or `named.boot` file to the `/global/dns` directory. Otherwise, create a `named.conf` file in this directory. See the `in.named`(1M) man page for information on the types of entries to place in `named.conf` or `named.boot`. One of the two files, `named.conf` or `named.boot`, must exist. Both files can exist.

6. **Place all the DNS database files (listed in the** `named.conf` **file) under the** `/global/dns/named` **directory.**

7. **On all the clients of the Sun Cluster HA for DNS data service, create an entry for the logical hostname of the DNS service in the `/etc/resolv.conf` file.**

   On all the nodes, edit the `/etc/resolv.conf` file to contain the logical hostname. The following example shows the entries for a four-node configuration (`phys-schost-1`, `phys-schost-2`, `phys-schost-3`, and `phys-schost-4`) with the logical hostname `schost-1.eng.sun.com`.

```
domain eng.sun.com

; schost-1.eng.sun.com
(Only entry to be added if the file is already present.)

nameserver 192.29.72.90

; phys-schost-2.eng
nameserver 129.146.1.151

; phys-schost-3.eng
nameserver 129.146.1.152

; phys-schost-4.eng
nameserver 129.144.134.19

; phys-schost-1.eng
nameserver 129.144.1.57
```

   Make the logical hostname the first entry after the domain name. DNS attempts to use the addresses in the order that they are listed in the `resolv.conf` file to access the server.

   Note – If the `/etc/resolv.conf` is already present on the nodes, just add the first entry that shows the logical hostname in the preceding example. The order of the entries determines the order in which DNS tries to access the server.

   ---

8. **On all the cluster nodes, edit the `/etc/inet/hosts` file to create an entry for the logical hostname of the DNS service.**

   In the following example, perform these steps.

   - Replace the *IPaddress* variable with your actual IP address, such as `129.146.87.53`.
   - Replace the *logical-hostname* variable with your actual logical hostname.

```
127.0.0.1      localhost
IPaddress      logical-hostname
```

9. **On all the cluster nodes, edit the `/etc/nsswitch.conf` file to add the string `dns` after `cluster` and `files` to the `hosts` entry.**

   The following example shows how to complete this step.

   ```
   hosts:  cluster files dns
   ```

10. **Test DNS.**

    Be sure to stop the `in.named` executable before you proceed. The following example shows how to test DNS.

    ```
    # cd /global/dns
    # /usr/sbin/in.named -c /global/dns/named.conf
    # nslookup phys-schost-1
    # pkill -x /usr/sbin/in.named
    ```

Where to Go From Here

If you installed the Sun Cluster HA for DNS packages during your Sun Cluster installation, go to "Registering and Configuring Sun Cluster HA for DNS" on page 112. Otherwise, go to "Installing Sun Cluster HA for DNS Packages" on page 111.

Installing Sun Cluster HA for DNS Packages

You can use the `scinstall`(1M) utility to install `SUNWscdns`, the Sun Cluster HA for DNS data-service package, on a cluster. Do not use the `-s` option to non-interactive `scinstall` to install all data-service packages.

If you installed the `SUNWscdns` data-service package during your initial Sun Cluster installation, proceed to "Registering and Configuring Sun Cluster HA for DNS" on page 112. Otherwise, use the following procedure to install the `SUNWscdns` package.

▼ How to Install Sun Cluster HA for DNS Packages

You need the Sun Cluster Agents CD to complete this procedure. Perform this procedure on all cluster nodes that can run the Sun Cluster HA for DNS data service.

1. **Load the Agents CD into the CD-ROM drive.**
2. **Run the `scinstall` utility with no options.**

   This step starts the `scinstall` utility in interactive mode.
3. **Select the Add Support for New Data Service to This Cluster Node menu option.**

   This option enables you to load software for any data services that exist on the CD.
4. **Exit the `scinstall` utility.**
5. **Unload the CD from the drive.**

Where to Go From Here

See "Registering and Configuring Sun Cluster HA for DNS" on page 112 to register the Sun Cluster HA for DNS data service and to configure the cluster for the data service.

Registering and Configuring Sun Cluster HA for DNS

This procedure describes how to use the `scrgadm`(1M) command to register and configure the Sun Cluster HA for DNS data service.

Note – Other options also enable you to register and configure the data service. See "Tools for Data-Service Resource Administration" on page 8 for details about these options.

▼ How to Register and Configure Sun Cluster HA for DNS

To perform this procedure, you need the following information about your configuration.

- The name of the resource type for the Sun Cluster HA for DNS data service. This name is `SUNW.dns`.
- The names of the cluster nodes that master the data service.
- The logical hostname that clients use to access the data service. This IP address is normally set up when the cluster is installed. See the section on how to set up logical hostnames in the *Sun Cluster 3.0 U1 Installation Guide* for details.
- The path to the DNS configuration files, which you must install on a cluster file system. This path maps to the `Config_dir` resource property that is configured in this procedure.

Note – Perform this procedure on any cluster member.

---

1. **Become superuser on a cluster member.**

2. **Register the resource type for the data service.**

```
# scrgadm -a -t SUNW.dns
```

`-a` Adds the data-service resource type.

`-t SUNW.dns` Specifies the predefined resource-type name for your data service.

3. **Create a resource group for logical hostnames and DNS resources to use.**

You can use the `-h` option to optionally select the set of nodes on which the data service can run.

```
# scrgadm -a -g resource-group [-h nodelist]
```

`-g` *resource-group* Specifies the name of the resource group. This name can be your choice but must be unique for the resource groups within the cluster.

`-h` *nodelist* Specifies an optional comma-separated list of physical node names or IDs that identify potential masters. The order here determines the order in which the nodes are considered as primary during failover.

Note – Use the `-h` option to specify the order of the node list. If all the nodes in the cluster are potential masters, you do not need to use the `-h` option.

---

4. **Verify that all logical hostnames that you will use have been added to your name-service database.**

   You should have performed this verification during the Sun Cluster installation. See the planning chapter in the *Sun Cluster 3.0 U1 Installation Guide* for details.

Note – To avoid any failures because of name-service lookup, verify that all logical hostnames are present in the server's and client's `/etc/hosts` file. Configure name-service mapping in the `/etc/nsswitch.conf` file on the servers to first check the local files before trying to access NIS or NIS+.

---

5. **Add logical-hostname resources to the resource group.**

```
# scrgadm -a -L -g resource-group \
-l logical-hostname[,logical-hostname] [-j resource] \
[-n netiflist]
```

`-L` Specifies the logical-hostname resources.

`-l` *logical-hostname*	Specifies a comma-separated list of logical hostnames.
`-j` *resource*	Specifies an optional network resource name. If you do not specify this name, the value defaults to the first name specified after the `-l` option.
`-n` *netiflist*	Specifies an optional comma-separated list that identifies the NAFO groups on each node. All the nodes in *nodelist* of the resource group must be represented in *netiflist*. If you do not specify this option, the `scrgadm` command attempts to discover a net adapter on the subnet that the *hostname* list identifies for each node in *nodelist*.

6. **Add a DNS application resource to the resource group.**

```
# scrgadm -a -j [resource] -g resource-group \
-t SUNW.dns -y Network_resources_used=network-resource, …\
-y Port_list=port-number/protocol -x DNS_mode=config-file \
-x Confdir_list=config-directory
```

`-j` *resource*	Specifies the DNS application resource name.
`-t SUNW.dns`	Specifies the name of the resource type to which this resource belongs. This entry is required.
`-y Network_resources_used=` *network-resource, …*	Specifies a comma-separated list of network resources (logical hostnames) that DNS will use. If you do not specify this property, the value defaults to all the logical hostnames contained in the resource group.
`-y Port_list=` *port-number/protocol*	Specifies a port number and the protocol to be used. If you do not specify this property, the value defaults to `53/udp`.
`-x DNS_mode=`*config-file*	Specifies the configuration file to use, either `conf(named.conf)` or `boot(named.boot)`. If you do not specify this property, the value defaults to `conf`.
`-x Confdir_list=`*config-directory*	Specifies the location of the DNS configuration directory paths, which must be on the cluster file system. The Sun Cluster HA for DNS data service requires this extension property.

7. **Run the `scswitch`(1M) command to complete the following tasks.**
   - Enable the resource and fault monitoring.
   - Move the resource group into a managed state.
   - Bring the resource group online.

```
# scswitch -Z -g resource-group
```

`-Z` Enables the resource and monitor, moves the resource group to the managed state, and brings the resource group online.

`-g` *resource-group* Specifies the name of the resource group.

Example – Registering Failover Sun Cluster HA for DNS

The following example shows how to register the Sun Cluster HA for DNS data service on a two-node cluster. Note that at the end, the `scswitch` command starts the Sun Cluster HA for DNS data service.

```
Cluster Information
Node names: phys-schost-1, phys-schost-2
Logical hostname: schost-1
Resource group: resource-group-1 (for all resources),
Resources: schost-1 (logical hostname), dns-1 (DNS application
    resource)

(Register the DNS resource type)
# scrgadm -a -t SUNW.dns

(Add the resource group to contain all resources.)
# scrgadm -a -g resource-group-1

(Add the logical hostname resource to the resource group.)
# scrgadm -a -L -g resource-group-1 -l schost-1

(Add DNS application resources to the resource group.)
# scrgadm -a -j dns-1 -g resource-group-1 -t SUNW.dns \
-y Network_resources_used=schost-1 -y Port_list=53/udp \
-x DNS_mode=conf -x Confdir_list=/global/dns

(Bring the failover resource group online.)

# scswitch -Z -g resource-group-1
```

▼ How to Configure SUNW.HAStorage Resource Type

The SUNW.HAStorage resource type synchronizes actions between HA storage and the data service. The Sun Cluster HA for DNS data service is not disk intensive and is not scalable, and therefore setting up the SUNW.HAStorage resource type is optional.

See the SUNW.HAStorage(5) man page and "Relationship Between Resource Groups and Disk Device Groups" on page 4 for background information. See "How to Set Up SUNW.HAStorage Resource Type for New Resources" on page 239 for the procedure.

Verifying Data Service Installation and Configuration

To verify that you have correctly installed and configured the Sun Cluster HA for DNS data service, run the following command after you complete the procedure "How to Register and Configure Sun Cluster HA for DNS" on page 113.

```
# nslookup logical-hostname logical-hostname
```

In this example, *logical-hostname* is the name of the network resource that you have configured to service DNS requests—for example, schost-1—as shown in the previous registration example. The output should indicate that the logical host that you specified answered (served) the query.

Configuring Sun Cluster HA for DNS Extension Properties

The only required extension property for creating a DNS resource is the Confdir_list property. Typically, you use the command line scrgadm -x *parameter=value* to configure extension properties when you create the DNS resource. You can also use the procedures in Chapter 11 to configure them later.

See Appendix A for details on all Sun Cluster properties.

TABLE 6-2 describes the Sun Cluster HA for DNS extension properties. You can update some extension properties dynamically. You can update others, however, only when you create the resource. The Tunable column indicates when you can update the property.

TABLE 6-2 Sun Cluster HA for DNS Extension Properties

Name/Data Type	Default	Range	Tunable	Description
`Confdir_list` (string array)	None	None	At creation	A comma-separated list of path names, each of which points to the directory that contains the `conf` directory for a DNS instance.
`DNS_mode`	`conf`	None	At creation	The DNS configuration file to use, either `conf` (`named.conf`) or `boot` (`named.boot`).
`Monitor_retry_count` (integer)	4	`0 – 2,147,483,641` `–1` indicates an infinite number of retry attempts.	Any time	The number of times that the process monitor facility (PMF) restarts the fault monitor during the time window that the `Monitor_retry_interval` property specifies. This property refers to restarts of the fault monitor itself rather than to the resource. The system-defined properties `Retry_interval` and `Retry_count` control restarts of the resource.
`Monitor_retry_interval` (integer)	2	`0 – 2,147,483,641` `–1` indicates an infinite retry interval.	Any time	The time (in minutes) over which failures of the fault monitor are counted. If the number of times that the fault monitor fails exceeds the value that is specified in the extension property `Monitor_retry_count` within this period, the PMF does not restart the fault monitor.
`Probe_timeout` (integer)	30	`0 – 2,147,483,641`	Any time	The time-out value (in seconds) that the fault monitor uses to probe a DNS instance.

Sun Cluster HA for DNS Fault Monitor

The probe uses the nslookup command to query the health of DNS. Before the probe actually queries the DNS server, a check is made to confirm that network resources are configured in the same resource group as the DNS data service. If no network resources are configured, an error message is logged, and the probe exits with failure.

The probe executes the following steps.

1. Run the nslookup command using the time-out value that the resource property Probe_timeout specifies.

   The result of this nslookup command can be either failure or success. If DNS successfully replied to the nslookup query, the probe returns to its infinite loop, waiting for the next probe time.

   If the nslookup fails, the probe considers this scenario a failure of the DNS data service and records the failure in its history. The DNS probe considers every failure a complete failure.

2. Based on the success/failure history, a failure can cause a local restart or a data-service failover. "Health Checks of the Data Service" on page 11 further describes this action.

CHAPTER 7

Installing and Configuring Sun Cluster HA for Network File System (NFS)

This chapter describes the steps to install and configure the Sun Cluster HA for Network File System (NFS) data service on your Sun Cluster servers and the steps to add the Sun Cluster HA for NFS data service to a system that already runs the Sun Cluster software.

This chapter contains the following procedures.

- "How to Install Sun Cluster HA for NFS Packages" on page 123
- "How to Set Up and Configure Sun Cluster HA for NFS" on page 124
- "How to Change Share Options on an NFS File System" on page 130
- "How to Tune Sun Cluster HA for NFS Method Timeouts" on page 131
- "How to Configure `SUNW.HAStorage` Resource Type" on page 132

You must configure the Sun Cluster HA for NFS data service as a failover service. See Chapter 1 and the *Sun Cluster 3.0 U1 Concepts* document for general information about data services, resource groups, resources, and other related topics.

Note – You can use SunPlex Manager to install and configure this data service. See the SunPlex Manager online help for details.

Use the worksheets in *Sun Cluster 3.0 U1 Release Notes* to plan your resources and resource groups before you install and configure the Sun Cluster HA for NFS data service.

The mount points for NFS file systems placed under the control of the Sun Cluster HA for NFS data service must be the same on all the nodes that are capable of mastering the disk device group that contains those file systems.

Note – To avoid any failures because of name-service lookup, ensure that all logical hostnames are present in the server's `/etc/hosts` file. Configure name-service mapping in the `/etc/nsswitch.conf` file on the servers to first check the local files before trying to access NIS or NIS+. Doing so prevents timing-related errors in this area and ensures that `ifconfig` and `statd` succeed in resolving logical hostnames.

Note – To avoid "stale file handle" errors on the client during NFS failover, if you use VERITAS Volume Manager, ensure that the `vxio` driver has identical pseudo-device major numbers on all the cluster nodes. You can find this number in the `/etc/name_to_major` file after you complete the installation.

Installing and Configuring Sun Cluster HA for NFS

The following table lists the sections that describe the installation and configuration tasks.

TABLE 7-1 Task Map: Installing and Configuring Sun Cluster HA for NFS

Task	For Instructions, Go To …
Install Sun Cluster HA for NFS packages	"Installing Sun Cluster HA for NFS Packages" on page 123
Set up and configure the Sun Cluster HA for NFS data service	"Setting Up and Configuring Sun Cluster HA for NFS" on page 124
Configure resource extension properties	"Configuring Sun Cluster HA for NFS Extension Properties" on page 132
View fault-monitor information	"Sun Cluster HA for NFS Fault Monitor" on page 134

Installing Sun Cluster HA for NFS Packages

You can use the `scinstall`(1M) utility to install `SUNWscnfs`, the Sun Cluster HA for NFS data-service package, on a cluster. Do not use the `-s` option to non-interactive `scinstall` to install all data-service packages.

If you installed the `SUNWscnfs` data-service package during your initial Sun Cluster installation, proceed to "Setting Up and Configuring Sun Cluster HA for NFS" on page 124. Otherwise, use the following procedure to install the `SUNWscnfs` package.

▼ How to Install Sun Cluster HA for NFS Packages

You need the Sun Cluster Agents CD to complete this procedure. Run this procedure on all cluster nodes that can run the Sun Cluster HA for NFS data service.

1. **Load the Agents CD into the CD-ROM drive.**
2. **Run the `scinstall` utility with no options.**

   This step starts the `scinstall` utility in interactive mode.
3. **Select the Add Support for New Data Service to This Cluster Node menu option.**

   This option enables you to load software for any data services that exist on the CD.
4. **Exit the `scinstall` utility.**
5. **Unload the CD from the drive.**

Where to Go From Here

See "Setting Up and Configuring Sun Cluster HA for NFS" on page 124 to register the Sun Cluster HA for NFS data service and to configure the cluster for the data service.

Setting Up and Configuring Sun Cluster HA for NFS

This procedure describes how to use the scrgadm(1M) command to register and configure the Sun Cluster HA for NFS data service.

Note – Other options also enable you to register and configure the data service. See "Tools for Data-Service Resource Administration" on page 8 for details about these options.

Before you set up and configure the Sun Cluster HA for NFS data service, run the following command to verify that the Sun Cluster HA for NFS package SUNWscnfs has been installed on the cluster.

```
# pkginfo -l SUNWscnfs
```

If the package has not been installed, see "Installing Sun Cluster HA for NFS Packages" on page 123 for instructions on how to install the package.

How to Set Up and Configure Sun Cluster HA for NFS

1. **Become superuser on a cluster member.**

2. **Verify that all nodes in the cluster are up and running.**

```
# scstat
```

3. **Create a failover resource group to contain the NFS resources.**

Note – You must define `Pathprefix`, which is a directory on a cluster file system that the Sun Cluster HA for NFS data service uses to maintain administrative and status information, for the Sun Cluster HA for NFS resource group. You can specify any directory for this purpose, but you must create the directory on the cluster file system.

```
# scrgadm -a -g resource-group -y Pathprefix=/global/admin-dir
```

Note – The *admin-dir* directory must be on a cluster file system. You must manually create this directory.

`-a -g` *resource-group*
Adds the named failover resource group.

`-y Pathprefix=`*path*
Specifies a directory on a cluster file system for Sun Cluster HA for NFS administration files to use. For example, `/global/nfs`. `Pathprefix` must be unique for each resource group that you create.

4. **Verify that all logical hostnames that you will use have been added to your name-service database.**

   You should have performed this verification during the Sun Cluster installation. See the planning chapter in the *Sun Cluster 3.0 U1 Installation Guide* for details on this step.

Note – To avoid any failures because of name-service lookup, verify that all logical hostnames are present in the server's and client's `/etc/hosts` file. Configure name-service mapping in the `/etc/nsswitch.conf` file on the servers to first check the local files before trying to access NIS or NIS+. Doing so prevents timing-related errors in this area and ensures that `ifconfig` and `statd` succeed in resolving logical hostnames.

5. **Add the desired logical-hostname resources into the failover resource group.**

   You must set up a `LogicalHostname` resource with this step. The hostname used with the Sun Cluster HA for NFS data service *cannot* be a `SharedAddress` resource.

```
# scrgadm -a -L -g resource-group -l logical-hostname, … [-n netiflist]
```

`-a -L -g` *resource-group* Specifies the failover resource group into which to place the logical-hostname resources.

`-l` *logical-hostname, …* Specifies the network resources (logical hostnames) to be added.

`-n` *netiflist* Specifies an optional comma-separated list that identifies the NAFO groups on each node. All the nodes in *nodelist* of the resource group must be represented in *netiflist*. If you do not specify this option, `scrgadm`(1M) attempts to discover a net adapter on the subnet that the *hostname* list identifies for each node in *nodelist*.

6. **From any node of the cluster, create a directory structure for the NFS configuration files.**

   Create the administrative subdirectory directory below the directory that the `Pathprefix` property identifies in Step 3, for example, `/global/nfs/SUNW.nfs`.

   ```
   # mkdir Pathprefix/SUNW.nfs
   ```

7. **Create a `dfstab.`*resource* file in the `SUNW.nfs` directory created in Step 6, and set up share options.**

   For example, create the *Pathprefix*`/SUNW.nfs/dfstab.`*resource* file, which contains a set of `share` commands with the shared path names. The shared paths should be subdirectories on a cluster file system.

   Choose a *resource*-name suffix to identify the NFS resource that you plan to create (in Step 9). A good resource name would refer to the task that this resource is expected to perform. For example, a name such as `user-nfs-home` is a good candidate for an NFS resource that shares user home directories.

   Set up the `share` options for each path that you have created to be shared. The format of this file is exactly the same as the format that is used in the `/etc/dfs/dfstab` file.

```
share [-F nfs] [-o] specific_options [-d “description”] pathname
```

`-F nfs`	Identifies the file system type as `nfs`.
`-o` *specific_options*	See the `share`(1M) man page for a list of options. You should set the `rw` option for Sun Cluster. This command grants read-write access to all clients.
`-d` *description*	Describes the file system being added.
pathname	Identifies the file system being shared.

   The `share -o rw` command grants write access to all clients, including the hostnames that the Sun Cluster software uses and enables Sun Cluster HA for NFS fault monitoring to operate most efficiently. See the following man pages for details.

   - `dfstab`(4)
   - `share`(1M)
   - `share_nfs`(1M)

   If you specify a client list in the `share` command, include all physical and logical hostnames that are associated with the cluster, as well as the hostnames for all clients on all public networks to which the cluster is connected.

   If you use net groups in the `share` command (rather than names of individual hosts), add all those cluster hostnames to the appropriate net group.

Note – Do not grant access to the hostnames on the cluster interconnect.

Grant read and write access to all the cluster nodes and logical hosts to enable the Sun Cluster HA for NFS monitoring to do a thorough job. However, you can restrict write access to the file system or make the file system entirely read-only. If you do so, Sun Cluster HA for NFS fault monitoring can still perform monitoring without having write access.

Note – When constructing `share` options, do not use the `root` option, and do not mix the `ro` and `rw` options.

8. **Register the NFS resource type.**

```
# scrgadm -a -t resource-type
```

`-a -t` *resource-type* Adds the specified resource type. For the Sun Cluster HA for NFS data service, the resource type is `SUNW.nfs`.

9. **Create the NFS resource in the failover resource group.**

```
# scrgadm -a -j resource -g resource-group -t resource-type
```

`-a` Adds a resource.

`-j` *resource* Specifies the name of the resource to add, which you defined in Step 7. This name can be your choice but must be unique within the cluster.

`-g` *resource-group* Specifies the name of a previously created resource group to which this resource is to be added.

`-t` *resource-type* Specifies the name of the resource type to which this resource belongs. This name must be the name of a registered resource type.

10. **Run the `scswitch`(1M) command to perform the following tasks.**

- Enable the resource and the resource monitor.
- Manage the resource group.

- Switch the resource group into the online state.

```
# scswitch -Z -g resource-group
```

Example – Setting Up and Configuring Sun Cluster HA for NFS

The following example shows how to set up and configure the Sun Cluster HA for NFS data service.

```
(Create a logical host resource group and specify the path to the administrative
files used by NFS (Pathprefix).)
# scrgadm -a -g resource-group-1 -y Pathprefix=/global/nfs

(Add logical hostname resources into the logical host resource group.)
# scrgadm -a -L -g resource-group-1 -l schost-1

(Make the directory structure contain the Sun Cluster HA for NFS configuration
files.)
# mkdir -p /global/nfs/SUNW.nfs

(Create the dfstab.resource file under the nfs/SUNW.nfs directory and set share
options.)

(Register the NFS Resource Type.)
# scrgadm -a -t SUNW.nfs

(Create the NFS resource in the resource group.)
# scrgadm -a -j r-nfs -g resource-group-1 -t SUNW.nfs

(Enable the resources and their monitors, manage the resource group, and switch
the resource group into online state.)
# scswitch -Z -g resource-group-1
```

Where to Go From Here

See "How to Change Share Options on an NFS File System" on page 130 to set share options for your NFS file systems. See "Configuring Sun Cluster HA for NFS Extension Properties" on page 132 to review or set extension properties.

How to Change Share Options on an NFS File System

If you use the `rw`, `rw=`, `ro`, or `ro=` options to the `share -o` command, NFS fault monitoring works best if you grant access to all the physical hosts or `netgroups` associated with all Sun Cluster servers.

If you use `netgroups` in the `share`(1M) command, add all of the Sun Cluster hostnames to the appropriate `netgroup`. Ideally, grant both read and write access to all of the Sun Cluster hostnames to enable the NFS fault probes to do a complete job.

Note – Before you change share options, read the `share_nfs`(1M) man page to understand which combinations of options are legal.

---

1. **Become superuser on a cluster node.**
2. **Turn off fault monitoring on the NFS resource.**

```
# scswitch -n -M -j resource
```

`-M`	Disables the resource monitor.

3. **Execute your proposed new `share` command.**

Before you edit the `dfstab.`*resource* file with new share options, execute the new `share` command to verify the validity of your combination of options.

```
# share -F nfs [-o] specific_options [-d "description"] pathname
```

`-F nfs`	Identifies the file system type as NFS.
`-o` *specific_options*	Specifies an option. You might use `rw`, which grants read-write access to all clients.
`-d` *description*	Describes the file system being added.
pathname	Identifies the file system being shared.

If the new `share` command fails, immediately execute another `share` command with the old options. When the new command executes successfully, proceed to Step 4.

4. **Edit the `dfstab.`*resource* file with the new share options.**

   The format of this file is exactly the same as the format that is used in the `/etc/dfs/dfstab` file. Each line consists of a `share` command.

5. **(Optional) If you are removing a path from the `dfstab.`*resource* file, execute the `unshare`(1M) command, then remove the `share` command for the path from the `dfstab.`*resource* file.**

```
# unshare [-F nfs] [-o rw] pathname
# vi dfstab.resource
```

`-F nfs`	Identifies the file system type as NFS.
`-o` *options*	Specifies the options that are specific to NFS file systems.
pathname	Identifies the file system being made unavailable.

6. **(Optional) If you are adding a path to or changing an existing path in the `dfstab.`*resource* file, verify that the mount point is valid, then perform Step 3 and Step 4.**

7. **Enable fault monitoring on the NFS resource.**

```
# scswitch -e -M -j resource
```

▼ How to Tune Sun Cluster HA for NFS Method Timeouts

The time Sun Cluster HA for NFS methods take to finish depends on the number of paths that the resources share through the `dfstab.`*resource* file. The default timeout for these methods is 300 seconds. A simple rule of thumb is to allocate 10 seconds toward the method time-out values for each path shared. Because the default timeouts are designed to handle 30 shared paths, if the number of shared paths is less than 30, do not reduce the timeout.

If the number of shared paths exceeds 30, however, multiply that number by 10 to compute the recommended timeout. For example, if the `dfstab.`*resource* file contains 50 shared paths, the recommended timeout is 500 seconds.

Change the following method timeouts.

`Prenet_start_timeout`	`Postnet_stop_timeout`	`Monitor_Start_timeout`
`Start_timeout`	`Validate_timeout`	`Monitor_Stop_timeout`
`Stop_timeout`	`Update_timeout`	`Monitor_Check_timeout`

To change method timeouts, use the `scrgadm -c` option, as in the following example.

```
% scrgadm -c -j resource -y Prenet_start_timeout=500
```

▼ How to Configure `SUNW.HAStorage` Resource Type

The `SUNW.HAStorage` resource type synchronizes actions between HA storage and the data service. The Sun Cluster HA for NFS data service is disk-intensive, and therefore you should set up the `SUNW.HAStorage` resource type.

See the `SUNW.HAStorage`(5) man page and "Relationship Between Resource Groups and Disk Device Groups" on page 4 for background information. See "How to Set Up `SUNW.HAStorage` Resource Type for New Resources" on page 239 for the procedure.

Configuring Sun Cluster HA for NFS Extension Properties

Typically, you use the command line `scrgadm -x` *parameter=value* to configure extension properties when you create the NFS resource. You can also use the procedures in Chapter 11 to configure them later. You are not required to set any extension properties for the Sun Cluster HA for NFS data service. See Appendix A for details of all Sun Cluster properties.

TABLE 7-2 describes extension properties that you can configure for the Sun Cluster HA for NFS data service. You can update some properties dynamically. You can update others, however, only when you create the resource. The Tunable column indicates when you can update the property.

TABLE 7-2 Sun Cluster HA for NFS Extension Properties

Name/Data Type	Default	Range	Tunable	Description
`Lockd_nullrpc_timeout` (integer)	`120`	Minimum = `60`	Any time	The time-out value (in seconds) to use when probing `lockd`.
`Monitor_retry_count` (integer)	`4`	`0` – `2,147,483,641` `-1` indicates an infinite number of restart attempts.	Any time	The number of times that the process monitor facility (PMF) restarts the fault monitor during the time window that the `Monitor_retry_interval` property specifies. Note that this property refers to restarts of the fault monitor itself, rather than the resource. The system-defined properties `Retry_interval` and `Retry_count` control restarts of the resource. See the `scrgadm`(1M) man page for a description of these properties.
`Monitor_retry_interval` (integer)	`2`	`0` – `2,147,483,641` `-1` indicates an infinite amount of time.	Any time	The time (in minutes) over which failures of the fault monitor are counted. If the number of times that the fault monitor fails is more than the value that is specified in the extension property `Monitor_retry_count` within this period, the PMF restarts the fault monitor.
`Mountd_nullrpc_restart` (Boolean)	`True`	None	Any time	A Boolean to indicate whether to restart `mountd` when a null `rpc` call fails.
`Mountd_nullrpc_timeout` (integer)	`120`	Minimum = `60`	Any time	The time-out value (in seconds) to use when probing `mountd`.
`Nfsd_nullrpc_restart` (Boolean)	`True`	None	Any time	A Boolean to indicate whether to restart `nfsd` when a null `rpc` call fails.
`Nfsd_nullrpc_timeout` (integer)	`120`	Minimum = `60`	Any time	The time-out value (in seconds) to use when probing `nfsd`.

TABLE 7-2 Sun Cluster HA for NFS Extension Properties *(Continued)*

Name/Data Type	Default	Range	Tunable	Description
`Rpcbind_nullrpc_reboot` (Boolean)	`True`	None	Any time	A Boolean to indicate whether to reboot the system when a null `rpc` call on `rpcbind` fails.
`Rpcbind_nullrpc_timeout` (integer)	`120`	Minimum = `60`	Any time	The time-out value (in seconds) to use when probing `rpcbind`.
`Statd_nullrpc_timeout` (integer)	`120`	Minimum = `60`	Any time	The time-out value (in seconds) to use when probing `statd`.

Sun Cluster HA for NFS Fault Monitor

The Sun Cluster HA for NFS fault monitor involves the following two processes.

- NFS system fault monitoring, which involves monitoring the NFS daemons (`nfsd`, `mountd`, `statd`, and `mountd`) and taking appropriate action when they have problems.
- Status check, which is specific to each NFS resource. The fault monitor of each resource checks the status of each shared path to monitor the file systems that the resource exports.

Fault Monitor Startup

An NFS resource start method starts the NFS system fault monitor. This start method first checks if the NFS system fault monitor (`nfs_daemons_probe`) is already running under the process monitor `pmfadm`. If not, the start method starts the `nfs_daemons_probe` process under the control of the process monitor. It then starts the resource fault monitor (`nfs_probe`), also under the control of the process monitor.

Fault Monitor Stops

The NFS resource `Monitor_stop` method stops the resource fault monitor. This method also stops the NFS system fault monitor if no other NFS resource fault monitor runs on the local node.

NFS System Fault Monitor Process

To check for the presence of the process and its response to a null rpc call, the system fault monitor probes rpcbind, statd, lockd, nfsd, and mountd. This monitor uses the following NFS extension properties.

Rpcbind_nullrpc_timeout	Lockd_nullrpc_timeout
Nfsd_nullrpc_timeout	Rpcbind_nullrpc_reboot
Mountd_nullrpc_timeout	Nfsd_nullrpc_restart
Statd_nullrpc_timeout	Mountd_nullrpc_restart

See "Configuring Sun Cluster HA for NFS Extension Properties" on page 132 to review or set extension properties.

Each system fault monitor probe cycle performs the following steps.

1. Sleeps for Cheap_probe_interval.
2. Probes rpcbind.

   If the process fails, reboots the system if Failover_mode=HARD.

   If a null rpc call fails and if Rpcbind_nullrpc_reboot=True and Failover_mode=HARD, reboots the system.
3. Probes statd and lockd.

   If either of these daemons fails, restarts both.

   If a null rpc call fails, logs a message to syslog but does not restart.
4. Probe nfsd and mountd.

   If the process fails, restart it.

   If a null rpc call fails, restart mountd if the cluster file system device is available and the extension property Mountd_nullrpc_restart=True.

If any of the NFS daemons fail to restart, the status of all online NFS resources is set to FAULTED. When all NFS daemons are restarted and healthy, the resource status is set to ONLINE again.

NFS Resource Monitor Process

Before starting the resource monitor probes, all shared paths are read from the dfstab file and stored in memory. In each probe cycle, all shared paths are probed in each iteration by performing stat() of the path.

Each resource monitor fault probe performs the following steps.

1. Sleeps for `Thorough_probe_interval`.
2. Refreshes the memory if `dfstab` has been changed since the last read.
3. Probes all shared paths in each iteration by preforming `stat()` of the path.

If any path is not functional, the resource status is set to `FAULTED`. If all paths are functional, the resource status is set to `ONLINE` again.

CHAPTER 8

Installing and Configuring Sun Cluster HA for Oracle Parallel Server

This chapter describes the steps to install and configure the Sun Cluster HA for Oracle Parallel Server (OPS) data service on your Sun Cluster servers. This chapter contains the following procedures.

- "How to Use VxVM" on page 140
- "How to Use Sun StorEdge A3500/A3500FC Disk Arrays With Hardware RAID Support" on page 141
- "How to Install Sun Cluster HA for Oracle Parallel Server Packages" on page 143
- "How to Prepare the Sun Cluster Nodes" on page 144
- "How to Install the UDLM Software" on page 146
- "How to Install the Oracle RDBMS Software" on page 147

Overview

Use the procedures in the Oracle documentation to install and configure OPS. Although OPS is not registered with or managed by the Sun Cluster Resource Group Manager (RGM), OPS depends on the RGM to query cluster information.

You can configure OPS to use the shared-disk architecture of the Sun Cluster software. In this configuration, a single database is shared among multiple instances of OPS that access the database concurrently. The Oracle UNIX Distributed Lock Manager (UDLM) controls conflicting access to the same data. If a process or a node crashes, the UDLM is reconfigured to recover from the failure.

If a node failure occurs in an OPS environment, you can configure Oracle clients to reconnect to the surviving server without the use of the IP failover that Sun Cluster failover data services use. The *Sun Cluster 3.0 U1 Concepts* document describes this failover process.

In an OPS environment, multiple Oracle instances cooperate to provide access to the same shared database. The Oracle clients can use any of the instances to access the database. Thus, if one or more instances have failed, clients can connect to a surviving instance and continue to access the database.

Installing and Configuring Sun Cluster HA for Oracle Parallel Server

The following table lists the sections that describe the installation and configuration tasks.

TABLE 8-1 Task Map: Installing and Configuring Sun Cluster HA for Oracle Parallel Server

Task	For Instructions, Go To …
(Optional) Install volume management software	"Installing Volume Management Software With Sun Cluster HA for Oracle Parallel Server" on page 140
Install Sun Cluster HA for Oracle Parallel Server packages	"Installing Sun Cluster HA for Oracle Parallel Server Packages" on page 143
Install the UNIX Distributed Lock Manager and Oracle software	"Installing the Oracle Software" on page 144

Installing Volume Management Software With Sun Cluster HA for Oracle Parallel Server

For Sun Cluster HA for Oracle Parallel Server disks, you can use the following configurations.

- VERITAS Volume Manager (VxVM) cluster utility
- Sun StorEdge™ A3500/A3500FC disk arrays with hardware RAID support

▼ How to Use VxVM

To use the VxVM software with the Sun Cluster HA for Oracle Parallel Server data service, perform the following tasks.

1. **Obtain a license for the Volume Manager cluster feature in addition to the basic VxVM license.**

   See your VxVM documentation for more information about VxVM licensing requirements.

   Caution – Failure to correctly install the license for the Volume Manager cluster feature might result in a panic when you install OPS support. Prior to installing the OPS packages, run the `vxlicense` check command to ensure that you have installed a valid license for the Volume Manager cluster feature.

   ---

2. **Install and configure the VxVM software on the cluster nodes.**

   See the VxVM appendix in the *Sun Cluster 3.0 U1 Installation Guide* and the VxVM documentation for more information.

▼ How to Use Sun StorEdge A3500/A3500FC Disk Arrays With Hardware RAID Support

If you use StorEdge A3500/A3500FC disk arrays with hardware RAID support and without VxVM software, configure raw device IDs (`/dev/did/rdsk*`) on top of the disk arrays' logical unit numbers (LUNs).

To set up the raw devices for OPS on a cluster that uses StorEdge A3500/A3500FC disk arrays with hardware RAID, perform the following steps.

1. **Create LUNs on the disk arrays.**

   See the *Sun Cluster 3.0 U1 Hardware Guide* for information on how to create LUNs.

2. **After you create the LUNs, run the `format`(1M) command to partition the disk arrays' LUNs into as many slices as you need.**

   The following example lists output from the `format` command.

```
# format

0. c0t2d0 <SUN18G cyl 7506 alt 2 hd 19 sec 248>
   /sbus@3,0/SUNW,fas@3,8800000/sd@2,0
1. c0t3d0 <SUN18G cyl 7506 alt 2 hd 19 sec 248>
   /sbus@3,0/SUNW,fas@3,8800000/sd@3,0
2. c1t5d0 <Symbios-StorEDGEA3000-0301 cyl 21541 alt 2 hd 64 sec 64>
   /pseudo/rdnexus@1/rdriver@5,0
3. c1t5d1 <Symbios-StorEDGEA3000-0301 cyl 21541 alt 2 hd 64 sec 64>
   /pseudo/rdnexus@1/rdriver@5,1
4. c2t5d0 <Symbios-StorEDGEA3000-0301 cyl 21541 alt 2 hd 64 sec 64>
   /pseudo/rdnexus@2/rdriver@5,0
5. c2t5d1 <Symbios-StorEDGEA3000-0301 cyl 21541 alt 2 hd 64 sec 64>
   /pseudo/rdnexus@2/rdriver@5,1
6. c3t4d2 <Symbios-StorEDGEA3000-0301 cyl 21541 alt 2 hd 64 sec 64>
   /pseudo/rdnexus@3/rdriver@4,2
```

Note – If you use slice 0, do not start the partition at cylinder 0.

---

3. **Run the** `scdidadm`**(1M) command to find the raw device ID (DID) that corresponds to the LUNs that you created in Step 1.**

   The following example lists output from the `scdidadm -L` command.

```
# scdidadm -L

1        phys-schost-1:/dev/rdsk/c0t2d0   /dev/did/rdsk/d1
1        phys-schost-2:/dev/rdsk/c0t2d0   /dev/did/rdsk/d1
2        phys-schost-1:/dev/rdsk/c0t3d0   /dev/did/rdsk/d2
2        phys-schost-2:/dev/rdsk/c0t3d0   /dev/did/rdsk/d2
3        phys-schost-2:/dev/rdsk/c4t4d0   /dev/did/rdsk/d3
3        phys-schost-1:/dev/rdsk/c1t5d0   /dev/did/rdsk/d3
4        phys-schost-2:/dev/rdsk/c3t5d0   /dev/did/rdsk/d4
4        phys-schost-1:/dev/rdsk/c2t5d0   /dev/did/rdsk/d4
5        phys-schost-2:/dev/rdsk/c4t4d1   /dev/did/rdsk/d5
5        phys-schost-1:/dev/rdsk/c1t5d1   /dev/did/rdsk/d5
6        phys-schost-2:/dev/rdsk/c3t5d1   /dev/did/rdsk/d6
6        phys-schost-1:/dev/rdsk/c2t5d1   /dev/did/rdsk/d6
```

4. **Use the DID that the** `scdidadm` **output identifies to set up the raw devices.**

   For example, the `scdidadm` output might identify that the raw DID that corresponds to the disk arrays’ LUNs is `d4`. In this instance, use the `/dev/did/rdsk/d4sx` raw device, where `x` is the slice number.

Installing Sun Cluster HA for Oracle Parallel Server Packages

Use this procedure to install the packages needed to run the Sun Cluster HA for Oracle Parallel Server data service.

▼ How to Install Sun Cluster HA for Oracle Parallel Server Packages

To complete this procedure, you need the Sun Cluster Agents CD. Perform this procedure on all cluster nodes that can run Sun Cluster HA for Oracle Parallel Server.

1. **Load the Agents CD into the CD-ROM drive.**

2. **Install the Sun Cluster HA for Oracle Parallel Server packages.**

   The packages vary, depending on whether you use hardware RAID or VERITAS Volume Manager for your volume manager.

   a. **If you use hardware RAID as your volume manager, install as follows.**

   ```
   # pkgadd -d . SUNWscucm SUNWudlm SUNWudlmr SUNWschwr
   ```

   b. **If you use VERITAS Volume Manager as your volume manager, install as follows.**

   ```
   # pkgadd -d . SUNWscucm SUNWudlm SUNWudlmr SUNWcvmr SUNWcvm
   ```

Caution – After you have installed the Sun Cluster HA for Oracle Parallel Server packages, do not reboot the nodes until the Oracle UDLM package is installed, otherwise a panic occurs.

Where to Go From Here

Go to "Installing the Oracle Software" on page 144 to install the UDLM and Oracle software.

Installing the Oracle Software

Use the procedures in this section to perform the following tasks.

- Prepare the Sun Cluster nodes.
- Install the Oracle UDLM software.
- Install the Oracle RDBMS software.

▼ How to Prepare the Sun Cluster Nodes

For the UDLM software to run correctly, sufficient shared memory must be available on all cluster nodes. See the OPS CD for all installation instructions. To prepare the Sun Cluster nodes, check that the following tasks have been completed.

- The Oracle user account and the `dba` group are set up correctly.
- The system is configured to support the shared memory requirements of the UDLM.

Note – Perform the following steps as superuser on each cluster node.

---

1. **On each node, create an entry for the database administrator group in the `/etc/group` file, and add potential users to the group.**

   This group normally is named `dba`. Verify that `root` and `oracle` are members of the `dba` group, and add entries as necessary for other DBA users. Verify that the group IDs are the same on all the nodes that run the Sun Cluster HA for Oracle Parallel Server data service. For example, add the following entry to the `/etc/group` file.

   ```
   dba:*:520:root,oracle
   ```

   You can make the name-service entries in a network name service (for example, NIS or NIS+) so that the information is available to Sun Cluster HA for Oracle Parallel Server clients. You can also make entries in the local `/etc` files to eliminate dependency on the network name service.

2. **On each node, create an entry for the Oracle user ID (the group and password) in the `/etc/passwd` file, and run the `pwconv`(1M) command to create an entry in the `/etc/shadow` file.**

   This Oracle user ID is normally `oracle`. For example, add the following entry to the `/etc/passwd` file.

   ```
   # useradd -u 120 -g dba -d /Oracle-home oracle
   ```

   Ensure that the user IDs are the same on all the nodes that run the Sun Cluster HA for Oracle Parallel Server data service.

Where to Go From Here

After you set up the cluster environment for OPS, install the UDLM software on each cluster node. See your OPS installation documentation for instructions.

▼ How to Install the UDLM Software

You must install the UDLM software on the local disk of each node.

1. **Become superuser on a cluster member.**
2. **Install the UDLM software.**

   See the appropriate OPS installation documentation.
3. **Update `/etc/system` with the shared memory configuration information.**

   You must configure these parameters based on the resources available in the cluster. Decide on the appropriate values, but be sure that the UDLM can create a shared memory segment according to its configuration requirements. The following is an example of the entries to configure in `/etc/system`.

   ```
   *SHARED MEMORY/ORACLE
   set shmsys:shminfo_shmmax=268435456
   set semsys:seminfo_semmap=1024
   set semsys:seminfo_semmni=2048
   set semsys:seminfo_semmns=2048
   set semsys:seminfo_semmsl=2048
   set semsys:seminfo_semmnu=2048
   set semsys:seminfo_semume=200
   set shmsys:shminfo_shmmin=200
   set shmsys:shminfo_shmmni=200
   set shmsys:shminfo_shmseg=200
   forceload: sys/shmsys
   forceload: sys/semsys
   forceload: sys/msgsys
   ```
4. **Shut down and reboot all the nodes.**

Caution – Before you reboot, ensure that the VxVM software is installed correctly and the license for cluster operation is valid. Also ensure that the UDLM software is installed and configured correctly. Otherwise, a panic occurs.

See the `scshutdown`(1M) man page for details.

First, run the following command on one node to shut down all of the nodes.

```
phys-schost-1# scshutdown -g0 -y
```

For each node, run the following command at the `ok` prompt.

```
ok boot
```

Where to Go From Here

After you have installed the UDLM software on each cluster node and rebooted all the nodes, install the Oracle RDBMS software. See the OPS installation documentation for instructions.

▼ How to Install the Oracle RDBMS Software

See your OPS installation documentation for instructions on installing the RDBMS software.

Where to Go From Here

See the instructions in the Oracle documentation to create your Oracle database when you install the Oracle RDBMS software.

CHAPTER 9

Installing and Configuring Sun Cluster HA for SAP

This chapter provides instructions on how to plan, set up, and configure the Sun Cluster HA for SAP data service on your Sun Cluster nodes.

This chapter includes the following procedures.

Installing and Configuring Sun Cluster HA for SAP

The following table lists the sections that describe the installation and configuration tasks.

TABLE 9-1 Task Map: Installing and Configuring Sun Cluster HA for SAP

Task	For Instructions, Go To
Plan the SAP installation	"Sun Cluster HA for SAP Overview" on page 151 "Configuration Guidelines for Sun Cluster HA for SAP" on page 151 "Sample Configurations" on page 152 "Pre-Installation Considerations" on page 153
Install and configure SAP and the database	"How to Install SAP and the Database" on page 155 "How to Enable SAP to Run in the Cluster" on page 155 "How to Verify SAP and Database Installation With Central Instance" on page 157 "How to Verify SAP and Database Installation With Application Server" on page 158
Configure the Sun Cluster HA for DBMS	"Configuring Sun Cluster HA for DBMS" on page 160
Configure the Sun Cluster HA for SAP data service	"How to Register and Configure Sun Cluster HA for SAP With Central Instance" on page 160 "How to Register and Configure Sun Cluster HA for SAP With Application Server" on page 161 "How to Verify the Sun Cluster HA for SAP Installation With Central Instance" on page 162 "How to Verify the Sun Cluster HA for SAP Installation With Application Server" on page 163
Configure SAP extension properties	"Configuring SAP Extension Properties" on page 164
View Sun Cluster HA for SAP fault-monitor information	"Sun Cluster HA for SAP Fault Monitor" on page 167

Sun Cluster HA for SAP Overview

The Sun Cluster HA for SAP data service provides fault monitoring and an automatic failover mechanism for the SAP application to eliminate single points of failure in an SAP system. The following table lists the data services that best protect SAP components in a Sun Cluster configuration.

TABLE 9-2 Protection of SAP Components

SAP Component	Protected by
SAP database	Sun Cluster HA for Oracle, if the database is Oracle
SAP central instance	Sun Cluster HA for SAP, the resource type is `SUNW.sap_ci`
SAP application server	Sun Cluster HA for SAP, the resource type is `SUNW.sap_as`
NFS file system	Sun Cluster HA for NFS

Use the `scinstall`(1M) command to install the Sun Cluster HA for SAP data service. The Sun Cluster HA for SAP data service requires a functioning cluster with the initial cluster framework already installed. See the *Sun Cluster 3.0 U1 Installation Guide* for details about initial installation of clusters and data services. Register the Sun Cluster HA for SAP data service after you successfully install the basic components of the Sun Cluster and SAP software.

Configuration Guidelines for Sun Cluster HA for SAP

When you design a Sun Cluster HA for SAP configuration, consider the following guidelines.

- **Use an SAP software version that is qualified with Sun Cluster 3.0 –** The Solaris 8 operating environment offers support for the Sun Cluster software.
- **Use an SAP software version with automatic enqueue-reconnect-mechanism capability –** The Sun Cluster HA for SAP data service relies on this capability. SAP 4.0 software with patch information and later releases should have automatic enqueue-reconnect-mechanism capability.
- **Read all related SAP online service system notes for the SAP software release and database you are installing on your Solaris platform –** Identify any known installation problems and fixes.

- **Consult SAP software documentation for memory and swap recommendations –** SAP software uses a large amount of memory and swap space.
- **Generously estimate the total possible load on nodes that might host the central instance, the database instance, and the application server, if you have an internal application server –** This guideline is especially important if you configure the cluster so that the central instance, database instance, and application server will all exist on one node if failover occurs.
- **Install application servers on the same cluster that hosts the central instance or on a separate cluster –** If you install and configure any application server outside of the cluster environment, the Sun Cluster HA for SAP data service does not fault-monitor and does not automatically restart or fail over those application servers. You must manually start and shut down application servers that you installed and configured outside of the cluster environment.
- **Limit node names and logical hostnames to eight characters or less –** This limitation is an SAP software requirement.

Sample Configurations

See your Enterprise Services representative for the most current information about supported SAP versions. The following figures illustrate sample configurations for the Sun Cluster HA for SAP data service.

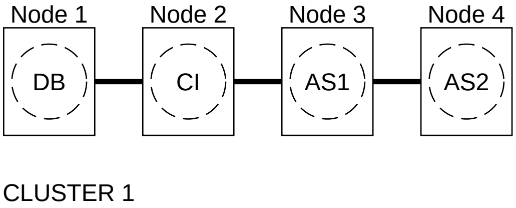

FIGURE 9-1 Four-Node Cluster With Central Instance, Application Servers, and Database

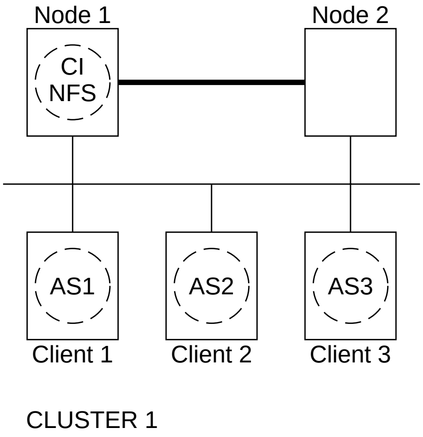

FIGURE 9-2 Two-Node Cluster With Central Instance, NFS, and Non-HA External Application Servers

---

Note – This figure was a common configuration under previous Sun Cluster releases. To use the Sun Cluster 3.0 software to the full extent, follow FIGURE 9-1 or FIGURE 9-3.

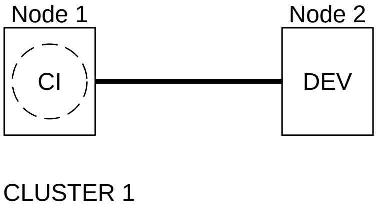

FIGURE 9-3 Two-Node Cluster With Central Instance and Development Node

Pre-Installation Considerations

Before installing the SAP software, see "Installing and Configuring SAP and the Database" on page 154, and consider the following cluster-related issues.

- **Install SAP binaries and SAP users' home directories –** Install SAP binaries and users' home directories on the cluster file system. Installation on the cluster file system, however, has some drawbacks with SAP software release upgrades. See "Determining the Location of the Application Binaries" on page 3 for information about drawbacks.
- **After you create all the file systems for the database and for SAP, create the mount points, and put the mount points in the `/etc/vfstab` file on all cluster nodes –** See the SAP installation guides, *Installation of the SAP R/3 on UNIX* and *R/3 Installation on UNIX-OS Dependencies,* for details on how to set up the database and SAP file systems.
- **Create the required groups and users on all cluster nodes –** Create the required groups and users for the SAP software on all cluster nodes according to the SAP installation guides, *Installation of the SAP R/3 on UNIX* and *R/3 Installation on UNIX-OS Dependencies.*
- **Configure the Sun Cluster HA for NFS data service on the cluster that hosts the central instance if you plan to install some external SAP application servers –** See "Installing and Configuring Sun Cluster HA for NFS" on page 122 for details on how to configure the Sun Cluster HA for NFS data service.
- **Set up the `/etc/nsswitch.conf` file so that the data service starts and stops correctly during switchovers or failovers –** On each node that can master the logical host that runs the Sun Cluster HA for SAP data service, the `/etc/nsswitch.conf` file must have one of the following entries for `group`.

```
group:
group:  files
group:  files [NOTFOUND=return] nis
group:  files [NOTFOUND=return] nisplus
```

The Sun Cluster HA for SAP data service uses the `su` *user* command to start and stop the database node. The network information name service might become unavailable when a cluster node's public network fails. Adding the preceding entries for `group` ensures that the `su`(1M) command does not refer to the NIS/NIS+ name services if this unavailability occurs.

Installing and Configuring SAP and the Database

Use the procedures in this section to perform the following tasks.

- Install SAP and the database.
- Enable SAP to run in the cluster.

- Verify SAP and database installation.

▼ How to Install SAP and the Database

This section describes how to install and configure SAP and the database and how to enable SAP to run in the cluster.

1. **Become superuser on one of the nodes in the cluster where you are installing the central instance.**

2. **Install SAP binaries on the cluster file system.**

Note – Before you install SAP software on the cluster file system, use the `scstat`(1M) command to verify that the Sun Cluster software is fully operational.

---

   a. **For all SAP-required kernel parameter changes, edit the `/etc/system` file on all cluster nodes that will run the SAP application.**

   After you edit the `/etc/system` file, reboot each node. See the SAP document *R/3 Installation on UNIX-OS Dependencies* for details on kernel parameter changes.

   b. **See the SAP document** *Installation of the SAP R/3 on UNIX* **for details on how to install the central instance and the database.**

Where to Go From Here

After you install SAP and the database, go to "How to Enable SAP to Run in the Cluster" on page 155.

▼ How to Enable SAP to Run in the Cluster

During SAP installation, the SAP software creates files and shell scripts on the server on which you installed the SAP central instance. These files and scripts use physical-server names. To run the SAP software with Sun Cluster software, replace references to a physical server with references to a logical hostname. Throughout these steps, the replaceable term *physicalserver* represents a physical server, and the replaceable term *logical-hostname* represents a logical hostname. The phrase *logical hostname* denotes the logical hostname where traffic between the database and the application server occurs. See the *Sun Cluster 3.0 U1 Concepts* for more information on logical hostnames.

Perform the following steps to enable SAP to run in the cluster.

Note – Make backup copies of the files that you will modify in the following steps.

---

1. **Log in to the node on which you installed the SAP software.**

2. **Shut down the SAP central instance and the database.**

Note – In addition to the central instance and database, shut down any application servers that are running.

---

3. **Modify all file names that include a physical-server name in the following directories.**
   - **The** *sapsid*`adm` **home directory –** Become the *sapsid*`adm` user before you edit these files.
   - **The** `ora`*sapsid* **home directory –** Become the `ora`*sapsid* user before you edit these files.
   - **SAP profile directory –** Become the *sapsid*`adm` user before you edit these files.

   For example, rename the `.sapenv_`*physicalserver*`.csh` file as `.sapenv_`*logical-hostname*`.csh`.

4. **Modify all file contents—except log file contents—that reference a physical-server name in the following directories.**
   - **The** *sapsid*`adm` **home directory –** Become the *sapsid*`adm` user before you edit these files.
   - **The** `ora`*sapsid* **home directory –** Become the `ora`*sapsid* user before you edit these files.
   - **SAP profile directory –** Become the *sapsid*`adm` user before you edit these files.

   For example, change any *physicalserver* reference in the startup and shutdown scripts to a logical-hostname reference.

5. **As user** *sapsid*`adm`**, add an entry such as the following example for the parameter** `SAPLOCALHOST`**.**

```
SAPLOCALHOST logical-hostname
```

   Add this entry to the *SAPSID_Service-StringSystem-Number_logical-hostname* profile file under the `/sapmnt/`*SAPSID*`/profile` directory.

   This entry enables the external application server to locate the central instance by using the logical hostname.

Where to Go From Here

After you enable SAP to run in the cluster, go to "How to Verify SAP and Database Installation With Central Instance" on page 157.

How to Verify SAP and Database Installation With Central Instance

Perform this procedure to test starting and stopping the SAP central instance on all potential nodes on which the central instance can run.

1. **Create the failover resource group to hold the network logical-hostname and central-instance resources.**

```
# scrgadm -a -g sap-ci-resource-group
```

Note – You can optionally select the set of nodes on which the SAP central instance can run with the `-h` option to the `scrgadm`(1M) command.

```
# scrgadm -a -g sap-ci-resource-group -h nodelist
```

2. **Verify that you have added all the logical hostnames that you use to your name-service database.**

3. **Run the `scrgadm` command to add a logical hostname to the failover resource group.**

```
# scrgadm -a -L -g sap-ci-resource-group -l logical-hostname -n
nafo0@node1,nafo0@node2
```

4. **Enable the resource group.**

   Run the `scswitch`(1M) command to move the resource group into a managed state and bring the resource group online.

```
# scswitch -Z -g sap-ci-resource-group
```

5. **Log in to the cluster member that hosts the central-instance resource group.**

6. **Start the central instance and the database.**

7. **Start the SAP GUI to verify that SAP initializes correctly.**

   The default dispatcher port is `3200`.

8. **Stop the central instance and the database.**

9. **Run the `scswitch` command.**

   In the following example, the replaceable term *sap-ci-resource-group* represents the resource group that contains the logical-hostname resource for the central-instance resource. Switch this resource group to another cluster member that can host the central instance.

   ```
   # scswitch -z -h node -g sap-ci-resource-group
   ```

10. **Repeat Step 5 through Step 7 until you verify startup and shutdown of the central instance on each cluster node that can host the central instance.**

Where to Go From Here

After you verify SAP and database installation with central instance, go to "How to Verify the Sun Cluster HA for SAP Installation With Application Server" on page 163.

▼ How to Verify SAP and Database Installation With Application Server

If you have installed and configured any application servers, perform this procedure on all potential nodes on which the application server can run. This procedure tests starting and stopping the application server.

1. **Create the failover resource group to hold the network logical-hostname and application-server resources.**

   ```
   # scrgadm -a -g sap-as-resource-group
   ```

Note – You can optionally select the set of nodes on which the SAP application server can run with the `-h` option to the `scrgadm` command.

```
# scrgadm -a -g sap-as-resource-group -h nodelist -n nafo0@node1,nafo0@node2
```

2. **Verify that you have added all the logical hostnames that you use to your name-service database.**

3. **Run the `scrgadm` command to add a logical hostname to the failover resource group.**

   ```
   # scrgadm -a -L -g sap-as-resource-group -l logical-hostname
   ```

4. **Enable the resource group.**

   Run the `scswitch`(1M) command to move the resource group into a managed state and bring the resource group online.

   ```
   # scswitch -Z -g sap-as-resource-group
   ```

5. **Log in to the cluster member that hosts the application-server resource group.**

6. **Start the application server.**

7. **Start the SAP GUI to verify that the SAP application server initializes correctly.**

8. **Stop the application server.**

9. **Run the `scswitch` command.**

   In the following example, the term *sap-as-resource-group* represents the resource group that contains the logical-hostname resource for the application-server resource. Switch this resource group to another cluster member that can host the application server.

   ```
   # scswitch -z -h node -g sap-as-resource-group
   ```

10. **Repeat Step 5 through Step 7 until you verify startup and shutdown of the application server on each cluster node that can host the application server.**

Where to Go From Here

After you finish all procedures to install and configure SAP and the database, go to "How to Register and Configure Sun Cluster HA for SAP With Central Instance" on page 160.

Configuring Sun Cluster HA for DBMS

SAP supports various databases. See the appropriate chapter of this book for details on how to configure the resource type, resource group, and resource for your highly available database. For example, see "Installing and Configuring Sun Cluster HA for Oracle" on page 16 for more information if you plan to use Oracle with SAP.

Additionally, see the appropriate chapter of this book and the appropriate chapter of your database installation book for details on other resource types to configure with your database. This book includes details on how to configure other resource types for Oracle databases. For instance, set up the resource type `SUNW.HAStorage` if you use Oracle. See the procedure "How to Configure `SUNW.HAStorage` Resource Type" on page 33 for more information.

Registering and Configuring Sun Cluster HA for SAP

Use the procedures in this section to perform the following tasks.

- Register and configure the Sun Cluster HA for SAP data service.
- Verify the Sun Cluster HA for SAP installation.

▼ How to Register and Configure Sun Cluster HA for SAP With Central Instance

To register and configure the Sun Cluster HA for SAP data service with a central instance, perform the following steps.

1. **Become superuser on one of the nodes in the cluster that hosts the central instance.**

2. **Register the resource type for the SAP data service.**

   For central instance, run the `scrgadm` command to register the resource type `SUNW.sap_ci`.

```
# scrgadm -a -t SUNW.sap_ci
```

3. **Run the `scrgadm` command to create SAP application resources in this failover resource group.**

```
# scrgadm -a -j sap-ci-resource -g sap-ci-resource-group -t SUNW.sap_ci
-x SAPSID=SAPSID
-x Ci_startup_script=ci-startup-script
-x Ci_shutdown_script=ci-shutdown-script
```

4. **Run the `scswitch` command to enable the resource group that now includes the SAP central-instance resource.**

```
# scswitch -Z -g sap-ci-resource-group
```

Where to Go From Here

After you register and configure Sun Cluster HA for SAP with central instance, go to "How to Register and Configure Sun Cluster HA for SAP With Application Server" on page 161.

▼ How to Register and Configure Sun Cluster HA for SAP With Application Server

To register and configure the Sun Cluster HA for SAP data service with an application server, perform the following steps.

1. **Become superuser on one of the nodes in the cluster that hosts the application server.**

2. **Register the resource type for the SAP data server.**

   For application server, run the scrgadm command to register the resource type SUNW.sap_as.

   ```
   # scrgadm -a -t SUNW.sap_as
   ```

3. **Run the scrgadm command to create SAP application server resources in this failover resource group.**

   ```
   # scrgadm -a -j sap-as-resource -g sap-as-resource-group -t SUNW.sap_as
   -x SAPSID=SAPSID
   -x As_instance_id=as-instance-id
   -x As_startup_script=as-startup-script
   -x As_shutdown_script=as-shutdown-script
   ```

4. **Run the scswitch command to enable the resource group that now includes the SAP application-server resource.**

   ```
   # scswitch -Z -g sap-as-resource-group
   ```

Where to Go From Here

After you register and configure Sun Cluster HA for SAP with application server, go to "How to Verify the Sun Cluster HA for SAP Installation With Central Instance" on page 162.

▼ How to Verify the Sun Cluster HA for SAP Installation With Central Instance

Perform the following steps to verify both the Sun Cluster HA for SAP installation with a central instance and the Sun Cluster HA for DBMS installation and configuration.

1. **Log in to the node that hosts the resource group that contains the SAP central-instance resource.**

2. **Become user *sapsid*adm.**

3. **Start the SAP GUI to check that the Sun Cluster HA for SAP data service is functioning correctly.**

4. **Use the central-instance `stopsap` script to shut down the SAP central instance.**

   The Sun Cluster software should restart the central instance because the Sun Cluster software controls the SAP software.

5. **Run the `scswitch` command to switch the SAP resource group to another cluster member.**

   ```
   # scswitch -z -h node2 -g sap-ci-resource-group
   ```

6. **Verify that the SAP central instance starts on this node.**

7. **Repeat Step 1 through Step 6 until you have tested all potential nodes on which the SAP central instance can run.**

Where to Go From Here

After you verify the Sun Cluster HA for SAP installation with central instance, go to "How to Verify the Sun Cluster HA for SAP Installation With Application Server" on page 163.

How to Verify the Sun Cluster HA for SAP Installation With Application Server

If you have installed and configured any SAP application servers, perform the following steps to verify the Sun Cluster HA for SAP installation and configuration with application servers.

1. **Log in to the node that currently hosts the resource group that contains the SAP application-server resource.**

2. **Become user *sapsid*`adm`.**

3. **Start the SAP GUI to check that the Sun Cluster HA for SAP data service is functioning correctly.**

4. **Use the application-server `stopsap` script to shut down the SAP application server.**

   The Sun Cluster software should restart the application server because the Sun Cluster software controls the SAP software.

5. **Run the scswitch command to switch the resource group that contains the SAP application-server resource to another cluster member.**

```
# scswitch -z -h node2 -g sap-as-resource-group
```

6. **Verify that the SAP application server starts on this node.**

7. **Repeat Step 1 through Step 6 until you have tested all potential nodes on which the SAP application server can run.**

Configuring SAP Extension Properties

This section describes how to configure Sun Cluster HA for SAP extension properties for the central instance and application servers. Typically, you use the command line `scrgadm -x` *parameter=value* to configure the extension properties when you create the central-instance or application resource. You can also use the procedures described in Chapter 11 to configure them later.

See the `r_properties`(5) and the `rg_properties`(5) man pages for details on all Sun Cluster extension properties.

TABLE 9-3 describes SAP extension properties you can set for the central instance. You can update some extension properties dynamically. You can update others, however, only when you create or disable the SAP resource. The Tunable column in the following table indicates when you can update each property.

TABLE 9-3 Sun Cluster HA for SAP Extension Properties for Central Instance

Property Category	Property Name	Default	Tunable	Description
SAP Configuration	SAPSID	None	When disabled	SAP system name or *SAPSID*.
	Ci_instance_id	00	When disabled	Two-digit SAP system number.
	Ci_services_string	DVEBMGS	When disabled	String of central-instance services.

TABLE 9-3 Sun Cluster HA for SAP Extension Properties for Central Instance

Property Category	Property Name	Default	Tunable	Description
Starting SAP	`Ci_start_retry_interval`	`30`	When disabled	The interval in seconds to wait between attempting to connect to the database before starting the central instance.
	`Ci_startup_script`	None	When disabled	Name of the SAP startup script for this instance in your *SID*`adm` home directory.
Stopping SAP	`Stop_sap_pct`	`95`	When disabled	Percentage of stop-timeout variables that are used to stop SAP processes. The SAP shutdown script is used to stop processes before calling Process Monitor Facility (PMF) to terminate and then kill the processes.
	`Ci_shutdown_script`	None	When disabled	Name of the SAP shutdown script for this instance in your *SID*`adm` home directory.
Probe	`Message_server_name`	`sapms` *SAPSID*	When disabled	The name of the SAP message server.
	`Lgtst_ms_with_logicalhostname`	`TRUE`	Any time	How to check the SAP message server with the SAP `lgtst` utility. The `lgtst` utility requires a hostname (IP address) as the location for the SAP message server. This hostname can be either a Sun Cluster logical hostname or a localhost (loopback) name. If you set this resource property to `TRUE`, use a logical hostname. Otherwise, use a localhost name.
	`Check_ms_retry`	`2`	When disabled	Maximum number of times the SAP message server check fails before a total failure is reported and the Resource Group Manager (RGM) starts.
	`Probe_timeout`	`60`	Any time	Time-out value in seconds for the probes.
	`Monitor_retry_count`	`4`	Any time	Number of PMF restarts that are allowed for the fault monitor.
	`Monitor_retry_interval`	`2`	Any time	Time interval in minutes for fault-monitor restarts.

TABLE 9-3 Sun Cluster HA for SAP Extension Properties for Central Instance

Property Category	Property Name	Default	Tunable	Description
Development System	`Shutdown_dev`	`FALSE`	When disabled	Whether the RGM should shut down the development system before starting up the central instance.
	`Dev_sapsid`	None	When disabled	SAP System Name for the development system (if you set `Shutdown_dev` to `TRUE`, the Sun Cluster HA for SAP data service requires this property).
	`Dev_shutdown_script`	None	When disabled	Script that is used to shut down the development system. If you set `Shutdown_dev` to `TRUE`, the Sun Cluster HA for SAP data service requires this property.
	`Dev_stop_pct`	`20`	When disabled	Percentage of startup timeouts the Sun Cluster HA for SAP data service uses to shut down the development system before starting the central instance.

The following table describes extension properties you can set for SAP with application servers.

TABLE 9-4 Sun Cluster HA for SAP Extension Properties for Application Server

Property Category	Property Name	Default	Tunable	Description
SAP Configuration	`SAPSID`	None	When disabled	SAP system name or *SAPSID* for the application server.
	`As_instance_id`	None	When disabled	Two-digit SAP system number for the application server.
	`As_services_string`	`D`	When disabled	String of application-server services.
Starting SAP	`As_db_retry_interval`	`30`	When disabled	The interval in seconds to wait between attempting to connect to the database and starting the application server.
	`As_startup_script`	None	When disabled	Name of the SAP startup script for the application server.

TABLE 9-4 Sun Cluster HA for SAP Extension Properties for Application Server

Property Category	Property Name	Default	Tunable	Description
Stopping SAP	`Stop_sap_pct`	95	When disabled	Percentage of stop-timeout variables that are used to stop SAP processes. The SAP shutdown script is used to stop processes before calling Process Monitor Facility (PMF) to terminate and then kill the processes.
	`As_shutdown_script`	None	When disabled	Name of the SAP shutdown script for the application server.
Probe	`Probe_timeout`	60	Any time	Time-out value in seconds for the probes.
	`Monitor_retry_count`	4	Any time	Number of PMF restarts that the probe allows for the fault monitor.
	`Monitor_retry_interval`	2	Any time	Time interval in minutes for fault-monitor restarts.

Sun Cluster HA for SAP Fault Monitor

The Sun Cluster HA for SAP fault monitor checks SAP process and database health. SAP process health impacts SAP resources' failure history. SAP resources' failure history in turn drives the fault monitor's actions, which include no action, restart, or failover.

In contrast to SAP process health, the health of the database SAP uses has no impact on SAP resources' failure history. Database health does, however, trigger the SAP fault monitor to log any `syslog` messages and to set the status accordingly for the SAP resource that uses the database.

Sun Cluster HA for SAP Fault Probes for Central Instance

For the central instance, the fault probe executes the following steps.

1. Retrieves the process IDs for the Message Server and the dispatcher.
2. Loops infinitely (sleeps for `Thorough_probe_interval`).
3. Checks the health of the SAP resources.

a. **Abnormal exit –** If the PMF detects that the SAP process tree has failed, the fault monitor treats this problem as a complete failure. The fault monitor restarts or fails over the SAP resource to another node based on the resources' failure history.

b. **Health check of the SAP resources through probe –** The probe uses the `ps`(1) command to check the SAP Message Server and main dispatcher processes. If any of the SAP Message Server or main dispatcher processes are missing from the system's active processes list, the fault monitor treats this problem as a complete failure.

   If you configure the parameter `Check_ms_retry` to have a value greater than zero, the probe checks the Message Server connection. If you have set the extension property `Lgtst_ms_with_logicalhostname` to its default value `TRUE`, the probe completes the Message Server connection test with the utility `lgtst`. The probe uses the logical hostname interface specified in the SAP resource group to call the SAP-supplied utility `lgtst`. If you set the extension property `Lgtst_ms_with_logicalhostname` to a value other than `TRUE`, the probe calls `lgtst` with the node's localhost name (loopback interface).

   If the `lgtst` utility call fails, the SAP Message Server connection is not functioning. In this situation, the fault monitor considers the problem to be a partial failure and does not trigger an SAP restart or a failover immediately. The fault monitor counts two partial failures as a complete failure if the following conditions occur.

   i. You configure the extension property `Check_ms_retry` to be `2`.

   ii. The fault monitor accumulates two partial failures that happen within the retry interval that the resource property `Retry_interval` sets.

   A complete failure triggers either a local restart or a failover, based on the resource's failure history.

c. **Database connection status through probe –** The probe calls the SAP-supplied utility `R3trans` to check the status of the database connection. Sun Cluster HA for SAP fault probes verify that SAP can connect to the database. Sun Cluster HA for SAP depends, however, on the highly available database fault probes to determine the health of the database. If the database connection status check fails, the fault monitor logs the message "Database might be down" to `syslog`. The fault monitor then sets the status of the SAP resource to `DEGRADED`. If the probe checks the status of the database again and the connection is reestablished, the fault monitor logs the message "Database is up" to `syslog` and sets the status of the SAP resource to `OK`.

4. Evaluates the failure history.

   Based on the failure history, the fault monitor completes one of the following actions.

- No action

- Local restart
- Failover

Sun Cluster HA for SAP Fault Probes for Application Server

For the application server, the fault probe executes the following steps.

1. Retrieves the process ID for the main dispatcher.

2. Loops infinitely (sleeps for `Thorough_probe_interval`).

3. Checks the health of the SAP resources.

   a. **Abnormal exit –** If the PMF detects that the SAP process tree has failed, the fault monitor treats this problem as a complete failure. The fault monitor restarts or fails over the SAP resource to another node, based on the resources' failure history.

   b. **Health check of the SAP resources through probe –** The probe uses the `ps`(1) command to check the SAP Message Server and main dispatcher processes. If the SAP main dispatcher process is missing from the system's active processes list, the fault monitor treats the problem as a complete failure.

   c. **Database connection status through probe –** The probe calls the SAP-supplied utility `R3trans` to check the status of the database connection. Sun Cluster HA for SAP fault probes verify that SAP can connect to the database. Sun Cluster HA for SAP depends, however, on the highly available database fault probes to determine the health of the database. If the database connection status check fails, the fault monitor logs the message "Database might be down" to `syslog` and sets the status of the SAP resource to `DEGRADED`. If the probe checks the status of the database again and the connection is reestablished, the fault monitor logs the message "Database is up" to `syslog`. The fault monitor then sets the status of the SAP resource to `OK`.

4. Evaluate the failure history.

   Based on the failure history, the fault monitor completes one of the following actions.

- No action
- Local restart
- Failover

CHAPTER 10

Installing and Configuring Sun Cluster HA for Sybase ASE

This chapter provides instructions on how to configure and administer the Sun Cluster HA for Sybase ASE data service on your Sun Cluster nodes.

This chapter contains the following procedures.

You must configure the Sun Cluster HA for Sybase ASE data service as a failover service. See the *Sun Cluster 3.0 U1 Concepts* document and Chapter 1 for general information about data services, resource groups, resources, and other related topics.

Installing and Configuring Sun Cluster HA for Sybase ASE

The following table lists sections that describe the installation and configuration tasks.

TABLE 10-1 Task Map: Installing and Configuring Sun Cluster HA for Sybase ASE

Task	For Instructions, Go To
Prepare to install the Sun Cluster HA for Sybase ASE data service	"Preparing to Install Sun Cluster HA for Sybase ASE" on page 173
Install the Sybase ASE 12.0 software	"Installing the Sybase ASE 12.0 Software" on page 173
Create the Sybase database environment	"Creating the Sybase ASE Database Environment" on page 178
Install the Sun Cluster HA for Sybase ASE package	"Installing the Sun Cluster HA for Sybase ASE Package" on page 181
Register Sun Cluster HA for Sybase ASE resource types and configure resource groups and resources	"Registering and Configuring Sun Cluster HA for Sybase ASE" on page 182
Verify the Sun Cluster HA for Sybase ASE installation	"Verifying the Sun Cluster HA for Sybase ASE Installation" on page 185
Understand Sun Cluster HA for Sybase ASE logging and security issues	"Understanding Sun Cluster HA for Sybase ASE Logging and Security Issues" on page 187
Configure Sun Cluster HA for Sybase ASE extension properties	"Configuring Sun Cluster HA for Sybase ASE Extension Properties" on page 188
View fault-monitor information	"Sun Cluster HA for Sybase ASE Fault Monitor" on page 190

Preparing to Install Sun Cluster HA for Sybase ASE

To prepare Sun Cluster nodes for the Sun Cluster HA for Sybase Adaptive Server 12.0 installation, select an installation location for the following files.

- **Sybase ASE application files –** These files include Sybase ASE binaries and libraries. You can install these files on either the local file system or the cluster file system.

  See "Determining the Location of the Application Binaries" on page 3 for the advantages and disadvantages of placing the Sybase ASE binaries on the local file system as opposed to the cluster file system.
- **Sybase ASE configuration files –** These files include the `interfaces` file, `config` file, and environment file. You can install these files on the local file system (with links) or on the cluster file system.
- **Database data files –** These files include Sybase device files. You must install these files on the cluster file system as either raw devices or regular files.

Installing the Sybase ASE 12.0 Software

Use the procedures in this section to complete the following tasks.

- Prepare the Sun Cluster nodes.
- Install the Sybase ASE software.
- Verify the Sybase ASE installation.

Note – Before you configure the Sun Cluster HA for Sybase ASE data service, use the procedures that the *Sun Cluster 3.0 U1 Installation Guide* describes to configure the Sun Cluster software on each node.

▼ How to Prepare the Nodes

This procedure describes how to prepare the cluster nodes for Sybase ASE software installation.

Caution – Perform all steps in this procedure on all Sun Cluster nodes. If you do not perform all steps on all nodes, the Sybase ASE installation will be incomplete, and the Sun Cluster HA for Sybase ASE data service will fail during startup.

Note – Consult the Sybase ASE documentation before you perform this procedure.

1. **Become superuser on all nodes.**

2. **Configure the `/etc/nsswitch.conf` file as follows so that the Sun Cluster HA for Sybase ASE data service starts and stops correctly if a switchover or failover occurs.**

   On each node that can master the logical host that runs the Sun Cluster HA for Sybase ASE data service, include one of the following entries for `group` in the `/etc/nsswitch.conf` file.

```
group:
group:
group: files [NOTFOUND=return] nis
group: file [NOTFOUND=return] nisplus
```

   The Sun Cluster HA for Sybase ASE data service uses the `su` *user* command to start and stop the database node. The *user* is typically `sybase_id`. The network information name service might become unavailable when a cluster node's public network fails. Adding one of the preceding entries for `group` ensures that the `su`(1M) command does not refer to the NIS/NIS+ name services if the network information name service is unavailable.

3. **Configure the cluster file system for the Sun Cluster HA for Sybase ASE data service.**

   If raw devices contain the databases, configure the global devices for raw-device access. See the *Sun Cluster 3.0 U1 Installation Guide* for information on how to configure global devices.

   If you use the Solstice DiskSuite volume manager, configure the Sybase ASE software to use UNIX file system (UFS) logging or raw-mirrored metadevices. See the Solstice DiskSuite documentation for information on how to configure raw-mirrored metadevices.

4. **Prepare the `SYBASE_HOME` directory on a local or multihost disk.**

Note – If you install the Sybase ASE binaries on a local disk, use a separate disk if possible. Installing the Sybase ASE binaries on a separate disk prevents the binaries from overwrites during operating environment reinstallation.

5. **On each node, create an entry for the database administrator (DBA) group in the `/etc/group` file, and add potential users to the group.**

   You typically name the DBA group `dba`. Verify that the `root` and `sybase_id` users are members of the `dba` group, and add entries as necessary for other DBA users. Make sure that group IDs are the same on all nodes that run the Sun Cluster HA for Sybase ASE data service, as the following example illustrates.

   ```
   dba:*:520:root,sybase_id
   ```

   You can create group entries in a network name service. If you do so, also add your entries to the local `/etc/inet/hosts` file to eliminate dependency on the network name service.

6. **On each node, create an entry for the Sybase system administrator.**

   You typically name the Sybase system administrator `sybase_id`. The following command updates the `/etc/passwd` and `/etc/shadow` files with an entry for the Sybase system administrator.

   ```
   # useradd -u 120 -g dba -d /Sybase-home sybase_id
   ```

   Make sure that the `sybase_id` user entry is the same on all nodes that run the Sun Cluster HA for Sybase ASE data service.

▼ How to Install the Sybase Software

Perform the following steps to install the Sybase ASE software.

1. **Become superuser on a cluster member.**

2. **Note the Sybase ASE installation requirements.**

   You can install Sybase ASE binaries on one of the following locations.

   - Local disks of the cluster nodes
   - Cluster file system

Note – Before you install the Sybase ASE software on the cluster file system, start the Sun Cluster software and become the owner of the disk device group.

See "Preparing to Install Sun Cluster HA for Sybase ASE" on page 173 for more information about installation locations.

3. **Create a failover resource group to hold the network and application resources.**

```
# scrgadm -a -g resource-group [-h nodelist]
```

`-g` *resource-group* Specifies the name of the resource group. This name can be your choice but must be unique for resource groups within the cluster.

`-h` *nodelist* Specifies an optional, comma-separated list of physical node names or IDs that identify potential masters. The order here determines the order in which the Resource Group Manager (RGM) considers primary nodes during failover.

Note – Use the `-h` option to specify the order of the node list. If all the nodes in the cluster are potential masters, you do not need to use the `-h` option.

4. **Verify that you have added all logical hostnames that the Sun Cluster HA for Sybase ASE data service uses to either the `/etc/inet/hosts` file or to your name-service (NIS, NIS+) database.**

5. **Add a logical hostname to the failover resource group.**

```
# scrgadm -a -L -g resource-group -l logical-hostname \
[-j resource] [-n netiflist]
```

`-l` *logical-hostname*	Specifies a logical hostname. The logical hostname is the network interface (IP address) that clients use to access the Sun Cluster HA for Sybase ASE data service.
`-j` *resource*	Specifies an optional name for the logical-hostname resource. If you do not specify a name, the default resource name appears as the first name after the `-l` option.
`-n` *netiflist*	Specifies an optional comma-separated list that identifies the NAFO groups on each node. The *netiflist* must represent all nodes in the resource group's *nodelist*. If you do not specify this option, the `scrgadm` command attempts to discover a network adapter on the subnet that the *hostnamelist* identifies for each *nodelist* node.

6. **Login as sybase_id.**

7. **Install the Sybase ASE software.**

   Regardless of where you install the Sybase ASE software, modify each node's `/etc/system` files as you would in standard Sybase ASE installation procedures. For instructions on installing Sybase ASE software, refer to the Sybase installation and configuration guides.

   Note – For every Sybase server, enter the logical hostname when asked to specify its name.

Where to Go From Here

After you install the Sybase ASE software, go to "How to Configure Sybase ASE Database Access With Solstice DiskSuite" on page 178 if you use the Solstice DiskSuite volume manager. Go to "How to Configure Sybase ASE Database Access With VxVM" on page 179 if you use the VERITAS Volume Manager (VxVM).

▼ How to Verify the Sybase ASE Installation

Perform the following steps to verify the Sybase ASE software installation.

1. **Verify that the `sybase_id` user and the `dba` group own the `$SYBASE_HOME` directory and `$SYBASE_HOME` children directories.**

2. **Run the `scstat`(1M) command to verify that the Sun Cluster software functions correctly.**

Creating the Sybase ASE Database Environment

The procedures in this section enable you to complete the following tasks.

- Configure Sybase ASE database access with Solstice DiskSuite or VxVM.
- Create the Sybase ASE database environment.

▼ How to Configure Sybase ASE Database Access With Solstice DiskSuite

If you use the Solstice DiskSuite volume manager, perform the following steps to configure Sybase ASE database access with the Solstice DiskSuite volume manager.

1. **Configure the disk devices for the Solstice DiskSuite software to use.**

   See the *Sun Cluster 3.0 U1 Installation Guide* for information on how to configure Solstice DiskSuite.

2. **If you use raw devices to contain the databases, run the following commands to change each raw-mirrored metadevice's owner, group, and mode.**

   If you do not use raw devices, do not perform this step.

   a. **If you create raw devices, run the following commands for each device on *each node* that can master the Sybase ASE resource group.**

```
# chown sybase_id /dev/md/metaset/rdsk/dn
# chgrp dba_id /dev/md/metaset/rdsk/dn
# chmod 600 /dev/md/metaset/rdsk/dn
```

metaset Specifies the name of the diskset.

/rdsk/d*n* Specifies the name of the raw disk device within the *metaset* diskset.

b. Verify that the changes are effective.

```
# ls -lL /dev/md/metaset/rdsk/dn
```

▼ How to Configure Sybase ASE Database Access With VxVM

If you use VxVM software, perform the following steps to configure Sybase ASE database access with the VxVM software.

1. **Configure the disk devices for the VxVM software to use.**

   See the *Sun Cluster 3.0 U1 Installation Guide* for information on how to configure VERITAS Volume Manager.

2. **If you use raw devices to contain the databases, run the following commands on the current disk-group primary to change each device's owner, group, and mode.**

   If you do not use raw devices, do not perform this step.

   a. If you create raw devices, run the following command for each raw device.

```
# vxedit -g diskgroup set user=sybase_id group=dba_id mode=0600 volume
```

diskgroup Specifies the name of the disk group.

volume Specifies the name of the volume within the disk group.

b. Verify that the changes are effective.

```
# ls -lL /dev/vx/rdsk/diskgroup/volume
```

c. Reregister the disk device group with the cluster to keep the VxVM namespace consistent throughout the cluster.

```
# scconf -c -D name=diskgroup
```

▼ How to Create the Sybase ASE Database Environment

Before you perform this procedure, ensure that you have completed the following tasks.

- Establish a highly available IP address and name, that is, a logical hostname that operates at installation time.
- Locate device paths for all Sybase ASE devices—including the master device and system devices—in the highly available cluster file system. Configure device paths as one of the following file types.
  - Regular files
  - Raw devices
  - Files that the Solstice DiskSuite software or the VxVM software manage
- Locate the Sybase ASE server logs in either the cluster file system or the local file system.
- The Sybase ASE 12.0 environment consists of the data server, backup server, monitor server, text server, and XP server. The data server is the only server that you must configure—you can choose whether to configure all other servers.
- The entire cluster must contain only one copy of the `interfaces` file. The `$SYBASE` directory contains the `interfaces` file. If you plan to maintain per-node file copies, make sure the file contents are identical.

  All clients that connect to Sybase ASE servers connect with Sybase OpenClient libraries and utilities. When you configure the Sybase ASE software, in the `interfaces` file, enter information about the logical hostname and various ports. All clients use this connection information to connect to the Sybase ASE servers.

Note – The Sun Cluster software supports only the Sybase ASE 12.0 Base 32-bit configuration.

Perform the following steps to create the Sybase ASE database environment.

1. **Run the GUI-based utility `srvbuild` to create the Sybase ASE database.**

   The `$SYBASE/ASE_12-0/bin` directory contains this utility. See the Sybase ASE document entitled “Installing Sybase Adaptive Server Enterprise on Sun Solaris 2.x (SPARC).”

2. **To verify successful database installation, make sure that all servers start correctly.**

   Run the `ps`(1) command to verify the operation of all servers. Sybase ASE server logs indicate any errors that have occurred.

3. **Set the password for the Sybase ASE system administrator account.**

   See the *Sybase Adaptive Server Enterprise System Administration Guide* for details on changing the "sa" login password.

4. **Create a new Sybase ASE account for fault monitoring.**

   This account enables the fault monitor to perform the following tasks.

   - Support queries to system tables.
   - Create and update user tables.

   Note – Do not use the `sa` account for these purposes.

   See "Sun Cluster HA for Sybase ASE Fault Monitor" on page 190 for more information.

5. **Update the stop file with the `sa` password.**

   Because the stop file contains the `sa` password, protect the file with the appropriate permissions, and place the file in a directory that the system administrator chooses. Enable only the `sybase_id` user to read, write, and execute the stop file.

   Note – If you set up another Sybase ASE configuration on the same cluster, do not use `sybase_id` as the user ID for the additional configuration.

   See "Important Security Issues" on page 187 for more information about the stop file.

Where to Go From Here

After you create the Sybase ASE database environment, go to "How to Install Sun Cluster HA for Sybase ASE Packages" on page 182.

Installing the Sun Cluster HA for Sybase ASE Package

You can use the `scinstall`(1M) utility to install `SUNWscsyb`, the Sun Cluster HA for Sybase ASE data-service package, on a cluster. Do not use the `-s` option to non-interactive `scinstall` to install all data-service packages.

If you installed the SUNWscsyb data-service package as part of your initial Sun Cluster installation, proceed to "Registering and Configuring Sun Cluster HA for Sybase ASE" on page 182. Otherwise, use the following procedure to install the SUNWscsyb package.

▼ How to Install Sun Cluster HA for Sybase ASE Packages

You need the Sun Cluster Agents CD to complete this procedure. Perform this procedure on all cluster nodes that run the Sun Cluster HA for Sybase ASE package.

1. **Load the Agents CD into the CD-ROM drive.**
2. **Run the scinstall utility with no options.**

   This step starts the scinstall utility in interactive mode.
3. **Select the Add Support for New Data Service to This Cluster Node menu option.**

   This option enables you to load software for any data services that exist on the CD.
4. **Exit the scinstall utility.**
5. **Unload the CD from the drive.**

Where to Go From Here

When you finish the Sun Cluster HA for Sybase ASE package installation, go to "How to Register and Configure Sun Cluster HA for Sybase ASE" on page 183.

Registering and Configuring Sun Cluster HA for Sybase ASE

Use the procedures in this section to complete the following tasks.

- **Register and configure the Sun Cluster HA for Sybase ASE data service –** Register and configure Sun Cluster HA for Sybase ASE as a failover data service.

- **Configure the `SUNW.HAStorage` resource type –** Register and configure resources and resource groups for the Sybase ASE server. See Chapter 1 and the *Sun Cluster 3.0 U1 Concepts* document for details on resources and resource groups.

▼ How to Register and Configure Sun Cluster HA for Sybase ASE

This procedure describes how to use the `scrgadm`(1M) command to register and configure the Sun Cluster HA for Sybase ASE data service.

Note – Other options also enable you to register and configure the data service. See "Tools for Data-Service Resource Administration" on page 8 for details about these options.

To perform this procedure, you must have the following information.

- The names of the cluster nodes that master the data service.
- The logical hostname that clients use to access the data service. You typically configure the IP address when you install the cluster. See the sections in the *Sun Cluster 3.0 U1 Installation Guide* on planning the Sun Cluster environment and on how to install the Solaris operating environment for details.
- The path to the Sybase ASE application installation.

Note – Perform the following steps on one cluster member.

---

1. **Become superuser on a cluster member.**

2. **Run the `scrgadm` command to register resource types for the Sun Cluster HA for Sybase ASE data service.**

```
# scrgadm -a -t SUNW.sybase
```

`-a` Adds the resource type for the data service.

`-t SUNW.sybase` Specifies the resource-type name that is predefined for your data service.

3. **Create Sybase ASE application resources in the failover resource group.**

```
# scrgadm -a -j resource -g resource-group \
-t SUNW.sybase \
-x Environment_File=environment-file-path \
-x Adaptive_Server_Name=adaptive-server-name \
-x Backup_Server_Name=backup-server-name \
-x Text_Server_Name=text-server-name \
-x Monitor_Server_Name=monitor-server-name \
-x Adaptive_Server_Log_File=log-file-path \
-x Stop_File=stop-file-path \
-x Connect_string=user/passwd \
```

`-j` *resource*	Specifies the resource name to add.
`-g` *resource-group*	Specifies the resource-group name into which the RGM places the resources.
`-t SUNW.sybase`	Specifies the resource type to add.
`-x Environment_File=` *environment-file*	Sets the environment-file name.
`-x Adaptive_Server_Name=` *adaptive-server-name*	Sets the adaptive-server name.
`-x Backup_Server_Name=` *backup-server-name*	Sets the backup-server name.
`-x Text_Server_Name=` *text-server-name*	Sets the text-server name.
`-x Monitor_Server_Name=` *monitor-server-name*	Sets the monitor-server name.
`-x Adaptive_Server_Log_File=` *log-file-path*	Sets the path to the log file for the adaptive server.
`-x Stop_File=`*stop-file-path*	Sets the path to the stop file.
`-x Connect_string=`*user/passwd*	Specifies the user name and password that the fault monitor uses to connect to the database.

You do not have to specify extension properties that have default values. See "Configuring Sun Cluster HA for Sybase ASE Extension Properties" on page 188 for more information.

4. **Run the `scswitch`(1M) command to complete the following tasks.**

- Enable the resource and fault monitoring.
- Move the resource group into a managed state.
- Bring the resource group online.

```
# scswitch -Z -g resource-group
```

▼ How to Configure the SUNW.HAStorage Resource Type

The SUNW.HAStorage resource type synchronizes actions between HA storage and the Sun Cluster HA for Sybase ASE data service. The Sun Cluster HA for Sybase ASE data service is disk intensive, and therefore you should configure the SUNW.HAStorage resource type.

See the SUNW.HAStorage(5) man page and "Relationship Between Resource Groups and Disk Device Groups" on page 4 for more information about the SUNW.HAStorage resource type. See "How to Set Up SUNW.HAStorage Resource Type for New Resources" on page 239 for the procedure on how to configure the SUNW.HAStorage resource type.

Where to Go From Here

After you register and configure the Sun Cluster HA for Sybase ASE data service, go to "How to Verify the Sun Cluster HA for Sybase ASE Installation" on page 186.

Verifying the Sun Cluster HA for Sybase ASE Installation

Perform the following verification tests to make sure that you have correctly installed and configured the Sun Cluster HA for Sybase ASE data service.

These sanity checks make sure that all nodes that run the Sun Cluster HA for Sybase ASE data service can start the Sybase ASE data server. These checks also ensure that other nodes in the configuration can access the Sybase ASE data server. Perform these sanity checks to isolate any problems with starting the Sybase ASE software from the Sun Cluster HA for Sybase ASE data service.

How to Verify the Sun Cluster HA for Sybase ASE Installation

1. **Log in to the node monitoring the Sybase ASE resource group.**

2. **Set the Sybase ASE environment variables.**

   The environment variables are the variables you specify with the `Environment_file` extension property. You typically name these variables `SYBASE.sh`.

3. **Verify that the Sun Cluster HA for Sybase ASE resource is online.**

```
# scstat -g
```

4. **Inspect the Sybase ASE logs to determine the cause of any errors that have occurred.**

5. **Confirm that you can connect to the data server and execute the following test command.**

```
# isql -S adaptive-server -U sa

isql> sp_help
isql> go
isql> quit
```

6. **Kill the process for the Sybase ASE data server.**

   The Sun Cluster software restarts the process.

7. **Switch the resource group that contains the Sybase ASE resource to another cluster member.**

```
# scswitch -z -g resource-group -h node
```

8. **Log in to the node that now contains the resource group.**

9. **Repeat Step 3 and Step 5.**

Note – Sybase ASE client connections cannot survive a Sun Cluster HA for Sybase ASE switchover. If a switchover occurs, the existing client connections to Sybase ASE terminate, and clients must reestablish their connections. After a switchover, the time that is required to replay the Sybase ASE transaction log determines Sun Cluster HA for Sybase ASE recovery time.

Understanding Sun Cluster HA for Sybase ASE Logging and Security Issues

The following sections contain information about Sun Cluster HA for Sybase ASE logging and security issues.

Sun Cluster HA for Sybase ASE Logging

The Sun Cluster HA for Sybase ASE data service logs messages to the file `message_log` in the `/opt/SUNWscsyb/log` directory. Although this file cannot exceed 512 Kbytes, the Sun Cluster HA for Sybase ASE data service does not delete old log files. The number of log files, therefore, can grow to a large number.

The Sun Cluster HA for Sybase ASE data service writes all error messages in the `syslog` file. The Sun Cluster HA for Sybase ASE data service also logs fault-monitor history to the file `restart_history` in the `log` directory. These files can also grow to a large number.

As part of your regular file maintenance, check the following log files and remove files that you no longer need.

- `syslog`
- `message_log`
- `restart_history`

Important Security Issues

The Sun Cluster HA for Sybase ASE data service requires that you embed the system administrator's password in a stop file. The `/opt/SUNWscsyb` directory contains the template for the stop file, `Sybase_stop_servers`. The Sun Cluster HA for Sybase ASE data service uses this file to log in to the Sybase ASE environment and to stop the Sybase ASE servers. Enable the `sybase_id` user to execute the stop file, but protect the file from general access. Give read, write, and execute privileges to only the following users.

- `sybase_id` user
- `sybase_id` group

Configuring Sun Cluster HA for Sybase ASE Extension Properties

This section describes how to configure Sun Cluster HA for Sybase ASE extension properties. Typically, you use the command line `scrgadm -x` *parameter*=*value* to configure extension properties when you create the Sybase ASE resources. You can also use the procedures described in Chapter 11 to configure them later.

See the `r_properties`(5) and the `rg_properties`(5) man pages for details on all Sun Cluster extension properties.

TABLE 10-2 describes the extension properties that you can set for the Sybase ASE server resource. You can update some extension properties dynamically. You can update others, however, only when you create or disable a resource. The Tunable column in the following table indicates when you can update each property.

TABLE 10-2 Sun Cluster HA for Sybase ASE Extension Properties

Name/Data Type	Default	Range	Tunable	Description
`Environment_File`	None	Minimum=1	When disabled	File that contains all Sybase ASE environment variables.
`Adaptive_Server_Name`	None	Minimum=1	When disabled	Data-server name. The Sun Cluster HA for Sybase ASE data service uses this property to locate the `RUN` server in the `$SYBASE/$ASE/install` directory.
`Backup_Server_Name`	`Null`		When disabled	Backup-server name. The Sun Cluster HA for Sybase ASE data service uses this property to locate the `RUN` server in the `$SYBASE/$ASE/install` directory. If you do not set this property, the Sun Cluster HA for Sybase ASE data service will not manage the server.

TABLE 10-2 Sun Cluster HA for Sybase ASE Extension Properties

Name/Data Type	Default	Range	Tunable	Description
`Monitor_Server_Name`	`Null`		When disabled	Monitor-server name. The Sun Cluster HA for Sybase ASE data service uses this property to locate the `RUN` server in the `$SYBASE/$ASE/install` directory. If you do not set this property, the Sun Cluster HA for Sybase ASE data service will not manage the server.
`Text_Server_Name`	`Null`		When disabled	Text-server name. The Sun Cluster HA for Sybase ASE data service uses this property to locate the `RUN` server in the `$SYBASE/$ASE/install` directory. If you do not set this property, the Sun Cluster HA for Sybase ASE data service will not manage the server.
`Adaptive_Server_Log_File`	None	Minimum=1	When disabled	The Sybase ASE data-server log. The Sun Cluster HA for Sybase ASE data service continually reads this property for error monitoring.
`Stop_File`	None	Minimum=1	When disabled	The Sun Cluster HA for Sybase ASE data service uses this property during server `stoppages`. This property contains the `sa` password. Protect this property from general access.
`Probe_timeout`	`30` seconds	`1 – 99999` seconds	Any time	Time-out value for the fault-monitor probe.
`Debug_level`	`0`	`0 – 15`	Any time	Debug level for writing to the Sun Cluster HA for Sybase ASE log.
`Connect_string`	None	Minimum=1	When disabled	String of format *user/password*. The Sun Cluster HA for Sybase ASE data service uses this property for database probes.
`Connect_cycle`	`5`	`1 – 100`	Any time	Number of fault-monitor probe cycles before the Sun Cluster HA for Sybase ASE data service establishes a new connection.
`Wait_for_online`	`FALSE`	`TRUE – FALSE`	Any time	Whether the start method waits for the database to come online before exiting.

Sun Cluster HA for Sybase ASE Fault Monitor

The Sun Cluster HA for Sybase ASE fault monitor queries the Sybase ASE server to determine server health.

The fault monitor consists of the following processes.

- a main fault-monitor process
- a database-client fault probe

The following sections describe the Sun Cluster HA for Sybase ASE fault-monitor processes and the extension properties that the fault monitor uses.

Main Fault-Monitor Process

The fault-monitor process diagnoses errors and checks statistics. The monitor labels an operation successful if the following conditions occur.

- The database is online.
- The activity check returns no errors.
- The test transaction returns no errors.

If an operation fails, the main process checks the action table for an action to perform and then performs the predetermined action. If an operation fails, the main process can perform the following actions, which execute external programs as separate processes in the background.

- switchover
- stopping the server
- restarting the server
- stopping the resource group
- restarting the resource group

The server fault monitor also scans the `Adaptive_Server_Log` file and acts to correct any errors that the scan identifies.

Database-Client Fault Probe

The database-client fault probe performs activity checks and test transactions. The extension property `Connect_string` specifies an account that performs all database operations. The extension property `Probe_timeout` sets the time-out value that the probe uses to determine time that has elapsed in a successful database probe.

Extension Properties

The fault monitor uses the following extension properties.

- `Thorough_probe_interval`
- `Retry_count`
- `Retry_interval`
- `Probe_timeout`
- `Connect_string`
- `Connect_cycle`
- `Adaptive_Server_Log`

See "Configuring Sun Cluster HA for Sybase ASE Extension Properties" on page 188 for more information about these extension properties.

CHAPTER **11**

Administering Data-Service Resources

This chapter describes how to use the `scrgadm`(1M) command to manage resources, resource groups, and resource types within the cluster. See "Tools for Data-Service Resource Administration" on page 8 to determine if you can use other tools to complete a procedure.

This chapter contains the following procedures.

- "How to Register a Resource Type" on page 196
- "How to Create a Failover Resource Group" on page 198
- "How to Create a Scalable Resource Group" on page 199
- "How to Add a Logical-Hostname Resource to a Resource Group" on page 202
- "How to Add a Shared-Address Resource to a Resource Group" on page 204
- "How to Add a Failover Application Resource to a Resource Group" on page 206
- "How to Add a Scalable Application Resource to a Resource Group" on page 208
- "How to Bring a Resource Group Online" on page 211
- "How to Disable a Resource Fault Monitor" on page 212
- "How to Enable a Resource Fault Monitor" on page 213
- "How to Remove a Resource Type" on page 214
- "How to Remove a Resource Group" on page 216
- "How to Remove a Resource" on page 218
- "How to Switch the Current Primary of a Resource Group" on page 219
- "How to Disable a Resource and Move Its Resource Group Into the Unmanaged State" on page 221
- "How to Display Resource Type, Resource Group, and Resource Configuration Information" on page 224
- "How to Change Resource-Type Properties" on page 225
- "How to Change Resource-Group Properties" on page 226
- "How to Change Resource Properties" on page 227
- "How to Clear the `STOP_FAILED` Error Flag on Resources" on page 229

- "How to Re-register Preregistered Resource Types" on page 231
- "How to Add a Node to a Resource Group" on page 232
- "How to Remove a Node From a Resource Group" on page 235
- "How to Set Up SUNW.HAStorage Resource Type for New Resources" on page 239

See Chapter 1 and the *Sun Cluster 3.0 U1 Concepts* document for overview information about resource types, resource groups, and resources.

Administering Data-Service Resources

TABLE 11-1 lists the sections that describe the administration tasks for data-service resources.

TABLE 11-1 Task Map: Data Service Administration

Task	For Instructions, Go To …
Register a resource type	"How to Register a Resource Type" on page 196
Create failover or scalable resource groups	"How to Create a Failover Resource Group" on page 198 "How to Create a Scalable Resource Group" on page 199
Add logical hostnames or shared addresses and data-service resources to resource groups	"How to Add a Logical-Hostname Resource to a Resource Group" on page 202 "How to Add a Shared-Address Resource to a Resource Group" on page 204 "How to Add a Failover Application Resource to a Resource Group" on page 206 "How to Add a Scalable Application Resource to a Resource Group" on page 208
Enable resources and resource monitors, manage the resource group, and bring the resource group and its associated resources online	"How to Bring a Resource Group Online" on page 211
Disable and enable resource monitors independent of the resource	"How to Disable a Resource Fault Monitor" on page 212 "How to Enable a Resource Fault Monitor" on page 213
Remove resource types from the cluster	"How to Remove a Resource Type" on page 214
Remove resource groups from the cluster	"How to Remove a Resource Group" on page 216

TABLE 11-1 Task Map: Data Service Administration *(Continued)*

Task	For Instructions, Go To …
Remove resources from resource groups	"How to Remove a Resource" on page 218
Switch the primary for a resource group	"How to Switch the Current Primary of a Resource Group" on page 219
Disable resources and move their resource group into the unmanaged state	"How to Disable a Resource and Move Its Resource Group Into the Unmanaged State" on page 221
Display resource type, resource group, and resource configuration information	"How to Display Resource Type, Resource Group, and Resource Configuration Information" on page 224
Change resource-type, resource group, and resource properties	"How to Change Resource-Type Properties" on page 225 "How to Change Resource-Group Properties" on page 226 "How to Change Resource Properties" on page 227
Clear error flags for failed Resource Group Manager (RGM) processes	"How to Clear the `STOP_FAILED` Error Flag on Resources" on page 229
Re-register the built-in resource types `LogicalHostname` and `SharedAddress`	"How to Re-register Preregistered Resource Types" on page 231
Update the network interface ID list for the network resources, and update the node list for the resource group	"How to Add a Node to a Resource Group" on page 232
Remove a node from a resource group	"How to Remove a Node From a Resource Group" on page 235
Set up `SUNW.HAStorage` for resource groups so as to synchronize the startups between those resource groups and disk device groups	"How to Set Up `SUNW.HAStorage` Resource Type for New Resources" on page 239

Note – The procedures in this chapter describe how to use the `scrgadm`(1M) command to complete these tasks. Other tools also enable you to administer your resources. See "Tools for Data-Service Resource Administration" on page 8 for details about these options.

Configuring and Administering Sun Cluster Data Services

Configuring a Sun Cluster data service is a single task composed of several procedures. The following procedures enable you to perform the following tasks.

- Register a resource type.

- Create resource groups.
- Add resources into the resource groups.
- Bring the resources online.

Use the procedures in this chapter to update your data service configuration after the initial configuration. For example, to change resource type, resource group, and resource properties, go to "Changing Resource Type, Resource Group, and Resource Properties" on page 224.

Registering a Resource Type

A resource type provides specification of common properties and callback methods that apply to all the resources of the given type. You must register a resource type before creating a resource of that type. See Chapter 1 for details about resource types.

How to Register a Resource Type

To complete this procedure, you must supply the name for the resource type you are registering, which is an abbreviation for the data-service name. This name maps to the name shown on your data-service license certificate. See the *Sun Cluster 3.0 U1 Release Notes* for the mapping between the names and the license certificate names.

See the `scrgadm`(1M) man page for additional information.

Note – Perform this procedure from any cluster node.

1. **Become superuser on a cluster member.**

2. **Register the resource type.**

```
# scrgadm -a -t resource-type
```

`-a`
Adds the specified resource type.

`-t` *resource-type*
Specifies name of the resource type to add. See the *Sun Cluster 3.0 U1 Release Notes* to determine the predefined name to supply.

3. **Verify that the resource type has been registered.**

```
# scrgadm -pv -t resource-type
```

Example – Registering Resource Types

The following example registers the Sun Cluster HA for iPlanet Web Server data service (internal name iws).

```
# scrgadm -a -t SUNW.iws
# scrgadm -pv -t SUNW.iws
Res Type name:                                   SUNW.iws
  (SUNW.iws) Res Type description:               None registered
  (SUNW.iws) Res Type base directory:            /opt/SUNWschtt/bin
  (SUNW.iws) Res Type single instance:           False
  (SUNW.iws) Res Type init nodes:                All potential masters
  (SUNW.iws) Res Type failover:                  False
  (SUNW.iws) Res Type version:                   1.0
  (SUNW.iws) Res Type API version:               2
  (SUNW.iws) Res Type installed on nodes:        All
  (SUNW.iws) Res Type packages:                  SUNWschtt
```

Where to Go From Here

After registering resource types, you can create resource groups and add resources to the resource group. See "Creating a Resource Group" on page 197 for details.

Creating a Resource Group

A resource group contains a set of resources, all of which are brought online or offline together on a given node or set of nodes. You must create an empty resource group before placing resources into it.

The two resource group types are *failover* and *scalable*. A failover resource group can be online on one node only at any time, while a scalable resource group can be online on multiple nodes simultaneously.

The following procedure describes how to use the scrgadm(1M) command to register and configure your data service.

See Chapter 1 and the *Sun Cluster 3.0 U1 Concepts* document for conceptual information on resource groups.

▼ How to Create a Failover Resource Group

A failover resource group contains network addresses, such as the built-in resource types `LogicalHostname` and `SharedAddress`, as well as failover resources, such as the data-service application resources for a failover data service. The network resources, along with their dependent data-service resources, move between cluster nodes when data services fail over or are switched over.

See the `scrgadm`(1M) man page for additional information.

Note – Perform this procedure from any cluster node.

---

1. **Become superuser on a cluster member.**
2. **Create the failover resource group.**

```
# scrgadm -a -g resource-group [-h nodelist]
```

`-a` Adds the specified resource group.

`-g` *resource-group* Specifies your choice of the name of the failover resource group to add. This name must begin with an ASCII character.

`-h` *nodelist* Specifies an optional ordered list of nodes that can master this resource group. If you do not specify this list, it defaults to all the nodes in the cluster.

3. **Verify that the resource group has been created.**

```
# scrgadm -pv -g resource-group
```

Example – Creating a Failover Resource Group

This example shows the addition of a failover resource group (resource-group-1) that two nodes (phys-schost-1 and phys-schost-2) can master.

```
# scrgadm -a -g resource-group-1 -h phys-schost1,phys-schost-2
# scrgadm -pv -g resource-group-1
Res Group name:                                            resource-group-1
  (resource-group-1) Res Group RG_description:             <NULL>
  (resource-group-1) Res Group management state:           Unmanaged
  (resource-group-1) Res Group Failback:                   False
  (resource-group-1) Res Group Nodelist:                   phys-schost-1
                                                           phys-schost-2
  (resource-group-1) Res Group Maximum_primaries:          1
  (resource-group-1) Res Group Desired_primaries:          1
  (resource-group-1) Res Group RG_dependencies:            <NULL>
  (resource-group-1) Res Group mode:                       Failover
  (resource-group-1) Res Group network dependencies:       True
  (resource-group-1) Res Group Global_resources_used:      All
  (resource-group-1) Res Group Pathprefix:
```

Where to Go From Here

After creating a failover resource group, you can add application resources to this resource group. See “Adding Resources to Resource Groups” on page 201 for the procedure.

▼ How to Create a Scalable Resource Group

A scalable resource group is used with scalable services. The shared-address feature is the Sun Cluster networking facility that enables the multiple instances of a scalable service to appear as a single service. You must first create a failover resource group that contains the shared addresses on which the scalable resources depend. Next, create a scalable resource group, and add scalable resources to that group.

See the scrgadm(1M) man page for additional information.

Note – Perform this procedure from any cluster node.

1. **Become superuser on a cluster member.**

2. **Create the failover resource group that holds the shared addresses that the scalable resource will use.**

3. **Create the scalable resource group.**

```
# scrgadm -a -g resource-group \
-y Maximum_primaries=m \
-y Desired_primaries=n \
-y RG_dependencies=depend-resource-group \
-h nodelist]
```

`-a`
Adds a scalable resource group.

`-g` *resource-group*
Specifies your choice of the name of the scalable resource group to add.

`-y Maximum_primaries=`*m*
Specifies the maximum number of active primaries for this resource group.

`-y Desired_primaries=`*n*
Specifies the number of active primaries on which the resource group should attempt to start.

`-y RG_dependencies=`*depend-resource-group*
Identifies the resource group that contains the shared-address resource on which the resource group being created depends.

`-h` *nodelist*
Specifies an optional list of nodes on which this resource group is to be available. If you do not specify this list, the value defaults to all nodes.

4. **Verify that the scalable resource group has been created.**

```
# scrgadm -pv -g resource-group
```

Example – Creating a Scalable Resource Group

This example shows the addition of a scalable resource group (resource-group-1) to be hosted on two nodes (phys-schost-1, phys-schost-2). The scalable resource group depends on the failover resource group (resource-group-2) that contains the shared addresses.

```
# scrgadm -a -g resource-group-1 \
-y Maximum_primaries=2 \
-y Desired_primaries=2 \
-y RG_dependencies=resource-group-2 \
-h phys-schost-1,phys-schost-2
# scrgadm -pv -g resource-group-1
Res Group name:                                          resource-group-1
  (resource-group-1) Res Group RG_description:           <NULL>
  (resource-group-1) Res Group management state:         Unmanaged
  (resource-group-1) Res Group Failback:                 False
  (resource-group-1) Res Group Nodelist:                 phys-schost-1
                                                         phys-schost-2
  (resource-group-1) Res Group Maximum_primaries:        2
  (resource-group-1) Res Group Desired_primaries:        2
  (resource-group-1) Res Group RG_dependencies:          resource-group-2
  (resource-group-1) Res Group mode:                     Scalable
  (resource-group-1) Res Group network dependencies:     True
  (resource-group-1) Res Group Global_resources_used:    All
  (resource-group-1) Res Group Pathprefix:
```

Where to Go From Here

After a scalable resource group has been created, you can add scalable application resources to the resource group. See "How to Add a Scalable Application Resource to a Resource Group" on page 208 for details.

Adding Resources to Resource Groups

A resource is an instantiation of a resource type. You must add resources to a resource group before the RGM can manage the resources. This section describes the following three resource types.

- logical-hostname resources
- shared-address resources
- data-service (application) resources

Logical-hostname resources and shared-address resources are always added to failover resource groups. Data-service resources for failover data services are added to failover resource groups. Failover resource groups contain both the logical-hostname resources and the application resources for the data service. Scalable resource groups contain only the application resources for scalable services. The shared-address resources on which the scalable service depends must reside in a separate failover resource group. You must specify dependencies between the scalable application resources and the shared-address resources for the data service to scale across cluster nodes.

See the *Sun Cluster 3.0 U1 Concepts* document and Chapter 1 for more information on resources.

How to Add a Logical-Hostname Resource to a Resource Group

To complete this procedure, you must supply the following information.

- the name of the failover resource group into which you are adding the resource
- the hostnames you are adding to the resource group

See the `scrgadm`(1M) man page for additional information.

Note – Perform this procedure from any cluster node.

---

1. **Become superuser on a cluster member.**

2. **Add the logical-hostname resource to the resource group.**

```
# scrgadm -a -L [-j resource] -g resource-group -l hostnamelist, … [-n netiflist]
```

`-a` Adds a logical-hostname resource.

`-L` Specifies the logical-hostname resource form of the command.

`-j` *resource* Specifies an optional resource name of your choice. If you do not specify this option, the name defaults to the first hostname specified with the `-l` option.

`-g` *resource-group* — Specifies the name of the resource group in which this resource resides.

`-l` *hostnamelist*, … — Specifies a comma-separated list of UNIX hostnames (logical hostnames) by which clients communicate with services in the resource group.

`-n` *netiflist* — Specifies an optional comma-separated list that identifies the NAFO groups on each node. All nodes in *nodelist* of the resource group must be represented in *netiflist*. See the `scrgadm`(1M) man page for a description of the syntax for specifying *netiflist*. If you do not specify this option, `scrgadm` attempts to discover a net adapter on the subnet that the *hostnamelist* identifies for each node in *nodelist*.

3. **Verify that the logical-hostname resource has been added.**

```
# scrgadm -pv -j resource
```

The resource addition action causes the Sun Cluster software to validate the resource. If the validation succeeds, the resource can be enabled, and the resource group can be moved into the state where the RGM manages it. If the validation fails, the `scrgadm` command produces an error message to that effect and exits. If the validation fails, check the `syslog` on each node for an error message. The message appears on the node that performed the validation, not necessarily the node on which you ran the `scrgadm` command.

Example – Adding a Logical-Hostname Resource to a Resource Group

This example shows the addition of logical-hostname resource (`resource-1`) to a resource group (`resource-group-1`).

```
# scrgadm -a -L -j resource-1 -g resource-group-1 -l schost-1
# scrgadm -pv -j resource-1
Res Group name: resource-group-1
(resource-group-1) Res name:                             resource-1
  (resource-group-1:resource-1) Res R_description:
  (resource-group-1:resource-1) Res resource type:       SUNW.LogicalHostname
  (resource-group-1:resource-1) Res resource group name:  resource-group-1
  (resource-group-1:resource-1) Res enabled:             False
  (resource-group-1:resource-1) Res monitor enabled:     True
```

Where to Go From Here

After adding logical-hostname resources, use the procedure "How to Bring a Resource Group Online" on page 211 to bring them online.

▼ How to Add a Shared-Address Resource to a Resource Group

To complete this procedure, you must supply the following information.

- The name of the resource group into which you are adding the resource. This group must be a failover resource group created previously.
- The hostnames you are adding to the resource group.

See the `scrgadm`(1M) man page for additional information.

Note – Perform this procedure from any cluster node.

---

1. **Become superuser on a cluster member.**

2. **Add the shared-address resource to the resource group.**

```
# scrgadm -a -S [-j resource] -g resource-group -l hostnamelist, … \
[-X auxnodelist] [-n netiflist]
```

`-a` Adds shared-address resources.

`-S` Specifies the shared-address resource form of the command.

`-j` *resource* Specifies an optional resource name of your choice. If you do not specify this option, the name defaults to the first hostname specified with the `-l` option.

`-g` *resource-group* Specifies the resource-group name.

-l *hostnamelist, …* Specifies a comma-separated list of shared address hostnames.

-X *auxnodelist* Specifies a comma-separated list of physical node names or IDs that identify the cluster nodes that can host the shared address but never serve as primary in the case of failover. These nodes are mutually exclusive, with the nodes identified in the resource group *nodelist* as potential masters.

-n *netiflist* Specifies an optional comma-separated list that identifies the NAFO groups on each node. All the nodes in *nodelist* of the resource group must be represented in the *netiflist*. See the scrgadm(1M) man page for a description of the syntax for specifying *netiflist*. If you do not specify this option, scrgadm attempts to discover a net adapter on the subnet that the *hostnamelist* identifies for each node in *nodelist*.

3. **Verify that the shared-address resource has been added and validated.**

```
# scrgadm -pv -j resource
```

The resource addition action causes the Sun Cluster software to validate the resource. If the resource is successfully validated, the resource can be enabled, and the resource group can be moved into the state where the RGM manages it. If the validation fails, the scrgadm command produces an error message to this effect and exits. If the validation fails, check the syslog on each node for an error message. The message appears on the node that performed the validation, not necessarily the node on which you ran the scrgadm command.

Example – Adding a Shared-Address Resource to a Resource Group

This example shows the addition of a shared-address resource (resource-1) to a resource group (resource-group-1).

```
# scrgadm -a -S -j resource-1 -g resource-group-1 -l schost-1
# scrgadm -pv -j resource-1
(resource-group-1) Res name:                                resource-1
    (resource-group-1:resource-1) Res R_description:
    (resource-group-1:resource-1) Res resource type:        SUNW.SharedAddress
    (resource-group-1:resource-1) Res resource group name:  resource-group-1
    (resource-group-1:resource-1) Res enabled:              False
    (resource-group-1:resource-1) Res monitor enabled:      True
```

Where to Go From Here

After adding a shared resource, use the procedure "How to Bring a Resource Group Online" on page 211 to enable the resource.

▼ How to Add a Failover Application Resource to a Resource Group

A failover application resource is an application resource that uses logical hostnames created in a failover resource group previously.

To complete this procedure, you must supply the following information.

- the name of the failover resource group into which you are adding the resource
- the name of the resource type for the resource
- the logical-hostname resources that the application resource uses, which are the logical hostnames previously included in the same resource group

See the `scrgadm`(1M) man page for additional information.

Note – Perform this procedure from any cluster node.

---

1. **Become superuser on a cluster member.**

2. **Add a failover application resource to the resource group.**

```
# scrgadm -a -j resource -g resource-group -t resource-type \
[-x Extension_property=value, …] [-y Standard_property=value, …]
```

`-a` Adds a resource.

`-j` *resource* Specifies your choice of the name of the resource to add.

`-g` *resource-group* Specifies the name of the failover resource group created previously.

`-t` *resource-type*

Specifies the name of the resource type for the resource.

`-x` *Extension_property=value, …*

Specifies a comma-separated list of extension properties that depend on the particular data service. See the chapter for each data service to determine whether the data service requires this property.

`-y` *Standard_property=value, …*

Specifies a comma-separated list of standard properties that depends on the particular data service. See the chapter for each data service and Appendix A to determine whether the data service requires this property.

Note – You can set additional properties. See Appendix A and the chapter in this book on how to install and configure your failover data service for details.

---

3. **Verify that the failover application resource has been added and validated.**

```
# scrgadm -pv -j resource
```

The resource addition action causes the Sun Cluster software to validate the resource. If the validation succeeds, the resource can be enabled, and the resource group can be moved into the state where the RGM manages it. If the validation fails, check the `syslog` on each node for an error message. The message appears on the node that performed the validation, not necessarily the node on which you ran the `scrgadm` command.

Example – Adding a Failover Application Resource to a Resource Group

This example shows the addition of a resource (`resource-1`) to a resource group (`resource-group-1`). The resource depends on logical-hostname resources (`schost-1`, `schost-2`), which must reside in the same failover resource groups that you defined previously.

```
# scrgadm -a -j resource-1 -g resource-group-1 -t resource-type-1 \
-y Network_resources_used=schost-1,schost2 \
# scrgadm -pv -j resource-1
(resource-group-1) Res name:                                resource-1
    (resource-group-1:resource-1) Res R_description:
    (resource-group-1:resource-1) Res resource type:        resource-type-1
    (resource-group-1:resource-1) Res resource group name:  resource-group-1
    (resource-group-1:resource-1) Res enabled:              False
    (resource-group-1:resource-1) Res monitor enabled:      True
```

Where to Go From Here

After adding a failover application resource, use the procedure "How to Bring a Resource Group Online" on page 211 to enable the resource.

▼ How to Add a Scalable Application Resource to a Resource Group

A scalable application resource is an application resource that uses shared addresses in a failover resource group.

To complete this procedure, you must supply the following information:

- the name of the scalable resource group into which you are adding the resource
- the name of the resource type for the resource
- the shared-address resources that the scalable service resource uses, which are the shared addresses previously included in a failover resource group

See the `scrgadm`(1M) man page for additional information.

Note – Perform this procedure from any cluster node.

---

1. **Become superuser on a cluster member.**

2. **Add a scalable application resource to the resource group.**

```
# scrgadm -a -j resource -g resource-group -t resource-type \
-y Network_resources_used=network-resource[,network-resource...] \
-y Scalable=True
[-x Extension_property=value, …] [-y Standard_property=value, …]
```

`-a`
Adds a resource.

`-j` *resource*
Specifies your choice of the name of the resource to add.

`-g` *resource-group*
Specifies the name of a scalable service resource group created previously.

`-t` *resource-type*
Specifies the name of the resource type for this resource.

`-y Network_resources_used=`*network-resource*[,*network-resource*...]
Specifies the list of network resources (shared addresses) on which this resource depends.

`-y Scalable=True`
Specifies that this resource is scalable.

`-x` *Extension_property=value, …*
Specifies a comma-separated list of extension properties that depend on the particular data service. See the chapter for each data service to determine whether the data service requires this property.

`-y` *Standard_property=value, …*
Specifies a comma-separated list of standard properties that depends on the particular data service. See the chapter for each data service and Appendix A to determine whether the data service requires this property.

Note – You can set additional properties. See Appendix A and the chapter in this book on how to install and configure your scalable data service for information on other configurable properties. Specifically for scalable services, you typically set the `Port_list`, `Load_balancing_weights`, and `Load_balancing_policy` properties, which Appendix A describes.

---

3. **Verify that the scalable application resource has been added and validated.**

```
# scrgadm -pv -j resource
```

The resource addition action causes the Sun Cluster software to validate the resource. If the validation succeeds, the resource can be enabled and the resource group can be moved into the state where the RGM manages it. If the validation fails, check the `syslog` on each node for an error message. The message appears on the node that performed the validation, not necessarily the node on which you ran the `scrgadm` command.

Example – Adding a Scalable Application Resource to a Resource Group

This example shows the addition of a resource (`resource-1`) to a resource group (`resource-group-1`). Note that `resource-group-1` depends on the failover resource group that contains the network addresses being used (`schost-1` and `schost-2` in the following example). The resource depends on shared-address resources (`schost-1`, `schost-2`), which must reside in one or more failover resource groups that you defined previously.

```
# scrgadm -a -j resource-1 -g resource-group-1 -t resource-type-1 \
-y Network_resources_used=schost-1,schost-2 \
-y Scalable=True
# scrgadm -pv -j resource-1
(resource-group-1) Res name:                                resource-1
    (resource-group-1:resource-1) Res R_description:
    (resource-group-1:resource-1) Res resource type:        resource-type-1
    (resource-group-1:resource-1) Res resource group name:  resource-group-1
    (resource-group-1:resource-1) Res enabled:              False
    (resource-group-1:resource-1) Res monitor enabled:      True
```

Where to Go From Here

After you add a scalable application resource, follow the procedure "How to Bring a Resource Group Online" on page 211 to enable the resource.

Bringing Resource Groups Online

To enable resources to begin providing HA services, you must enable the resources in the resource group, enable the resource monitors, make the resource group managed, and bring the resource group online. You can perform these tasks individually or by using the following one-step procedure. See the `scswitch`(1M) man page for details.

Note – Perform this procedure from any cluster node.

▼ How to Bring a Resource Group Online

1. **Become superuser on a cluster member.**
2. **Enable the resource, and bring the resource group online.**

   If the resource monitor has been previously disabled, it will be enabled also.

   ```
   # scswitch -Z -g resource-group
   ```

   `-Z` Brings a resource group online by first enabling its resources and fault monitors.

   `-g` *resource-group* Specifies the name of the resource group to bring online. The group must be an existing resource group.

3. **Verify that the resource is online.**

   Run the following command on any cluster node, and look for the resource group state field to see if the resource group is online on the nodes specified in the node list.

   ```
   # scstat -g
   ```

Example – Bring a Resource Group Online

This example shows how to bring a resource group (`resource-group-1`) online and verify its status.

```
# scswitch -Z -g resource-group-1
# scstat -g
```

Where to Go From Here

After a resource group has been brought online, the resource group is configured and ready to use. If a resource or node fails, the RGM maintains availability of the resource group by automatically switching the resource group online on alternate nodes.

Disabling and Enabling Resource Monitors

The following procedures disable or enable resource fault monitors, not the resources themselves. A resource can continue to normal operation while its fault monitor is disabled. However, if the fault monitor is disabled and a data service fault occurs, automatic fault recovery is not initiated.

See the `scswitch(1M)` man page for additional information.

Note – Run this procedure from any cluster node.

▼ How to Disable a Resource Fault Monitor

1. **Become superuser on a cluster member.**

2. **Disable the resource fault monitor.**

```
# scswitch -n -M -j resource
```

`-n` Disable a resource or resource monitor.

`-M` Disable the fault monitor for the specified resource.

`-j` *resource* The name of the resource.

3. **Verify that the resource fault monitor has been disabled.**

   Run the following command on each cluster node and look for monitored fields (`RS Monitored`).

```
# scrgadm -pv
```

Example–Disabling a Resource Fault Monitor

This example shows how to disable a resource fault monitor.

```
# scrgadm -n -M -j resource-1
# scrgadm -pv
...
RS Monitored: no
...
```

▼ How to Enable a Resource Fault Monitor

1. **Become superuser on a cluster member.**
2. **Enable the resource fault monitor.**

```
# scswitch -e -M -j resource
```

`-e` Enable a resource or resource monitor.

`-M` Enable the fault monitor for the specified resource.

`-j` *resource* The name of the resource.

3. **Verify that the resource fault monitor has been enabled.**

   Run the following command on each cluster node and look for monitored fields (`RS Monitored`).

```
# scrgadm -pv
```

Example–Enabling a Resource Fault Monitor

This example shows how to enable a resource fault monitor.

```
# scrgadm -e -M -j resource-1
# scrgadm -pv
...
RS Monitored: yes
...
```

Removing Resource Types

You do not need to remove resource types that are not in use. However, if you want to remove a resource type, you can use this procedure to do so.

See the `scrgadm`(1M) and `scswitch`(1M) man pages for additional information.

Note – Perform this procedure from any cluster node.

▼ How to Remove a Resource Type

Before you remove a resource type, you must disable and remove all the resources of that type in all the resource groups in the cluster. Use the `scrgadm -pv` command to identify the resources and resource groups in the cluster.

1. **Become superuser on a cluster member.**
2. **Disable each resource of the resource type to be removed.**

```
# scswitch -n -j resource
```

`-n`	Disables the resource.
`-j` *resource*	Specifies the name of the resource to disable.

3. **Remove each resource of the resource type to be removed.**

```
# scrgadm -r -j resource
```

`-r`	Removes the specified resource.
`-j`	Specifies the name of the resource to remove.

4. **Remove the resource type.**

```
# scrgadm -r -t resource-type
```

`-r`	Removes the specified resource type.
`-t` *resource-type*	Specifies the name of the resource type to remove.

5. **Verify that the resource type has been removed.**

```
# scrgadm -p
```

Example – Removing a Resource Type

This example shows how to disable and remove all resources of a resource type (`resource-type-1`) and then remove the resource type itself. Here, `resource-1` is a resource of the resource type `resource-type-1`.

```
# scswitch -n -j resource-1
# scrgadm -r -j resource-1
# scrgadm -r -t resource-type-1
```

Removing Resource Groups

To remove a resource group, you must first remove all the resources from the resource group.

See the `scrgadm`(1M) and `scswitch`(1M) man pages for additional information.

Note – Perform this procedure from any cluster node.

▼ How to Remove a Resource Group

1. **Become superuser on a cluster member.**

2. **Run the following command to take the resource group offline.**

```
# scswitch -F -g resource-group
```

`-F`	Switches a resource group offline.
`-g` *resource-group*	Specifies the name of the resource group to take offline.

3. **Disable all the resources that are part of the resource group.**

   You can use the `scrgadm -pv` command to view the resources in the resource group. Disable all the resources in the resource group to be removed.

```
# scswitch -n -j resource
```

`-n`	Disables the resource.
`-j` *resource*	Specifies the name of the resource to disable.

   If any dependent data-service resources exist in a resource group, you cannot disable the resource until you have disabled all the resources that depend on it.

4. **Remove all resources from the resource group.**

   Use the following `scrgadm` commands to perform the following tasks.

- Remove the resources.
- Remove the resource group.

```
# scrgadm -r -j resource
# scrgadm -r -g resource-group
```

`-r` Removes the specified resource or resource group.

`-j` *resource* Specifies the name of the resource to be removed.

`-g` *resource-group* Specifies the name of the resource group to be removed.

5. **Verify that the resource group has been removed.**

```
# scrgadm -p
```

Example – Removing a Resource Group

This example shows how to remove a resource group (`resource-group-1`) after you have removed its resource (`resource-1`).

```
# scswitch -F -g resource-group-1
# scrgadm -r -j resource-1
# scrgadm -r -g resource-group-1
```

Removing Resources

Disable the resource before removing it from a resource group.

See the `scrgadm`(1M) and `scswitch`(1M) man pages for additional information.

Note – Perform this procedure from any cluster node.

▼ How to Remove a Resource

1. **Become superuser on a cluster member.**
2. **Disable the resource that you want to remove.**

```
# scswitch -n -j resource
```

`-n` Disables the resource.

`-j` *resource* Specifies the name of the resource to disable.

3. **Remove the resource.**

```
# scrgadm -r -j resource
```

`-r` Removes the specified resource.

`-j` *resource* Specifies the name of the resource to remove.

4. **Verify that the resource has been removed.**

```
# scrgadm -p
```

Example – Removing a Resource

This example shows how to disable and remove a resource (`resource-1`).

```
# scswitch -n -j resource-1
# scrgadm -r -j resource-1
```

Switching the Current Primary of a Resource Group

Use the following procedure to switch over a resource group from its current primary to another node that will become the new primary.

See the `scrgadm`(1M) and `scswitch`(1M) man pages for additional information.

Note – Perform this procedure from any cluster node.

▼ How to Switch the Current Primary of a Resource Group

To complete this procedure, you must supply the following information.

- The name of the resource group to be switched over.
- The names of the nodes on which you want the resource group to be brought online or to remain online. These nodes must be cluster nodes that have been set up to be potential masters of the resource group to be switched. To see a list of potential primaries for the resource group, use the `scrgadm -pv` command.

1. **Become superuser on a cluster member.**

2. **Switch the primary to a potential primary.**

```
# scswitch -z -g resource-group -h nodelist
```

`-z`	Switches the specified resource group online.
`-g` *resource-group*	Specifies the name of the resource group to switch.
`-h` *nodelist*	Specifies the node or nodes on which the resource group is to be brought online or is to remain online. This resource group is then switched to be offline on all other nodes.

3. **Verify that the resource group has been switched to the new primary.**

   Run the following command and look for the output for the state of the resource group that has been switched over.

   ```
   # scstat -g
   ```

Example – Switching the Resource Group to a New Primary

This example shows how to switch a resource group (`resource-group-1`) from its current primary (`phys-schost-1`) to the potential primary (`phys-schost-2`). First, verify that the resource group is online on `phys-schost-1`, perform the switch, then verify that the group is switched to be online on `phys-schost-2`.

```
phys-schost-1# scstat -g
...
Resource Group Name:          resource-group-1
  Status
    Node Name:                phys-schost-1
    Status:                   Online

    Node Name:                phys-schost-2
    Status:                   Offline
...
phys-schost-1# scswitch -z -g resource-group-1 -h phys-schost-2
phys-schost-1# scstat -g
...
Resource Group Name:          resource-group-1
  Status
    Node Name:                phys-schost-2
    Status:                   Online

    Node Name:                phys-schost-1
    Status:                   Offline
...
```

Disabling Resources and Moving Their Resource Group Into the Unmanaged State

At times, you must bring a resource group into the unmanaged state before performing an administrative procedure on it. Before moving a resource group into the unmanaged state, you must disable all the resources that are part of the resource group and bring the resource group offline.

See the `scrgadm`(1M) and `scswitch`(1M) man pages for additional information.

Note – Perform this procedure from any cluster node.

How to Disable a Resource and Move Its Resource Group Into the Unmanaged State

To complete this procedure, you must supply the following information.

- the name of the resources to be disabled
- the name of the resource group to move into the unmanaged state

To determine the resource and resource group names that are needed for this procedure, use the `scrgadm -pv` command.

1. **Become superuser on a cluster member.**
2. **Disable the resource.**

   Repeat this step for all resources in the resource group.

```
# scswitch -n -j resource
```

`-n`	Disables the resource.
`-j` *resource*	Specifies the name of the resource to disable.

3. **Run the following command to take the resource group offline.**

```
# scswitch -F -g resource-group
```

`-F` Switches a resource group offline.

`-g` *resource-group* Specifies the name of the resource group to take offline.

4. **Bring the resource group into the unmanaged state.**

```
# scswitch -u -g resource-group
```

`-u` Puts the specified resource group in the unmanaged state.

`-g` *resource-group* Specifies the name of the resource group to move into the unmanaged state.

5. **Verify that the resources are disabled and the resource group is in the unmanaged state.**

```
# scrgadm -pv -g resource-group
```

Example – Disabling a Resource and Moving the Resource Group Into the Unmanaged State

This example shows how to disable the resource (`resource-1`) and then move the resource group (`resource-group-1`) into the unmanaged state.

```
# scswitch -n -j resource-1
# scswitch -F -g resource-group-1
# scswitch -u -g resource-group-1
# scrgadm -pv -g resource-group-1
Res Group name:                                               resource-group-1
  (resource-group-1) Res Group RG_description:                <NULL>
  (resource-group-1) Res Group management state:              Unmanaged
  (resource-group-1) Res Group Failback:                      False
  (resource-group-1) Res Group Nodelist:                      phys-schost-1
                                                              phys-schost-2
  (resource-group-1) Res Group Maximum_primaries:             2
  (resource-group-1) Res Group Desired_primaries:             2
  (resource-group-1) Res Group RG_dependencies:               <NULL>
  (resource-group-1) Res Group mode:                          Failover
  (resource-group-1) Res Group network dependencies:          True
  (resource-group-1) Res Group Global_resources_used:         All
  (resource-group-1) Res Group Pathprefix:

  (resource-group-1) Res name:                                resource-1
    (resource-group-1:resource-1) Res R_description:
    (resource-group-1:resource-1) Res resource type:          SUNW.apache
    (resource-group-1:resource-1) Res resource group name:    resource-group-1
    (resource-group-1:resource-1) Res enabled:                True
    (resource-group-1:resource-1) Res monitor enabled:        False
    (resource-group-1:resource-1) Res detached:               False
```

Displaying Resource Type, Resource Group, and Resource Configuration Information

Before you perform administrative procedures on resources, resource groups, or resource types, use the following procedure to view the current configuration settings for these objects.

See the `scrgadm`(1M) and `scswitch`(1M) man pages for additional information.

Note – Perform this procedure from any cluster node.

How to Display Resource Type, Resource Group, and Resource Configuration Information

The scrgadm command provides the following three levels of configuration status information.

- With the -p option, the output shows a very limited set of property values for resource types, resource groups, and resources.
- With the -pv option, the output shows more details on other resource type, resource group, and resource properties.
- With the -pvv option, the output provides a detailed view, including resource type methods, extension properties, and all resource and resource-group properties.

You can also view specific resource types, resource groups, and resources by using the -t, -g, and -j (resource type, resource group, and resource, respectively) options, followed by the name of the object you want to view. For example, the following command specifies that you want to view specific information on the resource apache-1 only.

```
# scrgadm -p[v[v]] -j apache-1
```

See the scrgadm(1M) man page for details.

Changing Resource Type, Resource Group, and Resource Properties

Resource groups and resources have standard configuration properties that you can change. The following procedures describe how to change these properties.

Resources also have extension properties—some of which the data service developer predefines—that you cannot change. See the individual data service chapters in this document for a list of the extension properties for each data service.

See the scrgadm(1M) man page for information on the standard configuration properties for resource groups and resources.

▼ How to Change Resource-Type Properties

To complete this procedure, you must supply the following information.

- The name of the resource type to change.
- The name of the resource-type property to change. For resource types, you can change only one property—the list of nodes on which resources of this type can be instantiated.

Note – Perform this procedure from any cluster node.

---

1. **Become superuser on a cluster member.**

2. **Run the** `scrgadm` **command to determine the name of the resource type needed for this procedure.**

```
# scrgadm -pv
```

3. **Change the resource-type property.**

   The only property that can be changed for a resource type is `Installed_node_list`.

```
# scrgadm -c -t resource-type -h installed-node-list
```

`-c` Changes the specified resource-type property.

`-t` *resource-type* Specifies the name of the resource type.

`-h` *installed-node-list* Specifies the names of nodes on which this resource type is installed.

4. **Verify that the resource-type property has been changed.**

```
# scrgadm -pv -t resource-type
```

Example – Changing a Resource-Type Property

This example shows how to change the SUNW.apache property to define that this resource type is installed on two nodes (phys-schost-1 and phys-schost-2).

```
# scrgadm -c -t SUNW.apache -h phys-schost-1,phys-schost-2
# scrgadm -pv -t SUNW.apache
Res Type name:                               SUNW.apache
  (SUNW.apache) Res Type description:        Apache Resource Type
  (SUNW.apache) Res Type base directory:     /opt/SUNWscapc/bin
  (SUNW.apache) Res Type single instance:    False
  (SUNW.apache) Res Type init nodes:         All potential masters
  (SUNW.apache) Res Type failover:           False
  (SUNW.apache) Res Type version:            1.0
  (SUNW.apache) Res Type API version:        2
  (SUNW.apache) Res Type installed on nodes: phys-schost1 phys-schost-2
  (SUNW.apache) Res Type packages:           SUNWscapc
```

▼ How to Change Resource-Group Properties

To complete this procedure, you must supply the following information.

- the name of the resource group to change
- the name of the resource-group property to change and its new value

This procedure describes the steps for changing resource-group properties. See Appendix A for a complete list of resource-group properties.

Note – Perform this procedure from any cluster node.

---

1. **Become superuser on a cluster member.**

2. **Change the resource-group property.**

```
# scrgadm -c -g resource-group -y property=new-value
```

-c	Changes the specified property.
-g *resource-group*	Specifies the name of the resource group.
-y *property*	Specifies the name of the property to change.

3. **Verify that the resource-group property has been changed.**

```
# scrgadm -pv -g resource-group
```

Example – Changing a Resource-Group Property

This example shows how to change the `Failback` property for the resource group (`resource-group-1`).

```
# scrgadm -c -g resource-group-1 -y Failback=True
# scrgadm -pv -g resource-group-1
```

▼ How to Change Resource Properties

To complete this procedure, you must supply the following information.

- the name of the resource with the property to change
- the name of the property to change

This procedure describes the steps for changing resource properties. See Appendix A for a complete list of resource-group properties.

Note – Perform this procedure from any cluster node.

---

1. **Become superuser on a cluster member.**

2. **Use the** `scrgadm -pvv` **command to view the current resource property settings.**

```
# scrgadm -pvv -j resource
```

3. **Change the resource property.**

```
# scrgadm -c -j resource -y property=new-value | -x extension-property=new-value
```

`-c` Changes the specified property.

`-j` *resource* Specifies the name of the resource.

`-y` *property*=*new-value* Specifies the name of the standard property to change.

`-x` *extension-property*=*new-value* Specifies the name of the extension property to change. For Sun-supplied data services, see the extension properties documented in the chapters on how to install and configure the individual data services.

4. **Verify that the resource property has been changed.**

```
# scrgadm pvv -j resource
```

Example – Changing a Standard Resource Property

This example shows how to change the system-defined `Start_timeout` property for the resource (`resource-1`).

```
# scrgadm -c -j resource-1 -y start_timeout=30
# scrgadm -pvv -j resource-1
```

Example – Changing an Extension Resource Property

This example shows how to change an extension property (`Log_level`) for the resource (`resource-1`).

```
# scrgadm -c -j resource-1 -x Log_level=3
# scrgadm -pvv -j resource-1
```

Clearing the STOP_FAILED Error Flag on Resources

When the Failover_mode resource property is NONE or SOFT and the STOP of a resource fails, the individual resource goes into the STOP_FAILED state and the resource group goes into the ERROR_STOP_FAILED state. You cannot bring a resource group in this state on any node online, nor can you edit it (create or delete resources, or change resource-group or resource properties).

▼ How to Clear the STOP_FAILED Error Flag on Resources

To complete this procedure, you must supply the following information.

- the name of the node where the resource is STOP_FAILED
- the name of the resource and resource group in STOP_FAILED state

See the scswitch(1M) man page for additional information.

Note – Perform this procedure from any cluster node.

1. **Become superuser on a cluster member.**

2. **Identify which resources have gone into the STOP_FAILED state and on which nodes.**

```
# scstat -g
```

3. **Manually stop the resources and their monitors on the nodes on which they are in STOP_FAILED state.**

   This step might require killing processes or running resource type-specific commands or other commands.

4. **Manually set the state of these resources to OFFLINE on all the nodes on which they were manually stopped.**

```
# scswitch -c -h nodelist -j resource -f STOP_FAILED
```

`-c` Clears the flag.

`-h` *nodelist* Specifies the node names on which the resource was running.

`-j` *resource* Specifies the name of the resource to take offline.

`-f STOP_FAILED` Specifies the flag name.

5. **Check the resource-group state on the nodes where the `STOP_FAILED` flag was cleared in Step 4.**

   The resource-group state should now be `OFFLINE` or `ONLINE`.

```
# scstat -g
```

If the resource group remains in the `ERROR_STOP_FAILED` state, which the command `scstat -g` indicates, run the following `scswitch` command to take the resource group offline on the nodes where the resource group is still in the `ERROR_STOP_FAILED` state.

```
# scswitch -F -g resource-group
```

`-F` Takes the resource group offline on all nodes that can master the group.

`-g` *resource-group* Specifies the name of the resource group to take offline.

This situation can occur if the resource group was being switched offline when the `STOP` method failure occurred and the resource that failed to stop had a dependency on other resources in the resource group. Otherwise, the resource group reverts to the `ONLINE` or `OFFLINE` state automatically after you have run the command in Step 4 on all `STOP_FAILED` resources.

Now you can switch the resource group to the `ONLINE` state.

Re-registering Preregistered Resource Types

Two preregistered resource types are `SUNW.LogicalHostname` and `SUNW.SharedAddress`. All logical hostname and shared-address resources use these resource types. You never need to register these two resource types, but you might accidentally delete them. If you have deleted resource types inadvertently, use the following procedure to re-register them.

See the `scrgadm`(1M) man page for additional information.

Note – Perform this procedure from any cluster node.

▼ How to Re-register Preregistered Resource Types

● **Re-register the resource type.**

```
# scrgadm -a -t SUNW.resource-type
```

`-a` Adds a resource type.

`-t SUNW.`*resource-type* Specifies the resource type to add (re-register). The resource type can be either `SUNW.LogicalHostname` or `SUNW.SharedAddress`.

Example – Re-registering a Preregistered Resource Type

This example shows how to re-register the `SUNW.LogicalHostname` resource type.

```
# scrgadm -a -t SUNW.LogicalHostname
```

Adding or Removing a Node to or From a Resource Group

This section contains the following two procedures.

- how to configure a cluster node to be an additional master of a resource group
- how to remove a node from a resource group

The procedures are slightly different, depending on whether you are adding or removing the node to or from a failover or scalable resource group.

Failover resource groups contain network resources that both failover and scalable services use. Each IP subnetwork connected to the cluster has its own network resource specified and included in a failover resource group. The network resource is either a logical hostname or a shared-address resource. Each network resource includes a list of NAFO groups that it uses. For failover resource groups, you must update the complete list of NAFO groups for each network resource included in the resource group (the `netiflist` resource property).

For scalable resource groups, in addition to changing the scalable group to be mastered on the new set of hosts, you must repeat the procedure for failover groups that contain the network resources that the scalable resource uses.

See the `scrgadm`(1M) man page for additional information.

Note – Run either of these procedures from any cluster node.

▼ How to Add a Node to a Resource Group

You must supply the following information to complete this procedure.

- the names and node IDs of all the cluster nodes
- the names of the resource groups to which you are adding the node
- the name of the NAFO group that will host the network resources used by the resource group on all the nodes

Also note the following points.

- Be sure to verify that the new node is already a cluster member.
- For failover resource groups, perform all the steps in the procedure "How to Add a Node to a Resource Group."

- For scalable resource groups, you must complete the tasks listed as “For Scalable Resource Groups Only.”

For Scalable Resource Groups Only

1. For each network resource that a scalable resource in the resource group uses, make the resource group where the network resource is located run on the new node (Steps 1 through 4 in the following procedure).

2. Add the new node to the list of nodes that can master the scalable resource group (the `nodelist` resource-group property) (Step 3 in the following procedure).

3. (Optional) Update the Load_balancing_weights property of the scalable resource to assign a weight to the node that you want to add to the resource group. Otherwise, the weight defaults to `1`. See the `scrgadm`(1M) man page for more information.

Procedure – How to Add a Node to a Resource Group

1. **Display the current node list and the current list of NAFO groups configured for each resource in the resource group.**

```
# scrgadm -pvv -g resource-group | grep -i nodelist
# scrgadm -pvv -g resource-group | grep -i netiflist
```

Note – The output of the command line for `nodelist` identifies the nodes by node name. The output for `netiflist` identifies them by node ID.

---

2. **Update `netiflist` for the network resources that the node addition affects.**

This step overwrites the previous value of `netiflist`, and therefore you must include all NAFO groups here. Also, you must input nodes to `netiflist` by node ID. To find the node ID, use `scconf -pv | grep “Node ID”`.

```
# scrgadm -c -j network-resource -x netiflist=netiflist
```

`-c` Changes a network resource.

`-j` *network-resource* Specifies the name of the network resource (logical hostname or shared address) being hosted on the *netiflist* entries.

`-x netiflist=`*netiflist* Specifies a comma-separated list that identifies the NAFO groups on each node. Each element in *netiflist* must be in the form of *NAFO-group-name@nodeid*.

3. **Update the node list to include all the nodes that can now master this resource group.**

   This step overwrites the previous value of `nodelist`, and therefore you must include all the nodes that can master the resource group here.

```
# scrgadm -c -g resource-group -h nodelist
```

`-c` Changes a resource group.

`-g` *resource-group* Specifies the name of the resource group to which the node is being added.

`-h` *nodelist* Specifies a comma-separated list of nodes that can master the resource group.

4. **Verify the updated information.**

```
# scrgadm -pvv -g resource-group | grep -i nodelist
# scrgadm -pvv -g resource-group | grep -i netiflist
```

Example – Adding a Node to a Resource Group

This example shows how to add a node (`phys-schost-2`) to a resource group (`resource-group-1`), which contains a logical-hostname resource (`schost-2`).

```
# scrgadm -pvv -g resource-group-1 | grep -i nodelist
(resource-group-1) Res Group Nodelist:    phys-schost-1 phys-schost-3
# scrgadm -pvv -g resource-group-1 | grep -i netiflist
(resource-group-1:schost-2) Res property name: NetIfList
(resource-group-1:schost-2:NetIfList) Res property class: extension
(resource-group-1:schost-2:NetIfList) List of NAFO interfaces on each node
(resource-group-1:schost-2:NetIfList) Res property type: stringarray
(resource-group-1:schost-2:NetIfList) Res property value: nafo0@1 nafo0@3

(Only nodes 1 and 3 have been assigned NAFO groups. You must add a NAFO group
for node 2.)

# scrgadm -c -j schost-2 -x netiflist=nafo0@1,nafo0@2,nafo0@3
# scrgadm -c -g resource-group-1 -h phys-schost-1,phys-schost-2,phys-schost-3
# scrgadm -pvv -g resource-group-1 | grep -i nodelist
(resource-group-1) Res Group Nodelist:     phys-schost-1 phys-schost-2
                                           phys-schost-3
# scrgadm -pvv -g resource-group-1 | grep -i netiflist
(resource-group-1:schost-2:NetIfList) Res property value: nafo0@1 nafo0@2
                                                          nafo0@3
```

▼ How to Remove a Node From a Resource Group

To complete this procedure, you must supply the following information.

- the names and node IDs of all the cluster nodes
- the name of the resource group or groups from which you are removing the node
- the name of the NAFO group that will host the network resources used by the resource group on all the nodes

Also note the following points.

- Be sure to verify that the resource group is *not* mastered on the node you will remove. If that's not the case, run the `scswitch` command to take the resource group offline on the node you want to remove.
- For failover resource groups, perform all the steps in the procedure "How to Remove a Node from a Resource Group."
- For scalable resource groups, you must complete the tasks listed as "For Scalable Resource Groups Only."

For Scalable Resource Groups Only

1. Remove the node from the list of nodes that can master the scalable resource group (the `nodelist` resource-group property) (Step 1 in the following procedure).
2. (Optional) For each network resource that a scalable resource in the resource group uses, update the resource group where the network resource is located to *not* be mastered on the removed node (Steps 1 through 4 in the following procedure).
3. (Optional) Update the `Load_balancing_weights` property of the scalable resource to remove the weight of the node that you want to remove from the resource group. See the `scrgadm`(1M) man page for more information.

Procedure – How to Remove a Node from a Resource Group

1. **Update the node list to include all the nodes that can now master this resource group.**

   This step removes the node and overwrites the previous value of `nodelist`. Be sure to include all the nodes that can master the resource group here.

```
# scrgadm -c -g resource-group -h nodelist
```

`-c` Changes a resource group.

`-g` *resource-group* Specifies the name of the resource group from which the node is being removed.

`-h` nodelist Specifies a comma-separated list of nodes that can master this resource group.

2. **Display the current list of NAFO groups that are configured for each resource in the resource group.**

```
# scrgadm -pvv -g resource-group | grep -i netiflist
```

Note – The output of the preceding command lines identifies the nodes by node ID.

---

3. **Update `netiflist` for network resources that the removal of the node affects.**

   This step overwrites the previous value of netiflist. Be sure to include all NAFO groups here. Also, you must input nodes to `netiflist` by node ID. Run `scconf -pv | grep "Node ID"` to find the node ID.

```
# scrgadm -c -j network-resource -x netiflist=netiflist
```

`-c` Changes a network resource.

`-j` *resource-group* Specifies the name of the network resource (logical hostname or shared address) that is being hosted on the `netiflist` entries.

`-x netiflist=`netiflist Specifies a comma-separated list that identifies the NAFO groups on each node. Each element in *netiflist* must be in the form of *NAFO-group-name@nodeid*.

4. **Verify the updated information.**

```
# scrgadm -pvv -g resource-group | grep -i nodelist
# scrgadm -pvv -g resource-group | grep -i netiflist
```

Example – Removing a Node From a Resource Group

This example shows how to remove a node (phys-schost-3) from a resource group (resource-group-1), which contains a logical-hostname resource (schost-1).

```
# scrgadm -pvv -g resource-group-1 | grep -i nodelist
(resource-group-1) Res Group Nodelist:       phys-schost-1 phys-schost-2
                                             phys-schost-3
# scrgadm -c -g resource-group-1 -h phys-schost-1,phys-schost-2
# scrgadm -pvv -g resource-group-1 | grep -i netiflist
(resource-group-1:schost-1) Res property name: NetIfList
(resource-group-1:schost-1:NetIfList) Res property class: extension
(resource-group-1:schost-1:NetIfList) List of NAFO interfaces on each node
(resource-group-1:schost-1:NetIfList) Res property type: stringarray
(resource-group-1:schost-1:NetIfList) Res property value: nafo0@1 nafo0@2
                                                          nafo0@3

(nafo0@3 is the NAFO group to be removed.)

# scrgadm -c -j schost-1 -x netiflist=nafo0@1,nafo0@2
# scrgadm -pvv -g resource-group-1 | grep -i nodelist
(resource-group-1) Res Group Nodelist:       phys-schost-1 phys-schost-2
# scrgadm -pvv -g resource-group-1 | grep -i netiflist
(resource-group-1:schost-1:NetIfList) Res property value: nafo0@1 nafo0@2
```

Synchronizing the Startups Between Resource Groups and Disk Device Groups

After a cluster boots up or services fail over to another node, global devices and cluster file systems might take awhile before they become available. However, a data service can run its START method before global devices and cluster file systems—on which the data service depends—come online. In this case, the START method times out, and you must reset the state of the resource groups that the data service uses and restart the data service manually.

The resource type SUNW.HAStorage monitors the global devices and cluster file systems and causes the START method of the other resources in the same resource group to wait until they become available. To avoid additional administrative tasks, set up SUNW.HAStorage for all the resource groups whose data-service resources depend on global devices or cluster file systems.

▼ How to Set Up SUNW.HAStorage Resource Type for New Resources

In the following example, the resource group resource-group-1 contains three data services.

- iWS, which depends on /global/resource-group-1
- Oracle, which depends on /dev/global/dsk/d5s2
- NFS, which depends on dsk/d6

To create a SUNW.HAStorage resource hastorage-1 for new resources in resource-group-1, perform the following steps.

1. **Become superuser on a cluster member.**

2. **Create the resource group** resource-group-1**.**

```
# scrgadm -a -g resource-group-1
```

3. **Register the resource type.**

```
# scrgadm -a -t SUNW.HAStorage
```

4. **Create the** SUNW.HAStorage **resource hastorage-1, and define the service paths.**

```
# scrgadm -a -j hastorage-1 -g resource-group-1 -t SUNW.HAStorage \
-x ServicePaths=
/global/resource-group-1,/dev/global/dsk/d5s2,dsk/d6
```

ServicePaths can contain the following values.

- global device group names, such as nfs-dg
- paths to global devices, such as /dev/global/dsk/d5s2 or dsk/d6
- cluster file system mount points, such as /global/nfs

5. **Enable the `hastorage-1` resource.**

```
# scswitch -e -j hastorage-1
```

6. **Add the resources (iWS, Oracle, and NFS) to `resource-group-1`, and set their dependency to `hastorage-1`.**

   For example, for iWS, run the following command.

```
# scrgadm -a -j resource -g resource-group-1 -t SUNW.iws \
-x Confdir_list=/global/iws/schost-1 \
-y Scalable=False -y Network_resources_used=schost-1 \
-y Port_list=80/tcp -y Resource_dependencies=hastorage-1
```

7. **Set `resource-group-1` to the managed state, and bring resource-group-1 online.**

```
# scswitch -Z -g resource-group-1
```

The `SUNW.HAStorage` resource type contains another extension property, `AffinityOn`, which is a Boolean that specifies whether `SUNW.HAStorage` must perform an affinity switchover for the global devices and cluster file systems defined in `ServicePaths`. See the `SUNW.HAStorage`(5) man page for details.

▼ How to Set Up `SUNW.HAStorage` Resource Type for Existing Resources

Perform the following steps to create a `SUNW.HAStorage` resource for existing resources.

1. **Register the resource type.**

```
# scrgadm -a -t SUNW.HAStorage
```

2. **Create the `SUNW.HAStorage` resource `hastorage-1`.**

```
# scrgadm -a -g resource-group -j hastorage-1 -t SUNW.HAStorage \
-x ServicePaths= … -x AffinityOn=True
```

3. **Enable the `hastorage-1` resource.**

```
# scswitch -e -j hastorage-1
```

4. **Set up the dependency for each of the existing resources, as required.**

```
# scrgadm -c -j resource -y Resource_Dependencies=hastorage-1
```

APPENDIX **A**

Standard Properties

This appendix describes the standard resource type, resource group, and resource properties. This appendix also describes the resource property attributes available for changing system-defined properties and creating extension properties.

This appendix contains the following sections.

- "Resource-Type Properties" on page 243
- "Resource Properties" on page 247
- "Resource-Group Properties" on page 255
- "Resource Property Attributes" on page 259

Note – The property values, such as `True` and `False`, are *not* case sensitive.

Resource-Type Properties

TABLE A-1 describes the resource-type properties that Sun Cluster defines. The property values are categorized as follows (in the Category column).

- **Required** – The property requires an explicit value in the Resource Type Registration (RTR) file or the object to which the property belongs cannot be created. A blank or the empty string is not allowed as a value.
- **Conditional** – To exist, the property must be declared in the RTR file. Otherwise, the RGM does not create the property and the property is not available to administrative utilities. A blank or the empty string is allowed. If the property is declared in the RTR file but no value is specified, the RGM supplies a default value.

- **Conditional/Explicit** – To exist, the property must be declared in the RTR file with an explicit value. Otherwise, the RGM does not create the property, and the property is not available to administrative utilities. A blank or the empty string is not allowed.
- **Optional** – The property can be declared in the RTR file. If the property is not declared in the RTR file, the RGM creates the property and supplies a default value. If the property is declared in the RTR file but no value is specified, the RGM supplies the same default value as if the property were not declared in the RTR file.

Resource-type properties are not updatable by administrative utilities with the exception of `Installed_nodes`, which cannot be declared in the RTR file and must be set by the administrator.

TABLE A-1 Resource-Type Properties *(1 of 4)*

Property Name	Description	Updatable	Category
`API_version` (integer)	The version of the resource management API that this resource type implementation uses. The default for SC 3.0 is 2.	N	Optional
`BOOT` (string)	An optional callback method – The path to the program that the RGM invokes on a node, which joins or rejoins the cluster when a resource of this type is already managed. This method is expected to do initialization actions for resources of this type similar to the `INIT` method.	N	Conditional /Explicit
`Failover` (Boolean)	`True` indicates that resources of this type cannot be configured in any group that can be online on multiple nodes at once. The default is `False`.	N	Optional
`FINI` (string)	An optional callback method – The path to the program that the RGM invokes when a resource of this type is removed from RGM management.	N	Conditional /Explicit
`INIT` (string)	An optional callback method – The path to the program that the RGM invokes when a resource of this type becomes managed by the RGM.	N	Conditional /Explicit
`Init_nodes` (enum)	The values can be `RG_primaries` (just the nodes that can master the resource) or `RT_installed_nodes` (all nodes on which the resource type is installed). Indicates the nodes on which the RGM is to call the `INIT`, `FINI`, `BOOT` and `VALIDATE` methods. The default value is `RG_primaries`.	N	Optional

TABLE A-1 Resource-Type Properties *(2 of 4)*

Property Name	Description	Updatable	Category
Installed_nodes (string array)	A list of the cluster node names on which the resource type is allowed to be run. The RGM automatically creates this property. The cluster administrator can set the value. You cannot declare this property in the RTR file. The default is all cluster nodes.	Y	Configurable by cluster administrator
Monitor_check (string)	An optional callback method – The path to the program that the RGM invokes before doing a monitor-requested failover of a resource of this type.	N	Conditional /Explicit
Monitor_start (string)	An optional callback method – The path to the program that the RGM invokes to start a fault monitor for a resource of this type.	N	Conditional /Explicit
Monitor_stop (string)	A callback method that is required if Monitor_start is set – The path to the program that the RGM invokes to stop a fault monitor for a resource of this type.	N	Conditional /Explicit
Pkglist (string array)	An optional list of packages that are included in the resource type installation.	N	Conditional /Explicit
Postnet_stop (string)	An optional callback method – The path to the program that the RGM invokes after calling the STOP method of any network-address resources (Network_resources_used) that a resource of this type is dependent on. This method is expected to do STOP actions that must be done after the network interfaces are configured down.	N	Conditional /Explicit
Prenet_start (string)	An optional callback method – The path to the program that the RGM invokes before calling the START method of any network-address resources (Network_resources_used) that a resource of this type is dependent on. This method is expected to do START actions that must be done before network interfaces are configured up.	N	Conditional /Explicit
RT_basedir (string)	The directory path that is used to complete relative paths for callback methods. This path is expected to be set to the installation location for the resource type packages. The path must be complete, that is, the path must start with a forward slash (/). This property is not required if all the method path names are absolute.	N	Required (unless all method path names are absolute)
RT_description (string)	A brief description of the resource type. The default is the empty string.	N	Conditional

TABLE A-1 Resource-Type Properties *(3 of 4)*

Property Name	Description	Updatable	Category
`Resource_type` (string)	The name of the resource type. Must be unique in the cluster installation. You must declare this property as the first entry in the RTR file. Otherwise, registration of the resource type fails. In addition, you can specify `Vendor_id` to identify the resource type. `Vendor_id` serves as a prefix that is separated from a resource type name by a ".", for example, `SUNW.http`. You can completely identify the resource type with `Resource_type` and Vendor_id or omit `Vendor_id`. For example, both `SUNW.http` and `http` are valid. If you specify the `Vendor_id`, use the stock symbol for the company that defines the resource type. If two resource types in the cluster differ only in the `Vendor_id` prefix, the use of the abbreviated name fails. The default is the empty string.	N	Required
`RT_version` (string)	An optional version string of this resource type implementation.	N	Conditional /Explicit
`Single_instance` (Boolean)	If `True`, indicates that only one resource of this type can exist in the cluster. Hence, the RGM allows only one resource of this type to run cluster-wide at one time. The default value is `False`.	N	Optional
`START` (string)	A callback method – The path to the program that the RGM invokes to start a resource of this type.	N	Required (unless the RTR file declares a `PRENET_START` method)
`STOP` (string)	A callback method – The path to the program that the RGM invokes to stop a resource of this type.	N	Required (unless the RTR file declares a `POSTNET_STOP` method)

TABLE A-1 Resource-Type Properties *(4 of 4)*

Property Name	Description	Updatable	Category
`UPDATE` (string)	An optional callback method – The path to the program that the RGM invokes when properties of a running resource of this type are changed.	N	Conditional /Explicit
`VALIDATE` (string)	An optional callback method – The path to the program that will be invoked to check values for properties of resources of this type.	N	Conditional /Explicit
`Vendor_ID` (string)	See the `Resource_type` property.	N	Conditional

Resource Properties

TABLE A-2 describes the resource properties that Sun Cluster defines. The property values are categorized as follows (in the Category column).

- **Required** – The administrator must specify a value when creating a resource with an administrative utility.
- **Optional** – If the administrator does not specify a value when creating a resource group, the system supplies a default value.
- **Conditional** – The RGM creates the property only if the property is declared in the RTR file. Otherwise, the property does not exist and is not available to system administrators. A conditional property declared in the RTR file is optional or required, depending on whether a default value is specified in the RTR file. For details, see the description of each conditional property.
- **Query-only** – An administrative tool cannot directly set this property.

TABLE A-2 also lists whether and when resource properties are updatable (in the Updatable column), as follows

`None` or `False`	Never.
`True` or `Anytime`	Any time.
`At_creation`	When the resource is added to a cluster.
`When_disabled`	When the resource is disabled.

TABLE A-2 Resource Properties *(1 of 8)*

Property Name	Description	Updatable	Category
Cheap_probe_interval (integer)	The number of seconds between invocations of a quick fault probe of the resource. Only the RGM creates this property, and the property is only available to the administrator if the property is declared in the RTR file. This property is optional if a default value is specified in the RTR file. If the Tunable attribute is not specified in the resource type file, the Tunable value for the property is When_disabled. This property is required if the Default attribute is not specified in the property declaration in the RTR file.	When disabled	Conditional
Extension properties	Extension properties as declared in the RTR file of the resource's type. The implementation of the resource type defines these properties. For information on the individual attributes you can set for extension properties. see TABLE A-4.	Depends on the specific property	Conditional
Failover_mode (enum)	Controls whether the RGM relocates a resource group or aborts a node in response to a failure of a START or STOP method call on the resource. None indicates that the RGM should just set the resource state on method failure and wait for operator intervention. Soft indicates that failure of a START method should cause the RGM to relocate the resource's group to a different node while failure of a STOP method should cause the RGM to set the resource state and wait for operator intervention. Hard indicates that failure of a START method should cause the relocation of the group and failure of a STOP method should cause the forcible stop of the resource by aborting the cluster node. The default is None.	Any time	Optional

TABLE A-2 Resource Properties *(2 of 8)*

Property Name	Description	Updatable	Category
`Load_balancing_policy` (string)	A string that defines the load-balancing policy in use. This property is used only for scalable services. The RGM automatically creates this property if the `Scalable` property is declared in the RTR file. Load_balancing_policy can take the following values. `Lb_weighted` (the default). The load is distributed among various nodes according to the weights set in the `Load_balancing_weights` property. `Lb_sticky`. A given client (identified by the client IP address) of the scalable service is always sent to the same node of the cluster. `Lb_sticky_wild`. A given client (identified by the client's IP address) that connects to an IP address of a wildcard sticky service is always sent to the same cluster node regardless of the port number the client is coming to. The default value is `Lb_weighted`.	At creation	Conditional Optional

TABLE A-2 Resource Properties *(3 of 8)*

Property Name	Description	Updatable	Category
`Load_balancing_weights` (string array)	For scalable resources only. The RGM automatically creates this property if the `Scalable` property is declared in the RTR file. The format is *weight@node,weight@node*, where *weight* is an integer that reflects the relative portion of load distributed to the specified *node*. The fraction of load distributed to a node is the weight for this node divided by the sum of all weights. For example, `1@1,3@2` specifies that node 1 receives 1/4 of the load and node 2 receives 3/4. The empty string (""), the default, sets a uniform distribution. Any node that is not assigned an explicit weight, receives a default weight of 1. If the `Tunable` attribute is not specified in the resource type file, the `Tunable` value for the property is `Anytime`. Changing this property revises the distribution for new connections only. The default value is the empty string ("").	Any time	Conditional Optional
method`_timeout` for each callback method in the Type. (integer)	A time lapse, in seconds, after which the RGM concludes that an invocation of the method has failed. The default is 3,600 (one hour) if the method itself is declared in the RTR file.	Any time	Conditional Optional
`Monitored_switch` (enum)	The RGM sets to `Enabled` or `Disabled` if the cluster administrator enables or disables the monitor with an administrative utility. If `Disabled`, the monitor does not have its `START` method called until the value is set to enabled again. If the resource does not have a monitor callback method, this property does not exist. The default is `Enabled`.	Never	Query-only

TABLE A-2 Resource Properties *(4 of 8)*

Property Name	Description	Updatable	Category
`Network_resources_used` (string array)	A list of logical hostname or shared address network resources that the resource uses. For scalable services, this property must refer to shared address resources that exist in a separate resource group. For failover services, this property refers to logical hostname or shared address resources that exist in the same resource group. The RGM automatically creates this property if the `Scalable` property is declared in the RTR file. If `Scalable` is not declared in the RTR file, `Network_resources_used` is unavailable unless the property is explicitly declared in the RTR file. If the `Tunable` attribute is not specified in the resource type file, the `Tunable` value for the property is `At_creation`.	At creation	Conditional Required
`On_off_switch` (enum)	The RGM sets to `Enabled` or `Disabled` if the cluster administrator enables or disables the resource with an administrative utility. If disabled, a resource has no callbacks invoked until the property is enabled again. The default is `Disabled`.	Never	Query-only
`Port_list` (string array)	A comma-separated list of port numbers on which the server is listening. Appended to each port number is the protocol that the port uses, for example, `Port_list=80/tcp`. If the `Scalable` property is declared in the RTR file, the RGM automatically creates `Port_list`. Otherwise, this property is unavailable unless the property is explicitly declared in the RTR file. For specifics on setting up this property for Apache, see the Apache chapter in the *Sun Cluster 3.0 U1 Data Services Installation and Configuration Guide*.	At creation	Conditional Required
`R_description` (string)	A brief description of the resource. The default is the empty string.	Any time	Optional

TABLE A-2 Resource Properties *(5 of 8)*

Property Name	Description	Updatable	Category
`Resource_dependencies` (string array)	A list of resources in the same group that must be online in order for this resource to be online. This resource cannot be started if the start of any resource in the list fails. When bringing the group offline, this resource is stopped before those in the list. Resources in the list are not allowed to be disabled unless this resource is disabled first. The default is the empty list.	Any time	Optional
`Resource_dependencies_weak` (string array)	A list of resources in the same group that determines the order of method calls within the group. The RGM calls the `START` methods of the resources in this list before the `START` method of this resource and the `STOP` methods of this resource before the `STOP` methods of those in the list. The resource can still be online if those in the list fail to start or are disabled. The default is the empty list.	Any time	Optional
`Resource_name` (string)	The name of the resource instance. Must be unique within the cluster configuration and cannot be changed after a resource has been created.	Never	Required
`Resource_state`, on each cluster node (enum)	The RGM-determined state of the resource on each cluster node. Possible states are `Online`, `Offline`, `Stop_failed`, `Start_failed`, `Monitor_failed`, and `Online_not_monitored`. This property is not user configurable.	Never	Query-only

TABLE A-2 Resource Properties *(6 of 8)*

Property Name	Description	Updatable	Category
`Retry_count` (integer)	The number of times a monitor attempts to restart a resource if the resource fails. Only the RGM creates this property, and the property is only available to the administrator if the property is declared in the RTR file. Setting the property is optional if a default value is specified in the RTR file. If the `Tunable` attribute is not specified in the resource type file, the `Tunable` value for the property is `When_disabled`. This property is required if the `Default` attribute is not specified in the property declaration in the RTR file.	When disabled	Conditional
`Retry_interval` (integer)	The number of seconds over which to count attempts to restart a failed resource. The resource monitor uses this property in conjunction with `Retry_count`. Only the RGM creates this property, and this property is only available to the administrator if the property is declared in the RTR file. Setting the property is optional if a default value is specified in the RTR file. If the `Tunable` attribute is not specified in the resource type file, the `Tunable` value for the property is `When_disabled`. This property is required if the `Default` attribute is not specified in the property declaration in the RTR file.	When disabled	Conditional

TABLE A-2 Resource Properties *(7 of 8)*

Property Name	Description	Updatable	Category
Scalable (Boolean)	Indicates whether the resource is scalable. If this property is declared in the RTR file, the RGM automatically creates scalable service properties, including Network_resources_used, Port_list, Load_balancing_policy, and Load_balancing_weights, for resources of that type. These properties have their default values unless they are explicitly declared in the RTR file. The default for Scalable—when the property is declared in the RTR file—is True. When this property is declared in RTR file, the Tunable attribute must be set to At_creation or resource creation fails. If this property is not declared in the RTR file, the resource is not scalable, the cluster administrator cannot tune this property, and no scalable service properties are set by the RGM. However, you can explicitly declare the Network_resources_used and Port_list properties in the RTR file, if desired, because they can be useful in a non-scalable service as well as in a scalable service.	At creation	Optional
Status, on each cluster node (enum)	The resource monitor sets. Possible values are OK, degraded, faulted, unknown, and offline. The RGM sets the value to unknown when the resource is brought online and to Offline when resource is brought offline.	Never	Query-only

TABLE A-2 Resource Properties *(8 of 8)*

Property Name	**Description**	**Updatable**	**Category**
`Status_msg`, on each cluster node (string)	The resource monitor sets at the same time as the `Status` property. This property is settable per resource per node. The RGM sets the property to the empty string when the resource is brought offline.	Never	Query-only
`Thorough_probe_interval` (integer)	The number of seconds between invocations of a high-overhead fault probe of the resource. Only the RGM creates this property, and this property is only available to the administrator if the property is declared in the RTR file. This property is optional if a default value is specified in the RTR file. If the `Tunable` attribute is not specified in the resource type file, the `Tunable` value for the property is `When_disabled`. This property is required if the `Default` attribute is not specified in the property declaration in the RTR file.	When disabled	Conditional
`Type` (string)	The resource type of which this resource is an instance.	Never	Required

Resource-Group Properties

TABLE A-3 describes the resource-group properties that Sun Cluster defines. The property values are categorized as follows (in the Category column).

- **Required** – The administrator must specify a value when creating a resource group with an administrative utility.
- **Optional** – If the administrator does not specify a value when creating a resource group, the system supplies a default value.
- **Query-only** – An administrative tool cannot set the property directly.

The Updatable column shows whether the property is updatable (Y) or not (N) after the property is initially set.

TABLE A-3 Resource-Group Properties

Property Name	Description	Updatable	Category
`Desired_primaries` (integer)	The number of nodes where the group is desired to be online at once. The default is 1. If the `RG_mode` property is `Failover`, the value of this property must be no greater than 1. If the `RG_mode` property is `Scalable`, a value greater than 1 is allowed.	Y	Optional
`Failback` (Boolean)	A Boolean value that indicates whether to recalculate the set of nodes where the group is online when the cluster membership changes. A recalculation can cause the RGM to bring the group offline on less preferred nodes and online on more preferred nodes. The default is `False`.	Y	Optional
`Global_resources_used` (string array)	Indicates whether any resource in this resource group uses cluster file systems. Legal values that the administrator can specify are an asterisk (*) to indicate all global resources, and the empty string (“”) to indicate no global resources. The default is all global resources.	Y	Optional
`Implicit_network_dependencies` (Boolean)	A Boolean value that indicates, when `True`, that the RGM should enforce implicit strong dependencies of non-network-address resources on network-address resources within the group. Network-address resources include the logical hostname and shared address resource types. In a scalable resource group, this property has no effect because a scalable resource group does not contain any network-address resources. The default is `True`.	Y	Optional

TABLE A-3 Resource-Group Properties *(Continued)*

Property Name	Description	Updatable	Category
Maximum_primaries (integer)	The maximum number of nodes where the group might be online at once. The default is 1. If the RG_mode property is Failover, the value of this property must be no greater than 1. If the RG_mode property is Scalable, a value greater than 1 is allowed.	Y	Optional
Nodelist (string array)	A list of cluster nodes where the group can be brought online in order of preference. These nodes are known as the potential primaries or masters of the resource group. The default is the list of all cluster nodes.	Y	Optional
Pathprefix (string)	A directory in the cluster file system in which resources in the group can write can write essential administrative files. Some resources might require this property. Make Pathprefix unique for each resource group. The default is the empty string.	Y	Optional

TABLE A-3 Resource-Group Properties *(Continued)*

Property Name	Description	Updatable	Category
`Pingpong_interval` (integer)	A non-negative integer value (in seconds) that the RGM uses to determine where to bring the resource group online in a reconfiguration or as the result of an `scha_control giveover` command or function being executed. In a reconfiguration, if the resource group fails to come online more than once within the past `Pingpong_interval` seconds on a particular node (because the resource's `START` or `PRENET_START` method exited non-zero or timed out), that node is considered ineligible to host the resource group and the RGM looks for another master. If a call to a resource's `scha_control`(1ha)(3ha) command or function causes the resource group to be brought offline on a particular node within the past `Pingpong_interval` seconds, that node is ineligible to host the resource group as the result of a subsequent call to `scha_control` originating from another node. The default value is 3,600 (one hour).	Y	Optional
`Resource_list` (string array)	The list of resources that are contained in the group. The administrator does not set this property directly. Rather, the RGM updates this property as the administrator adds or removes resources from the resource group. The default is the empty list.	N	Query-only
`RG_dependencies` (string array)	Optional list of resource groups indicating a preferred ordering for bringing other groups online or offline on the same node. Has no effect if the groups are brought online on different nodes. The default is the empty list.	Y	Optional
`RG_description` (string)	A brief description of the resource group. The default is the empty string.	Y	Optional

TABLE A-3 Resource-Group Properties *(Continued)*

Property Name	Description	Updatable	Category
RG_mode (enum)	Indicates whether the resource group is a failover or scalable group. If the value is Failover, the RGM sets the Maximum_primaries property of the group to 1 and restricts the resource group to being mastered by a single node. If the value of this property is Scalable, the RGM allows the Maximum_primaries property to have a value greater than 1, meaning that multiple nodes can master the group simultaneously. The RGM does not allow a resource whose Failover property is True to be added to a resource group whose RG_mode is Scalable. The default is Failover if Maximum_primaries is 1 and Scalable if Maximum_primaries is greater than 1.	N	Optional
RG_name (string)	The name of the resource group. Must be unique within the cluster.	N	Required
RG_state, on each cluster node (enum)	The RGM sets to Online, Offline, Pending_online, Pending_offline or Error_stop_failed to describe the state of the group on each cluster node. A group can also exist in an unmanaged state when the group is not under the control of the RGM. This property is not user configurable. The default is Offline.	N	Query-only

Resource Property Attributes

TABLE A-4 describes the resource property attributes that can be used to change system-defined properties or to create extension properties.

Caution – You cannot specify NULL or the empty string (“”) as the default value for boolean, enum, or int types.

TABLE A-4 Resource Property Attributes

Property	Description
Property	The name of the resource property.
Extension	If used, indicates that the RTR file entry declares an extension property that the resource type implementation defines. Otherwise, the entry is a system-defined property.
Description	A string annotation intended to be a brief description of the property. The description attribute cannot be set in the RTR file for system-defined properties.
Type of the property	Allowable types are string, boolean, int, enum, and stringarray. You cannot set the type attribute in an RTR file entry for system-defined properties. The type determines acceptable property values and the type-specific attributes that are allowed in the rtr file entry. an enum type is a set of string values.
Default	Indicates a default value for the property.
Tunable	Indicates when the cluster administrator can set the value of this property in a resource. Can be set to None or False to prevent the administrator from setting the property. Values that allow administrator tuning are True or Anytime (at any time), At_creation (only when the resource is created), or When_disabled (when the resource is offline). The default is True (Anytime).
Enumlist	For an enum type, a set of string values permitted for the property.
Min	For an int type, the minimal value permitted for the property.
Max	For an int type, the maximum value permitted for the property.
Minlength	For string and stringarray types, the minimum string length permitted.
Maxlength	For string and stringarray types, the maximum string length permitted.
Array_minsize	For stringarray type, the minimum number of array elements permitted.
Array_maxsize	For stringarray type, the maximum number of array elements permitted.

APPENDIX **B**

Legal RGM Names and Values

This appendix lists the requirements for legal characters for RGM names and values.

RGM Legal Names

Resource Group Manager (RGM) names fall into the following five categories.

- resource-group names
- resource-type names
- resource names
- property names
- enumeration literal names

Except for resource type names, all names must comply with the following rules.

- must be in ASCII
- must start with a letter
- can contain upper and lowercase letters, digits, dashes (-), and underscores (_)
- must not exceed 255 characters

A resource type name can be a simple name (specified by the `Resource_type` property in the RTR file) or a complete name (specified by the `Vendor_id` and `Resource_type` properties in the RTR file). When you specify both these properties, the RGM inserts a period between the `Vendor_id` and `Resource_type` to form the complete name. For example, if `Vendor_id=SUNW` and `Resource_type=sample`, the complete name is `SUNW.sample`. This instance is the only case where a period is a legal character in an RGM name.

RGM Values

RGM values fall into two categories—property values and description values—both of which share the same rules, as follows.

- Values must be in ASCII.
- The maximum length of a value is 4 megabytes minus 1, that is, 4,194,303 bytes.
- Values cannot contain any of the following characters.
  - null
  - newline
  - comma
  - semicolon