Sun Cluster 3.0 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 Number 806-1421
November 2000, Revision A

Please Recycle

Contents

Preface

The *Sun™ Cluster 3.0 Data Services Installation and Configuration Guide* contains procedures for installing and configuring 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 Installation Guide*	806-1419
Hardware	*Sun Cluster 3.0 Hardware Guide*	806-1420
API development	*Sun Cluster 3.0 Data Services Developers' Guide*	806-1422
Administration	*Sun Cluster 3.0 System Administration Guide*	806-1423
Cluster concepts	*Sun Cluster 3.0 Concepts*	806-1424
Release Notes	*Sun Cluster 3.0 Release Notes*	806-1428

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 email 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. It contains the following sections:

For conceptual information about data services, resource types, resources, and resource groups, see the *Sun Cluster 3.0 Concepts* document.

For applications not currently offered as Sun Cluster data services, see the *Sun Cluster 3.0 Data Services Developers' Guide* for information on how to configure those applications to become highly available data services.

Sun Cluster Data Services Installation and Configuration Tasks

Table 1–1 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 Sun Cluster HA for Oracle	Chapter 2
Install and configure Sun Cluster HA for iPlanet™ Web Server	Chapter 3
Install and configure Sun Cluster HA for Netscape Directory Server	Chapter 4
Install and configure Sun Cluster HA for Apache	Chapter 5
Install and configure Sun Cluster HA for DNS	Chapter 6
Install and configure Sun Cluster HA for NFS	Chapter 7
Install and configure Sun Cluster HA for Oracle Parallel Server	Chapter 8
Administer data services	Chapter 9
Understand data service fault monitors	Chapter 10

Configuration Guidelines for Sun Cluster Data Services

This section provides configuration guidelines for Sun Cluster data services.

Determining the Location of the Application Binaries

The two locations where you can install the application software and application configuration files are: on the local disks of each cluster node or on the cluster file system. 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 cluster. The disadvantage is that you then have several copies of the software and configuration files to maintain and administer.

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.

For information on creating cluster file systems, see the planning chapter of the *Sun Cluster 3.0 Installation Guide.*

Verifying the `nsswitch.conf` File Contents

The `nsswitch.conf` file is the configuration file for name service lookups. This file determines which databases within the Solaris environment to use for name service lookups and in what order to consult the databases.

For some data services, direct “group” lookups to “files” first. Change the “group” line in the file so that the “files” entry is listed first. To determine whether you need to change the “group” line, see the chapter for the data service you are configuring.

For additional information on how to configure `nsswitch.conf` for the Sun Cluster environment, see the planning chapter in the *Sun Cluster 3.0 Installation Guide.*

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. To determine whether any special considerations apply, see the chapter for the data service you are configuring.

For information on creating cluster file systems, see the planning chapter of the *Sun Cluster 3.0 Installation Guide.*

Relationship Between Resource Groups and Disk Device Groups

Sun Cluster has the concept of a *node list* for disk device groups and resource groups. These node lists are ordered lists of nodes that are potential masters of the disk device group or resource group. Associated with the node list is a failback policy. This policy describes the action to be taken when the node that masters the disk device group or resource group (the *primary*) leaves the configuration and later rejoins—that is, whether the disk device group or resource group is once again mastered by the primary when it rejoins the cluster.

To ensure high availability of a failover resource group, make the group's node list match the node list of any associated disk device group. 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 can contain exactly two nodes only. 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 `dg-schost-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, `rg-schost-1`, which uses `dg-schost-1` to hold its application data. When you set up `rg-schost-1`, also specify `phys-schost-1` and `phys-schost-2` for its node list and set its failback policy to `True`.

To ensure high availability of a scalable resource group, make the 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 node connected to the data is up and in the cluster, the scalable resource group is running on those same nodes, making the scalable services available also.

For information on setting up disk device groups, refer to the *Sun Cluster 3.0 Installation Guide*. For more details on the relationship between disk device groups and resource groups, see the *Sun Cluster 3.0 Concepts* document.

`SUNW.HAStorage` Resource Type

The resource type `SUNW.HAStorage` serves the following purposes:

- Coordinates the boot order by monitoring the global devices and cluster file systems and causing the `START` methods of the other resources in the same resource group that contains 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.

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 `SUNW.HAStorage`. This requirement is to coordinate the boot order between the storage and the data services.
- If your data service is disk-intensive, such as Sun Cluster HA for Oracle and Sun Cluster HA for NFS, then we recommend that you 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`. That way, the resource groups and disk device groups are colocated on the same node. On the other hand, if your data service is *not* disk-intensive—such as one that reads all its files at startup (for example, Sun Cluster HA for DNS), configuring `SUNW.HAStorage` is optional.
- If your cluster contains only two nodes, configuring `SUNW.HAStorage` is optional. However, if you plan to add nodes and run scalable services later on, you must configure `SUNW.HAStorage` at that time. To get prepared, you can go ahead and set up `SUNW.HAStorage` now and add nodes to the node list later.

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

For the procedure on how to set up `SUNW.HAStorage`, see “How to Set Up `SUNW.HAStorage` Resource Type for New Resources” on page 168. 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 allowed to be run.
2. `nodelist` — A property of a resource group. 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, you configure only one resource group node list. For scalable services, you configure two resource groups and thus two node lists. One lists the nodes on which the shared addresses are hosted. This list is a failover resource group on which the scalable resources depend. The other is a list of nodes on which the application resources are hosted. The node list for the resource group that contains the shared addresses must be a superset of the node list for the application resources because the application resources depend on the shared addresses.
3. `auxnodelist` — A property of a resource group that contains shared addresses. 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. These auxiliary nodes can never serve as masters of the resource group. This list pertains to scalable services only.

Overview of the Installation and Configuration Process

You use three procedures to install and configure data services, as follows:

- Install the data service packages from the Sun Cluster data services CD.
- Install and configure the application to run in the cluster environment.
- Configure the resources and resource groups used by the data service. When you configure a data service, you specify the resource types, resources, and resource groups to be managed by the Resource Group Manager (RGM). These procedures are described in the chapters for each of the data services.

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

Installation and Configuration Task Flow

Table 1–2 shows a task map of the procedure for installing and configuring a Sun Cluster failover data service.

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

Task	**For Instructions, Go to …**
Install Solaris and Sun Cluster	*Sun Cluster 3.0 Installation Guide*
Set up NAFO groups	*Sun Cluster 3.0 Installation Guide*
Set up multihost disks	*Sun Cluster 3.0 Installation Guide*
Plan resources and resource groups	*Sun Cluster 3.0 Release Notes*
Decide the location for application binaries; 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 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

This 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. Sun Cluster HA for Oracle is a Sun-supplied data service so 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, it uses logical host name network resources, which are the IP addresses that fail over from a primary node to a

secondary node. You place the logical host name 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 HA Oracle data service run on the cluster, you must define the following objects:

- `LogicalHostname` resource type — This resource type is built in so you need not define it explicitly.
- Oracle resource types — Sun Cluster HA for Oracle defines two Oracle resource types, a database server and a listener.
- Logical host name resources — These resources host the IP addresses that fail over in the event of a node failure.
- Oracle resources — You must specify two resource instances for Sun Cluster HA for Oracle, a server and a listener.
- Failover resource group — This container is composed of the Oracle server and listener and logical host name resources that will fail over as a group.

CHAPTER 2

Installing and Configuring Sun Cluster HA for Oracle

This chapter provides instructions for setting up and administering the Sun Cluster HA for Oracle data service on your Sun Cluster nodes.

This chapter contains the following procedures:

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

Installing and Configuring Sun Cluster HA for Oracle

Table 2–1 lists the 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 Sun Cluster HA for Oracle	"Preparing to Install Sun Cluster HA for Oracle" on page 25
Install the Oracle application software	"Installing the Oracle Server Software" on page 25
Create an Oracle database	"Creating an Oracle Database" on page 28
Set up Oracle database permissions	"Setting Up Oracle Database Permissions" on page 30
Install the Sun Cluster HA for Oracle packages	"Installing Sun Cluster HA for Oracle Packages" on page 34
Register resource types and configure resource groups and resources	"Registering and Configuring Sun Cluster HA for Oracle" on page 35
Verify the Sun Cluster HA for Oracle installation	"Verifying the Sun Cluster HA for Oracle Installation" on page 40
Configure extension properties	"Configuring Sun Cluster HA for Oracle Extension Properties" on page 41

Preparing to Install Sun Cluster HA for Oracle

To prepare Sun Cluster nodes for Sun Cluster HA for Oracle installation, you must select an install location for the Oracle application files (Oracle binaries, configuration files, and parameter files) and for the database-related files (control file, redo logs, and data files).

Table 2–2 shows the possible install location combinations—either on the cluster file system, raw global devices, or the local disk of the physical host. For the advantages and disadvantages of placing the Oracle binaries on the local versus the global file system, see "Determining the Location of the Application Binaries" on page 17.

TABLE 2–2 Location of Oracle Application and Database Files

Oracle Binaries, Configuration, and Parameter Files	Database Control, Redo Logs, and Data Files
Local file system	Cluster file system
Local file system	Raw global devices
Cluster file system	Raw global devices
Cluster file system	Cluster file system

Installing the Oracle Server Software

Use the procedures in this section to do the following:

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

Before setting up Sun Cluster HA for Oracle, you must have configured the Sun Cluster software on each node by using the procedures described in the *Sun Cluster 3.0 Installation Guide*.

▼ 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.

Consult the Oracle documentation before performing this procedure.

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

1. **Become superuser on all the nodes in the cluster.**

2. **Set up the `/etc/nsswitch.conf` files as follows so that the data service starts and stops correctly in case of switchovers or failovers.**

   On each node that can master the logical host running Sun Cluster HA for Oracle, 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
```

   Sun Cluster HA for Oracle uses the `su` *user* command when starting and stopping the database node. The above settings ensure that the `su`(1M) command does not refer to NIS/NIS+ when the network information name service is not available because of a failure of the public network on the cluster node.

3. **Set up the cluster file system for Sun Cluster HA for Oracle.**

   If you are using raw devices to contain the databases, you must configure the global devices for raw device access. For information on configuring global devices, see the *Sun Cluster 3.0 Installation Guide*.

   When using Solstice™ DiskSuite, configure Oracle to use UFS logging or raw mirrored meta devices. For more information on setting up raw mirrored meta devices, see the Solstice DiskSuite documentation.

4. **Prepare the Oracle home directory for Oracle installation.**

   Choose a location for the `$ORACLE_HOME` directory on either a local or multihost disk.

Note - If you choose to install the Oracle binaries on a local disk of the physical hosts, use a separate disk, if possible. Doing so prevents Oracle binaries from being overwritten if the operating environment is reinstalled.

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

   Normally, this group is named `dba`. Verify that `root` and `oracle_id` 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 Sun Cluster HA for Oracle. For example:

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

   You can make the group entries in a network name service (for example, NIS or NIS+). You should also make entries in the local `/etc` files to eliminate dependency on the network name service.

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

   Normally, oracle_id is `oracle`. The following command line updates the `/etc/passwd` and `/etc/shadow` files as required.

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

   Ensure that oracle_id is the same on all the nodes that run Sun Cluster HA for Oracle.

▼ How to Install the Oracle Software

1. **Become superuser on a node in the cluster.**

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

   You can install Oracle binaries on either the local disks of the physical hosts or on the cluster file system. For more information about installation locations, see “Preparing to Install Sun Cluster HA for Oracle” on page 25.

   If you plan to install Oracle software on the cluster file system, you must first start Sun Cluster and take ownership of the disk device group.

3. **Install the Oracle software.**

   Regardless of where the Oracle software is to be installed, on each node, modify the `/etc/system` files according to standard Oracle installation procedures. Reboot afterward.

Log in as `oracle_id` to ensure ownership of the entire directory before performing this step. For instructions on installing Oracle software, refer to the appropriate Oracle installation and configuration guides.

▼ How to Verify the Oracle Installation

1. **Verify that `$ORACLE_HOME/bin/oracle` is owned by the `oracle_id` user and the `dba` group.**

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

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

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

Where to Go from Here

When you have completed the work in this section, go to "Creating an Oracle Database" on page 28.

Creating an Oracle Database

Complete both procedures in this section to configure and create the initial Oracle database in a Sun Cluster configuration. If you are creating and configuring additional databases, perform only the procedure, "How to Create an Oracle Database" on page 29.

▼ How to Configure Oracle Database Access

1. **Configure the disk devices for use by your volume manager.**

   For more information, see the appendix for your volume manager in the *Sun Cluster 3.0 Installation Guide.*

2. **If you are using Solstice DiskSuite, set up UFS logging or raw mirrored meta devices on all the nodes that are running Sun Cluster HA for Oracle.**

   If you are using raw devices to contain the databases, change the owner, group, and mode of each of the raw mirrored meta devices and verify the changes. If you are not using raw devices, skip this step. Instructions for configuring raw devices are provided in the *Sun Cluster 3.0 Installation Guide.*

If you are creating raw devices, type the following commands for each device on *each node* that can master the Oracle resource group:

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

Verify that the changes have taken effect.

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

3. **If you are using VERITAS Volume Manager, set up UFS logs or raw devices on all the nodes.**

   For information about Solaris UFS logging, see the `mount_ufs`(1M) man page and the *Solaris Transition Guide*. If you are using raw devices to contain the databases, change the owner, group, and mode of each device. If not, skip this step.

   If you are creating raw devices, type the following command for each raw device:

```
# vxedit -g disk_device_group set user=oracle_id \
group=dba mode=600 volume_name
```

   Verify that the changes have taken effect.

```
# ls -lL /dev/vx/rdsk/disk_device_group/volume_name
```

   Next, re-register the disk device group with the cluster. This step is necessary to keep the VxVM namespace consistent throughout the cluster. If you are using Solstice DiskSuite, you can skip this step.

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

▼ 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 the cluster file system. For information on installation locations, refer to “Preparing to Install Sun Cluster HA for Oracle” on page 25.

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 are using 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 create the database by using the Oracle `svrmgrl` command.

   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 30.

Setting Up Oracle Database Permissions

Use this procedure to set up Oracle database permissions.

▼ How to Set Up Oracle Database Permissions

Depending on which authentication method you choose, Oracle authentication or Solaris authentication, perform either Step 1 on page 31 or Step 2 on page 31 of this procedure.

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

   You must complete this step if you do not enable Solaris authentication, as described in Step 2 on page 31.

   For all supported Oracle releases, enable access by typing the following script into the screen displayed by the `srvmgrl`(1M) command.

```
# 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;
```

2. **Grant permission for the database to use Solaris authentication.**

   Perform this step if you choose not to use Step 1 on page 31.

Note - The user for which you enable Solaris authentication is the user who owns the files under `$ORACLE_HOME`. 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;
```

(continued)

(Continuation)

```
```

3. Configure NET8 for Sun Cluster.

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

Sun Cluster HA for Oracle 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 = logicalhostname) <- use logical host name
    (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 host name
        (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:

```
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:

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

(continued)

(Continuation)

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

4. Verify that Sun Cluster is installed and running on all nodes.

```
# scstat
```

Where to Go from Here

To register and configure the Sun Cluster HA for Oracle data service, go to “Installing Sun Cluster HA for Oracle Packages” on page 34.

Installing Sun Cluster HA for Oracle Packages

The `scinstall`(1M) utility installs `SUNWscor`, the Sun Cluster HA for Oracle data service package, on a cluster. You can install specific data service packages from the Sun Cluster data service CD by using interactive `scinstall`, or you can install all data service packages on the CD by using the `–s` option to non-interactive `scinstall`. The preferred method is to use interactive `scinstall`, as described in the following procedure.

The data service packages might have been installed as part of your initial Sun Cluster installation. If not, use this procedure to install them now.

▼ How to Install Sun Cluster HA for Oracle Packages

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

1. **Load the data service CD into the CD-ROM drive.**

2. **Run `scinstall` with no options.**
   This step starts `scinstall` in interactive mode.

3. **Select the menu option: "Add support for new data service to this cluster node."**
   You can then load software for any data services that exist on the CD.

4. **Exit `scinstall` and unload the CD from the drive.**

Where to Go from Here

See "Registering and Configuring Sun Cluster HA for Oracle" on page 35 to register Sun Cluster HA for Oracle and configure the cluster for the data service.

Registering and Configuring Sun Cluster HA for Oracle

Sun Cluster HA for Oracle is registered and configured as a failover data service. You must register the data service and configure resource groups and resources for the Oracle server and listener. For details on resources and resource groups, see Chapter 1 and the *Sun Cluster 3.0 Concepts* document.

▼ How to Register and Configure Sun Cluster HA for Oracle

To register and configure the Sun Cluster HA for Oracle data service, use the Cluster Module of Sun Management Center or the following command-line procedure.

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

- The names of the cluster nodes that master the data service.

- The logical host name to be used by clients to access the data service. Normally, you set up this IP address up when you install the cluster. For details, see the section on setting up logical host names in the *Sun Cluster 3.0 Installation Guide*.
- The path to the Oracle application binaries for the resources that you plan to configure.

Perform this procedure on any cluster member.

1. Become superuser on a node in the cluster.

2. Register the resource types for the data service.

For Sun Cluster HA for Oracle, 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-name [–h nodelist]
```

–g ***resource-group-name***	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 -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 logical host names that you are using have been added to your name service database.**

   You should have done this verification as part of the Sun Cluster installation.

Note - To avoid any failures because of name service lookup, ensure that all logical host names are present in the server's and client's `/etc/hosts` file.

---

5. **Add a logical host name to the failover resource group.**

```
# scrgadm –a –L –g resource-group-name –l logical-hostname \
[–j resource-name] [–n network-interface-id-list]
```

–l ***logical-hostname***	Specifies a logical hostname.
–j ***resource-name***	An optional name for the logical host name resource. If a name is not specified, the default resource name is the first name to appear after the –l option.
–n ***network-interface-id-list***	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 *network-interface-list*. If you do not specify this option, `scrgadm` attempts to discover a net adapter on the subnet identified by the *hostname* list for each node in *nodelist*.

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

```
# scrgadm -a -j resource-name -g resource-group-name \

-t SUNW.oracle_server \
–x Connect_string=user/passwd \
–x ORACLE_SID=instance-name \
–x ORACLE_HOME=Oracle-home \
–x Alert_log_file=path-to-log
```

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

–j ***resource-name***	Specifies the name of the resource to add.
–g ***resource-group-name***	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 used by the fault monitor to connect to the database. These settings must agree with the permissions you set up in "How to Set Up Oracle Database Permissions" on page 31. If you are using Solaris authorization, type a slash (/) instead of the user name and password.
–x ORACLE_SID=***instance-name***	Sets the Oracle system identifier.
–x LISTENER_NAME=***listener-name***	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.

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 41 for a list of extension properties.

---

7. Enable the resource and fault monitoring, then move the resource group into a managed state and bring it online.

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

–Z	Enables the resource and monitor, moves the resource group to the managed state, and brings it online.
–g ***resource-group-name***	Specifies the name of the resource group.

Example—Registering Sun Cluster HA for Oracle

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

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

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

(Add the logical hostname resource to the resource group.)
# scrgadm -a -L -g oracle-rg -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 -g oracle-rg \
-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 -g oracle-rg \
-t SUNW.oracle_listener -x ORACLE_HOME=/global/oracle \
-x LISTENER_NAME=ora-lsnr

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

▼ How to Configure SUNW.HAStorage Resource Type

The SUNW.HAStorage resource type synchronizes actions between HA storage and data service. Because Sun Cluster HA for Oracle is disk-intensive, we strongly recommend that you set up SUNW.HAStorage.

For details on the background, see the SUNW.HAStorage(5) man page and “Relationship Between Resource Groups and Disk Device Groups” on page 18. For the procedure, see “How to Set Up SUNW.HAStorage Resource Type for New Resources” on page 168.

Verifying the Sun Cluster HA for Oracle Installation

Perform the following verification tests to ensure the Sun Cluster HA for Oracle was installed correctly.

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

▼ How to Verify the Sun Cluster HA for Oracle Installation

1. **Log in to the node monitoring the resource group that contains the Oracle resources and set the Oracle environment variables.**

   Log in as *oracle_id* to the node that currently masters the Oracle resource group and set the environment variables ORACLE_SID and ORACLE_HOME.

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

3. **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
   ```

4. **Shut down the Oracle instance.**

   Because the Oracle instance is under Sun Cluster control, Sun Cluster restarts it afterward.

5. **Switch the resource group that contains the Oracle database resource to another node in the cluster.**

   For example:

```
# scswitch -z -g rg-name -h phys-nodename
```

6. **Log in to the node now that contains the resource group and repeat the checks listed in Step 1.**

   Log in as `oracle_id` to the new master node and confirm interactions with the Oracle instance.

Oracle Clients

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

For example, in the `tnsnames.ora` file, you must specify the logical host name as the host on which the database instance is running. See "How to Set Up Oracle Database Permissions" on page 31.

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 how to configure Sun Cluster HA for Oracle extension properties.

For details on all Sun Cluster properties, see Appendix A.

▼ How to Configure Sun Cluster HA for Oracle Extension Properties

Typically, you configure extension properties by using the Cluster Module of Sun Management Center or the command line `scrgadm -x` *parameter=value* at the time you create the Oracle server and listener resources. You can also configure them later by using the procedures described in Chapter 9.

Some extension properties can be updated dynamically and others only when the resource is created or when it is disabled. The Tunable column in the following two tables indicates when the property can be updated.

The required extension property for creating an Oracle listener resource is `ORACLE_HOME`. Table 2–3 describes the extension properties that you can set for the Oracle listener resource.

TABLE 2–3 Sun Cluster HA for Oracle Listener Extension Properties

Name/Data Type	Default	Range	Tunable	Description
`LISTENER_NA`EISTENER (string)		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, which is to be set before listener startup and shutdown

Table 2–4 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 `ORACLE_HOME`, `ORACLE_SID`, `Alert_log_file`, and `Connect_string`.

TABLE 2–4 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 are used by the fault monitor to connect to the database

TABLE 2–4 Sun Cluster HA for Oracle Server Extension Properties *(continued)*

Name/Data Type	Default	Range	Tunable	Description
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 it is not specified, this property defaults to Oracle's default.
Probe_timeout (integer)	60	0– 99,999	Any time	The time-out value (in seconds) used by the fault monitor 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

CHAPTER 3

Installing and Configuring Sun Cluster HA for iPlanet Web Server

This chapter provides the procedures for installing and configuring Sun Cluster HA for iPlanet Web Server. 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 they refer to iPlanet Web Server.

This chapter contains the following procedures:

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

Note - If you are running multiple data services in your Sun Cluster configuration, you can set up the data services in any order, with one exception: If Sun Cluster HA for iPlanet Web Server depends on Sun Cluster HA for DNS, you must set up DNS first. 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, 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 it is started, the iPlanet Web Server is controlled by Sun Cluster.

Planning the Installation and Configuration

Use the following section in conjunction with the worksheets in the *Sun Cluster 3.0 Release Notes* as a checklist before installing and configuring Sun Cluster HA for iPlanet Web Server.

Consider the following prior to starting your installation:

- Will you be running Sun Cluster HA for iPlanet Web Server as a failover or as a scalable service? For information on the two types of services, see the *Sun Cluster 3.0 Concepts* document. For scalable services, consider the following:
  - What nodes will host the scalable service? In most cases, you will want to scale across all nodes; however, you can limit the set of nodes that host the service.
  - Will your iPlanet Web Server instances require sticky IP? This 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, hence a different node might service a subsequent client request even if the client had been previously serviced by another node in the cluster.

    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. For a description of the resource property `Resource_dependencies` that sets these dependencies, see Appendix A.

- Determine the resource groups to use for network addresses and application resources and the dependencies between them. For a description of the resource group property `RG_dependencies` that sets these dependencies, see Appendix A.
- Provide the logical host name (for failover services) or shared address (for scalable services) to be used by clients to access the data service.
- Since you can configure iPlanet Web Server to bind to `INADDR_ANY`, if you plan on running 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; however, the number of entries must be the same and must map to each other in the order specified. For details, see "How to Register and Configure Sun Cluster HA for iPlanet Web Server" on page 54.
- 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

Table 3–1 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 48
Install the Sun Cluster HA for iPlanet Web Server data service packages	"Installing Sun Cluster HA for iPlanet Web Server Packages" on page 53

TABLE 3–1 Task Map: Installing and Configuring Sun Cluster HA for iPlanet Web Server *(continued)*

Task	For Instructions, Go To ...
Configure the Sun Cluster HA for iPlanet Web Server data service	"Registering and Configuring Sun Cluster HA for iPlanet Web Server" on page 54
Configure resource extension properties	"Configuring Sun Cluster HA for iPlanet Web Server Extension Properties" on page 63

Installing and Configuring an iPlanet Web Server

This section describes the steps for installing the iPlanet Web Server (by using the `setup` command) and enabling it 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`.

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

▼ 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, refer to "Determining the Location of the Application Binaries" on page 17.

- The logical host name (for failover services) or shared address (for scalable services) to be used by clients to access the data service. You must configure these addresses and they must be online.

Note - If you are running 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 node in the cluster.**

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

3. **When prompted, type 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 `setup` on all the cluster nodes that are potential primaries of the network resource (logical host name or shared address) specified in the next step.

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

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

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

   ---

5. **Select "Run admin server as root" when asked.**

   Note the port number selected by the iPlanet install script 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 configuring the iPlanet server instance.

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

   Follow the guidelines for your system.

   When a message indicating that the admin server will be started is displayed, 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 50.

▼ How to Configure an iPlanet Web Server

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

Note the following before performing this procedure:

- Before starting, 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 the cluster file system.
- After the service has started, if you are running secure instances, you must install certificates installed from each cluster node. 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`.

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

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. **Type the URL of the iPlanet admin server in the Netscape browser.**

   The URL consists of the physical host name and port number that was established by the iPlanet installation script in Step 4 on page 49 of the server installation procedure, for example, `n1.eng.sun.com:8888`. When you perform Step 2 on page 50 above, the admin URL is displayed by the `./start` command.

   When prompted, log in to the iPlanet administration server interface by using the user ID and password you specified in Step 6 on page 49 of the server installation procedure.

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 `root`.
   - Change the bind address field to:
     - A logical host name or shared address if you are using DNS as your name service
     - The IP address associated with the logical host name or shared address if you are using 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 managed by iPlanet Web Server.**

   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.

   For example:

```
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. You must edit this file to change the location to that of the directory you created in Step 5 on page 51. 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 magnus.conf 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, it 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 this 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 log file to be in the directory you created in Step 5 on page 51.

For example:

```
Log File: /var/pathname/http_instance/logs/access
```

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

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

Where to Go from Here

If the data service packages for Sun Cluster HA for iPlanet Web Server have not been installed from the Sun Cluster data service CD, go to "Installing Sun Cluster HA for iPlanet Web Server Packages" on page 53. Otherwise, go to "Registering and Configuring Sun Cluster HA for iPlanet Web Server" on page 54.

Installing Sun Cluster HA for iPlanet Web Server Packages

The `scinstall`(1M) utility installs `SUNWschtt`, the Sun Cluster HA for iPlanet Web Server data service package, on a cluster. You can install specific data service packages from the Sun Cluster data service CD by using interactive `scinstall`, or you can install all data service packages on the CD by using the `–s` option to non-interactive `scinstall`. The preferred method is to use interactive `scinstall`, as described in the following procedure.

The data service packages might have been installed as part of your initial Sun Cluster installation. If not, use this procedure to install them now.

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

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

1. **Load the data service CD into the CD-ROM drive.**

2. **Run `scinstall` with no options.**

   This command starts `scinstall` in interactive mode.

3. **Select the menu option: "Add support for new data service to this cluster node."**

   You can then load software for any data services that exist on the CD.

4. **Exit `scinstall` and unload the CD from the drive.**

Where to Go from Here

See “Registering and Configuring Sun Cluster HA for iPlanet Web Server” on page 54 to register Sun Cluster HA for iPlanet Web Server and configure the cluster for the data service.

Registering and Configuring Sun Cluster HA for iPlanet Web Server

You can configure Sun Cluster HA for iPlanet Web Server 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

To register and configure the Sun Cluster HA for iPlanet Web Server data service, use the Cluster Module of Sun Management Center or the following command-line procedure.

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

- The name of the resource type for Sun Cluster HA for iPlanet Web Server. 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 host name (for failover services) or shared address (for scalable services) to be used by clients 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. For a discussion of the advantages and disadvantages of each location, see “Determining the Location of the Application Binaries” on page 17.

Note - The `Network_resources_used` setting on the iPlanet application resource determines the set of IP addresses that are used by iPlanet Web Server. The `Port_list` setting on the resource determines the list of port numbers in use by iPlanet Web Server. 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 it into separate instances that do.

Perform this procedure on any cluster member.

1. **Become superuser on a node in the cluster.**

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

```
# 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 fo-resource-group-name [–h nodelist]
```

–g ***fo-resource-group-name***	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 are using have been added to your name service database.**

   You should have done this verification as part of the Sun Cluster installation. For details, see the planning chapter in the *Sun Cluster 3.0 Installation Guide.*

Note - To avoid any failures because of name service lookup, ensure that all logical host names 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 host name or shared address) to the failover resource group.**

```
# scrgadm –a {–S | –L} –g fo-resource-group-name \
–l network-resource,… [–j resource-name] \
[–X auxnodelist=nodeid, …] [–n network-interface-id-list]
```

–S | –L	You use –S for shared address resources or –L for logical host name resources.
–g ***fo-resource-group-name***	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-name***	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=***nodeid, …***	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 in the case of failover. These nodes are mutually exclusive with the nodes identified in *nodelist* for the resource group, if specified.

–n ***network-interface-id-list*** — 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 *network-interface-list*. If you do not specify this option, `scrgadm` attempts to discover a net adapter on the subnet identified by the *hostname* list for each node in *nodelist*.

6. Scalable services only: Create a scalable resource group to run on all desired nodes of the cluster.

If you are running Sun Cluster HA for iPlanet Web Server as a failover data service, skip Step 7 on page 58.

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 you created in Step 3 on page 55. This dependency ensures that in the event of failover, the resource manager starts up the network resource before any data services that depend on it.

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

–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-name*** Identifies the resource group that contains the shared address resource on which the resource group being created depends.

7. Scalable services only: Create an application resource in the scalable resource group.

If you are running Sun Cluster HA for iPlanet Web Server as a failover data service, skip to Step 8 on page 59. 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`. For a description of these properties, see Appendix A. See also the examples that follow this section.

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

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

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

–t ***resource-type-name*** 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 used by the data service.

–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. This is a required extension property for Sun Cluster HA for iPlanet Web Server.

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. Failover services only: Create an application resource in the failover resource group.

Perform this step only if you are running Sun Cluster HA for iPlanet Web Server as a failover data service. If you are running Sun Cluster HA for iPlanet Web Server as a scalable service, you must have performed Step 6 on page 57 and Step 7 on page 58 previously and must now go to Step 10 on page 60. 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-name -g fo-resource-group-name \
-t resource-type-name -y Network_resources_used=logical-hostname-list \
–y Port_list=port-number/protocol \
–x Confdir_list=config-directory
```

–j ***resource-name***	Specifies the name of the resource to add.
–g ***fo-resource-group-name***	Specifies the name of the failover resource group into which the resources are to be placed.
–t ***resource-type-name***	Specifies the type of the resource to add.
–y `Network_resources_used=`***network-resources***	Specifies a comma-separated list of network resources that identify the logical hosts used by the data service.
–y `Port_list=`***port-number/protocol***	Specifies the port number and protocol to be used, 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. `Confdir_list` for failover services must have exactly one entry only. *config-directory* must contain a directory called `config`. This is a required extension property.

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

---

9. Bring the failover resource group online.

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

–Z

Enables the network resource and fault monitoring, switches the resource group into a managed state, and brings it online.

–g ***fo-resource-group-name***

Specifies the name of the failover resource group.

10. Scalable services only: Bring the scalable resource group online.

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

–Z

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

–g ***ss-resource-group-name***

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
```

(continued)

(Continuation)

```
Resource groups: sa-schost-1 (for shared addresses),
 iws-schost-1 (for scalable iPlanet application resources)
Resources: schost-1 (shared address),
 iplanet-insecure (insecure iPlanet application resource).
    iplanet-secure (secure iPlanet application resource)

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

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

(Add a scalable resource group.)
# scrgadm -a -g iws-schost-1 -y Maximum_primaries=2 \
-y Desired_primaries=2 –y RG_dependencies=sa-schost-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 -g iws-schost-1 \
-t SUNW.iws \
-x Confdir_List=/opt/iplanet/https-iplanet-insecure \
-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 -g iws-schost-1 \
-t SUNW.iws \
-x Confdir_List=/opt/iplanet/https-iplanet-secure \
-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-schost-1

(Bring the scalable resource group online.)
# scswitch -Z -g iws-schost-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: lh-schost-1 (for all resources),
Resources: schost-1 (logical hostname),
 iplanet-insecure (insecure iPlanet application resource),
  iplanet-secure (secure iPlanet application resource)

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

(Add the logical hostname resource to the resource group.)
# scrgadm -a -L -g lh-schost-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 -g lh-schost-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 -g lh-schost-1 \
-t SUNW.iws \
-x Confdir_List=/opt/iplanet/https-iplanet-secure \
-y Scalable=False -y Network_resources_used=schost-1 \
-y Port_list=443/tcp

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

Where to Go from Here

To set or modify resource extension properties, see "Configuring Sun Cluster HA for iPlanet Web Server Extension Properties" on page 63.

▼ How to Configure SUNW.HAStorage Resource Type

The SUNW.HAStorage resource type synchronizes actions between HA storage and data service. Because Sun Cluster HA for iPlanet Web Server is scalable, we strongly recommend that you set up SUNW.HAStorage.

For details on the background, see the SUNW.HAStorage(5) man page and "Relationship Between Resource Groups and Disk Device Groups" on page 18. For the procedure, see "How to Set Up SUNW.HAStorage Resource Type for New Resources" on page 168.

Configuring 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.

For details on all Sun Cluster properties, see Appendix A.

▼ How to Configure Sun Cluster HA for iPlanet Web Server Extension Properties

Typically, you configure extension properties by using the Cluster Module of Sun Management Center or the command line `scrgadm -x` *parameter=value* at the time you create the iPlanet Web Server resource. You can also configure them later by using the procedures described in Chapter 9.

Some extension properties can be updated dynamically and others only when the resource is created. The only required extension property for creating an iPlanet server resource is `Confdir_list`. Table 3–2 describes extension properties you can configure for the iPlanet server. The Tunable column indicates when the property can be updated.

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

Name/Data Type	Default Range	Tunable	Description
`Confdir_list` (string array)	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)	0 – 2,147,483,641 –1 indicates an infinite number of retry attempts.	Any time	The number of times the fault monitor is to be restarted by the process monitor facility during the time window specified by the `Monitor_retry_interval` property. Note that this property refers to restarts of the fault monitor itself rather than to the resource. Restarts of the resource are controlled by the system-defined properties `Retry_interval` and `Retry_count`.
`Monitor_retry_interval` (integer)	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 fault monitor is not restarted by the process monitor facility.
`Probe_timeout` (integer)	30 – 2,147,483,641	Any time	The time-out value (in seconds) used by the fault monitor to probe an iPlanet Web Server instance.

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.

This chapter contains the following procedures:

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

Planning the Installation and Configuration

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

Consider the following 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 they can be viewed or updated from any cluster node.

- If you plan on using multiple NDS instances on a node, you must set the `listenhost` directive in `slapd.conf` with the appropriate network resource as the IP address (a logical host name). 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 host name `nds-1`, put the following into its `slapd.conf file`: `listenhost nds-1`. That way, the instance binds to the logical host name `nds-1` only rather than to all the IP addresses on the node.

Installing and Configuring Sun Cluster HA for Netscape Directory Server

Table 4–1 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 69

TABLE 4–1 Task Map: Installing and Configuring Sun Cluster HA for Netscape Directory Server *(continued)*

Task	For Instructions, Go To ...
Install the Sun Cluster HA for Netscape Directory Server data service packages	"Installing Sun Cluster HA for Netscape Directory Server Packages" on page 71
Configure application resources and start Sun Cluster HA for Netscape Directory Server	"Completing the Sun Cluster HA for Netscape Directory Server Configuration" on page 72
Configure resource extension properties	"Configuring Sun Cluster HA for Netscape Directory Server Extension Properties" on page 75

Note - If you are running multiple data services in your Sun Cluster configuration, you can set up the data services in any order, with one exception: If you use Sun Cluster HA for DNS, you must set it up before setting 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, do not manually start and stop Netscape Directory Server except by using the cluster administration command `scswitch`(1M). Refer to the man page for details. After Netscape Directory Server is started, it is controlled by Sun Cluster.

Configuring and Activating Network Resources

Before you install and configure Netscape Directory Server, set up the network resources the server attempts to use after it has been installed and configured. To configure and activate the network resources, use the Cluster Module of Sun Management Center or 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 host name to be used by clients to access Sun Cluster HA for Netscape Directory Server. Normally, you set up this host name when you install the cluster. For details on setting up logical host names, see the section in the *Sun Cluster 3.0 Installation Guide* on setting up logical host names.

Perform this procedure on any cluster member.

1. **Become superuser on a node in the cluster.**

2. **Verify that all network addresses that you are using have been added to your name service database.**

   You should have done this verification as part of the Sun Cluster installation. For details, see the planning chapter in the *Sun Cluster 3.0 Installation Guide.*

   Note - To avoid any failures because of name service lookup, ensure that all logical host names and shared addresses are present in the `/etc/hosts` file on all cluster nodes. Configure name service mapping in `/etc/nsswitch.conf` 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-name [–h nodelist]
   ```

   –g ***resource-group-name*** 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 `-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. **Add logical host name resources to the resource group.**

```
# scrgadm -a -L -g resource-group-name -l hostname, …
```

–L	Specifies a logical host name resource is being added.
–g ***resource-group-name***	Specifies the name of the resource group.
–l ***hostname, …***	Specifies a comma-separated list of logical host names.

5. Verify that all logical host names that you are using have been added to your name service database.

You should have done this verification as part of the Sun Cluster installation. For details, see the planning chapter in the *Sun Cluster 3.0 Installation Guide*.

6. Enable the resource group and bring it online.

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

–Z	Moves the resource group to the managed state and brings it online.
–g ***resource-group-name***	Specifies the name of the resource group.

Where to Go from Here

After the network resources have been configured and activated, proceed to install and configure Netscape Directory Server by using the procedure in the next section, “Installing and Configuring Netscape Directory Server” on page 69.

Installing and Configuring Netscape Directory Server

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

Netscape Directory Server requires some variation from the default installation parameters, notably:

- 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 host name (IP address) that can fail over between nodes. This requirement means that before you begin the installation, you must set up the logical host name in your name services. This step is normally done as part of the Sun Cluster installation and is described in the *Sun Cluster 3.0 Installation Guide*.

- 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 Sun Cluster HA for Netscape Directory Server are included here. For the other sections, choose or change the default values as appropriate. These are the basic steps only; for details, see the Netscape LDAP documentation.

1. **Become superuser on a node in the cluster.**

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 by using a Custom Installation.**

   Supply the logical host name 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 host name 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 are running the Netscape Directory Server installation. This state is necessary because at the end of the Netscape Directory Server installation, it automatically starts up Netscape Directory Server and fails if the logical host is offline on that node.

   ---

5. **Select the logical host name 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 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

1. **Use the Netscape admin server to configure and test Netscape Directory Server.**

   See your Netscape documentation for details.

   Upon completion of the configuration, Netscape Directory Server starts automatically. Before proceeding to the next part of the installation and configuration process, you must stop the server by using `stop-slapd`.

Where to Go from Here

If the data service packages for Netscape Directory Server have not been installed from the Sun Cluster data service CD, go to "Installing Sun Cluster HA for Netscape Directory Server Packages" on page 71. If the packages have been installed, go to "Completing the Sun Cluster HA for Netscape Directory Server Configuration" on page 72.

Installing Sun Cluster HA for Netscape Directory Server Packages

The `scinstall`(1M) utility installs `SUNWscnsl`, the Sun Cluster HA for Netscape Directory Server data service package, on a cluster. You can install specific data service packages from the Sun Cluster data service CD by using interactive `scinstall`, or you can install all data service packages on the CD by using the `–s` option to non-interactive `scinstall`. The preferred method is to use interactive `scinstall`, as described in the following procedure.

The data service packages might have been installed as part of your initial Sun Cluster installation. If not, use this procedure to install them now.

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

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

1. **Load the data service CD into the CD-ROM drive.**

2. **Run `scinstall` with no options.**
   This command starts `scinstall` in interactive mode.

3. **Select the menu option: "Add support for new data service to this cluster node."**
   You can then load software for any data services that exist on the CD.

4. **Exit `scinstall` and unload the CD from the drive.**

Where to Go from Here

See "Completing the Sun Cluster HA for Netscape Directory Server Configuration" on page 72 to register Sun Cluster HA for Netscape Directory Server and configure the cluster for the data service.

Completing the Sun Cluster HA for Netscape Directory Server Configuration

To complete the Sun Cluster HA for Netscape Directory Server configuration, use the Cluster Module of Sun Management Center or the following command-line procedure. The example that follows the procedure shows the complete set of steps for installing and configuring Sun Cluster HA for Netscape Directory Server.

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

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

- The name of the resource type for Sun Cluster HA for Netscape Directory Server. This name is `SUNW.nsldap`.
- The names of the cluster nodes that can master the data service.

- The logical host name to be used by clients to access Sun Cluster HA for Netscape Directory Server. Normally, you set up this logical host name when you install the cluster. For details, see the section on setting up logical host names in the *Sun Cluster 3.0 Installation Guide.*
- The path to the Netscape Directory Server application binaries that are the resources for Sun Cluster HA for Netscape Directory Server. You can install the binaries on the local disks or the cluster file system. For a discussion of the advantages and disadvantages of each location, see Chapter 1.
- 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`; 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 `Port_list`. For instructions on setting resource properties, see Chapter 9.

Run this procedure on any cluster member.

1. Become superuser on a node in the cluster.

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 in the failover resource group created previously.

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

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

–j ***resource-name*** Specifies the LDAP application resource name.

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

–t ***resource-type-name***	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`. `Port_list` must have exactly one entry.
–x `Confdir_list=`***path***	Specifies a path for your LDAP configuration directory. The `Confdir_list` extension property is required. `Confdir_list` must have exactly one entry.

4. Enable the resource and its monitor.

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

–e	Enables the resource and its monitor.
–g ***resource-name***	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 Sun Cluster HA for Netscape Directory Server.

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

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

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

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

(Install and configure Netscape Directory Server.)

(Stop the LDAP server.)

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

(continued)

```
(Create an LDAP resource and add it to the resource group.)
# scrgadm -a -j nsldap -g lh-schost-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
```

▼ How to Configure SUNW.HAStorage Resource Type

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

For details on the background, see the SUNW.HAStorage(5) man page and “Relationship Between Resource Groups and Disk Device Groups” on page 18. For the procedure, see “How to Set Up SUNW.HAStorage Resource Type for New Resources” on page 168.

Configuring Sun Cluster HA for Netscape Directory Server Extension Properties

Table 4–2 describes the extension properties you can configure for Netscape Directory Server. The only required extension property for creating an Netscape Directory Server resource is Confdir_list, which specifies a directory in which the Netscape Directory Server configuration files reside.

▼ How to Configure Sun Cluster HA for Netscape Directory Server Extension Properties

Typically, you configure the extension properties by using the Cluster Module of Sun Management Center or the command line scrgadm -x *parameter=value* at the time

you create the Netscape Directory Server resource. You can also configure them later by using the procedures described in Chapter 9.

See Appendix A for details on all Sun Cluster properties.

Table 4–2 describes the Sun Cluster HA for Netscape Directory Server extension properties. Some extension properties can be updated dynamically and others only when the resource is created. The Tunable column indicates when the property can be updated.

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

Name/Data Type	Range/Default	Tunable	Description
`Confdir_list` (string array)	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. This is a required extension property and 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)	0 – 2,147,483,641 –1 indicates an infinite number of retry attempts.	Any time	The number of times the fault monitor is to be restarted by the process monitor facility during the time window specified by the `Monitor_retry_interval` property. Note that this property refers to restarts of the fault monitor itself rather than to the resource. Restarts of the resource are controlled by the system-defined properties `Retry_interval` and `Retry_count`.

TABLE 4–2 Sun Cluster HA for Netscape Directory Server Extension Properties *(continued)*

Name/Data Type	Default Range	Tunable	Description
`Monitor_retry_interval` (integer)	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 fault monitor cannot be restarted by the process monitor facility.
`Probe_timeout` (integer)	30 – 2,147,483,641	Any time	The time-out value (in seconds) used by the fault monitor to probe an Netscape Directory Server instance.

CHAPTER 5

Installing and Configuring Sun Cluster HA for Apache

This chapter describes the steps for installing and configuring Sun Cluster HA for Apache on your Sun Cluster servers.

This chapter contains the following procedures:

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

Planning the Installation and Configuration

Prior to installation of Sun Cluster HA for Apache, update the following information in the Apache configuration file, `httpd.conf`:

- The `ServerName` directive that contains the host name. For Sun Cluster HA for Apache to be highly available, you must set this directive to the name of the network address (logical host name or shared address) that is used to access the server. The logical host name or shared address should have been set up when the cluster was installed; if not, refer to the *Sun Cluster 3.0 Installation Guide* for information on how to set up logical host names and shared addresses and set one up now.
- The `BindAddress` directive, which you must set to the logical host or shared address. Since you can configure Apache to bind to `INADDR_ANY`, if you plan on running 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, which 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 it 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 17.

  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. For example:

```
(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
```

  To start up the instances by hand, as you might if you are verifying your setup, use the following commands. Also, when instructed by the Resource Group Manager (RGM), the data service in effect issues the following commands to start up 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 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 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, preserve availability by locating the CGI directory on the cluster file system. For example, you might map your CGI directory to `/global/`*disk-device-group*`/`*ServerRoot*`/cgi-bin`, where *disk-device-group* 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 "back-end" server is also controlled by Sun Cluster. If the server is an RDBMS supported by Sun Cluster, use one of the highly available RDBMS packages. Alternatively, you can put the server under Sun Cluster control by using the APIs documented in the *Sun Cluster 3.0 Data Services Developers' Guide.*

- If you are using a lock file, set the value of the `LockFile` directive in your `httpd.conf` file to a local file.
- Use a `PidFile` directive to point to a local file. For example:

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

- The `Port` directive setting accessed by the server port or ports. 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.

  `Port_list` assumes that the Web server serves all combinations of ports and IP addresses from the network resources as defined in `Network_resources_used`. For example:

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

  probes the following IP-port combinations:

Host	Port	Protocol
host-1	80	tcp
host-1	443	tcp
host-2	8080	tcp
host-2	80	tcp
host-2	443	tcp
host-2	8080	tcp

However, if *host-1* serves 80 and 443 only and *host-2* serves ports 80 and 8080 only, you can configure `Port_list` for Apache as follows:

```
Port_list=host-1/80/tcp,host-1/443/tcp,host-2/80/tcp,host-2/8080/tcp
```

Bear in mind the following rules:

- You must specify host names or IP addresses (not network resource names) for *host-1* and *host-2*.
- If Apache serves *host-n*`/`*port* for every *host-n* in `Network_resources_used`, you can use a short form to replace the combination of *host-1*`/`*port-1*, *host-2*`/`*port-2*, and so on. See the following examples.

Example One:

```
Port_list=''80/tcp,host-1/443/tcp,host-2/8080/tcp''
Network_resources_used=host-1,host-2
```

probes the following IP-port combinations:

Host	Port	Protocol
host-1	80	tcp
host-1	443	tcp

host-2	80	`tcp`
host-2	8080	`tcp`

Example Two:

```
Port_list=''host-1/80/tcp,host-2/80/tcp''
Network_resources_used=net-1,net-2
#net-1 contains host-1.
#net-2 contains host-2 and host-3.
```

probes the following IP-port combinations:

Host	**Port**	**Protocol**
host-1	80	`tcp`
host-2	80	`tcp`

- All host names (IP addresses) that are specified in `Port_list` 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 an IP address is already in use by another scalable resource, 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 Sun Cluster HA for Apache, you must consider or provide information on the following:

- Decide whether to run Sun Cluster HA for Apache as a failover or a scalable service.
- Decide which fault monitoring resource properties (such as `Thorough_probe_interval` or `Probe_timeout`) to set. In most cases, the default values suffice. For information on these properties, refer to "Configuring Sun Cluster HA for Apache Extension Properties" on page 96.
- Provide the name of the resource type for Sun Cluster HA for Apache. This name is `SUNW.apache`.
- Provide the names of the cluster nodes that will master the data service.

- Provide the logical host name (failover services) or shared address (scalable services) to be used by clients to access the data service. This IP address is normally set up when the cluster is installed. For details on how to set up network addresses, see the *Sun Cluster 3.0 Installation Guide.*
- Provide 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 17.
- Provide the path to the `conf` directory.
- 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, hence a different node might service a subsequent client request even if the client had been previously serviced by another node in the cluster.

  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, Confdir_list can have only one entry; Port_list can have multiple entries. For scalable services, both properties can have multiple entries. For details, see "How to Register and Configure Sun Cluster HA for Apache" on page 88.

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 Apache	"How to Install and Configure the Apache Application Software" on page 85
Install the Sun Cluster HA for Apache data service packages	"How to Install Sun Cluster HA for Apache Packages" on page 87
Configure and start the Sun Cluster HA for Apache data service	"How to Register and Configure Sun Cluster HA for Apache" on page 88
Configure resource extension properties	"How to Configure Sun Cluster HA for Apache Extension Properties" on page 96

Installing and Configuring Apache

This section describes the steps for installing the Apache server and enable it to run as the Sun Cluster HA for Apache data service.

Sun Cluster HA for Apache works with Apache configured as either a Web server or a proxy server.

For standard installation instructions, refer to Apache documentation at `http://www.apache.org`. For a list of Apache releases supported for use with Sun Cluster, see the *Sun Cluster 3.0 Release Notes*.

▼ How to Install and Configure the Apache Application Software

1. **Become superuser on a node in the cluster.**

2. **Install Apache by using the steps described in Apache documentation.**

   Refer to the documentation you received with your Apache software or to the Apache Web site: `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 the next step for more information.
- Make changes to run as a proxy server if you choose to run Apache as a proxy server. See the Apache documentation. If you will be running Apache 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 `httpd.conf` 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); or, while configuring Sun Cluster HA for Apache, you can set the `Port_list` standard property to match the setting in `httpd.conf`.

5. **(Optional) If you will be using 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. **Verify your configuration changes.**

   Check the Apache `httpd.conf` file for correct syntax by running `apachectl configtest`.

   Ensure that any logical host names or shared addresses in use by Apache are configured and online.

   Start up your Apache server by hand by issuing `apachectl start`. If Apache does not start up correctly, correct the problem.

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

Where to Go from Here

If the data service packages for Apache have not been installed from the Sun Cluster data service CD, go to “Installing Sun Cluster HA for Apache Packages” on page 87. Otherwise, go to “Registering and Configuring Sun Cluster HA for Apache” on page 88.

Installing Sun Cluster HA for Apache Packages

The `scinstall`(1M) utility installs `SUNWscapc`, the Sun Cluster HA for Apache data service package, on a cluster. You can install specific data service packages from the Sun Cluster data service CD with interactive `scinstall`, or you can install all data service packages on the CD with the `–s` option to noninteractive `scinstall`. The preferred method is to use interactive `scinstall`, as described in the following procedure.

The data service packages might have been installed as part of your initial Sun Cluster installation. If not, use the following procedure to install them now.

▼ How to Install Sun Cluster HA for Apache Packages

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

1. **Load the data service CD into the CD-ROM drive.**

2. **Run `scinstall` with no options.**

   This step starts `scinstall` in interactive mode.

3. **Select the menu option: "Add support for new data service to this cluster node."**

   You can then load software for any data services that exist on the CD.

4. **Exit `scinstall` and unload the CD from the drive.**

Where to Go from Here

See "How to Register and Configure Sun Cluster HA for Apache" on page 88 to register Sun Cluster HA for Apache and configure the cluster for the data service.

Registering and Configuring Sun Cluster HA for Apache

To register and configure Sun Cluster HA for Apache, you can use the Cluster Module of Sun Management Center or the following command-line procedure.

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

- When Apache is configured as a failover service, you place the Apache application resources and the network resources in a single resource group.
- When Apache is configured 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. These steps are identified by the leading text “Scalable services only:” in the following procedure. If you are not configuring Apache as a scalable service, skip those steps.

▼ How to Register and Configure Sun Cluster HA for Apache

Run this procedure on any cluster member.

1. **Become superuser on a node in the cluster.**

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, it contains both network and failover application resources. For scalable services, it 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 fo-resource-group-name [–h nodelist]
```

–a	Adds a new configuration.
–g ***fo-resource-group-name***	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 are using have been added to your name service database.

This verification should have been done as part of the Sun Cluster installation. For details, see the planning chapter in the *Sun Cluster 3.0 Installation Guide.*

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 `/etc/nsswitch.conf` on the servers to first check the local files prior to accessing NIS, NIS+, or DNS.

---

5. Add a network resource (logical host name or shared address) to the failover resource group created in Step 3 on page 88.

```
# scrgadm –a {–S | –L} –g fo-resource-group-name \
–l hostname, … [–j resource-name] \
[–X auxnodelist=nodeid, …] [–n network-interface-id-list]
```

–S | –L	--S specifies shared address resources; –L specifies logical host name 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 ***fo-resource-group-name***	Specifies the name of the failover resource group created in Step 3 on page 88.
–j ***resource-name***	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 specified after the –l option.
–X auxnodelist=***nodeid, ...***	Specifies a comma-separated 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 *nodelist* for the resource group, if specified.
–n ***network-interface-id-list***	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 *network-interface-list*. If you do not specify this option, scrgadm attempts to discover a net adapter on the subnet identified by the *hostname* list for each node in *nodelist*.

6. Scalable services only: Create a scalable resource group to run on all desired nodes of the cluster.

If you are running Sun Cluster HA for Apache as a failover data service, skip to Step 8 on page 92.

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 you created in Step 3 on page 88. This dependency ensures that in the event of failover, 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 it.

```
# scrgadm –a –g ss-resource-group-name \
-y Maximum_primaries=m –y Desired_primaries=n \
```

(Continuation)

```
-y RG_dependencies=resource-group-name
```

–g ***ss-resource-group-name*** — 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-name*** — 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 on page 88.

7. Scalable services only: Create an application resource in the scalable resource group.

If you are running Sun Cluster HA for Apache as a failover data service, skip to Step 8 on page 92.

```
# scrgadm -a -j resource-name -g ss-resource-group-name \
-t resource-type-name -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-name*** — Specifies your choice for the name of the resource to add.

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

–t ***resource-type-name***	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 used by the data service.
–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. This is a required extension property.
–x Bin_dir=***bin-directory***	Specifies the location where the Apache binaries are installed. This is a required 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. Failover services only: Create an application resource in the failover resource group.

Perform this step only if you are running Sun Cluster HA for Apache as a failover data service. If you are running Sun Cluster HA for Apache as a scalable service, you should have performed Step 6 on page 90 and Step 7 on page 91 and should now go to Step 10 on page 93.

```
# scrgadm -a -j resource-name -g resource-group-name \
-t resource-type-name -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-name***	Specifies your choice for the name of the resource to add.

–g ***resource-group-name***	Specifies the name of the resource group into which the resources are to be placed, created in Step 3 on page 88.
–t ***resource-type-name***	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 used by the data service.
–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 it is set to `False` or can be omitted.
–x Confdir_list=***config-directory***	Specifies the location of the Apache configuration file. This is a required extension property and must have exactly one entry only.
–x Bin_dir=***bin-directory***	Specifies the location where the Apache binaries are installed. This is a required extension property.

9. Bring the failover resource group online.

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

–Z	Enables the shared address resource and fault monitoring, switches the resource group into a managed state, and brings it online.
–g ***fo-resource-group-name***	Specifies the name of the failover resource group.

10. Scalable services only: Bring the scalable resource group online.

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

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

–g ***ss-resource-group-name*** Specifies the name of the scalable resource group.

Example—Registering Scalable Sun Cluster HA for Apache

For scalable services, you create two resource groups: a failover resource group that contains the network resources and 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: sa-schost-1 (for shared addresses),
 ap-schost-1 (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 sa-schost-1

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

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

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

(Add Apache application resources to the scalable resource group.)
# scrgadm -a -j apache-1 -g ap-schost-1 \
-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 sa-schost-1

(Bring the scalable resource group online on both nodes.)
# scswitch -Z -g ap-schost-1
```

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: lh-schost-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 lh-schost-1

(Add the logical host name resource to the failover resource group.)
# scrgadm -a -L -g lh-schost-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 lh-schost-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 lh-schost-1
```

Where to Go from Here

Verify the installation by using the information in the section “How to Verify Data Service Installation and Configuration” on page 96. To set or modify resource extension properties, see “Configuring Sun Cluster HA for Apache Extension Properties” on page 96.

▼ How to Configure SUNW.HAStorage Resource Type

The SUNW.HAStorage resource type synchronizes actions between HA storage and data service. Because Sun Cluster HA for Apache is scalable, we strongly recommend that you set up SUNW.HAStorage.

For details on the background, see the SUNW.HAStorage(5) man page and “Relationship Between Resource Groups and Disk Device Groups” on page 18. For the procedure, see “How to Set Up SUNW.HAStorage Resource Type for New Resources” on page 168.

▼ How to Verify Data Service Installation and Configuration

After you have configured Sun Cluster HA for Apache, verify that you can open a Web page with the network resources (logical host names 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 `Confdir_list` and `Bin_dir`. `Confdir_list` specifies a directory that contains a subdirectory named `conf`, in which the Apache configuration properties (`httpd.conf`) reside.

For details on all Sun Cluster properties, see Appendix A.

▼ How to Configure Sun Cluster HA for Apache Extension Properties

Typically, you configure these properties by using the Cluster Module of Sun Management Center or the command-line `scrgadm -x` *parameter=value* at the time you create the Apache server resource. You can also configure them later by following the procedures described in Chapter 9.

Some extension properties can be updated dynamically, others only when the resource is created. Table 5–2 describes extension properties your can configure for the Apache server. The Tunable column indicates when the property can be updated.

TABLE 5–2 Sun Cluster HA for Apache Extension Properties

Name/Data Type	Default Range	Tunable	Description
`Bin_dir` (string)	None	At creation	The path to the Apache binaries. This is a required extension property.
`Confdir_list` (string array)	None	At creation	The directory that contains a subdirectory called `conf`, which contains the `httpd.conf` configuration file. This is a required 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 the fault monitor is to be restarted by the process monitor facility during the time window specified by the `Monitor_retry_interval` property. This property refers to restarts of the fault monitor itself rather than to the resource. Restarts of the resource are controlled by the system-defined properties `Retry_interval` and `Retry_count`.
`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 the fault monitor fails exceeds the value specified in the extension property `Monitor_retry_count` within this period, the fault monitor is not restarted by the process monitor facility.
`Probe_timeout` (integer)	30 0 – 2,147,483,641	At creation	The time-out value (in seconds) used by the fault monitor to probe an Apache instance.

CHAPTER 6

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

This chapter describes the steps for installing and configuring Sun Cluster HA for Domain Name Service (DNS) on your Sun Cluster servers.

This chapter contains the following procedures:

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

Installing and Configuring Sun Cluster HA for DNS

Table 6–1 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 100
Install Sun Cluster HA for DNS packages	"Installing Sun Cluster HA for DNS Packages" on page 103
Configure and start Sun Cluster HA for DNS data service	"Registering and Configuring Sun Cluster HA for DNS" on page 104
Configure resource extension properties	"Configuring Sun Cluster HA for DNS Extension Properties" on page 108

Installing DNS

This section describes the steps for installing DNS and enabling it 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. For information on setting up DNS, see the `in.named`(1M) man page. The differences in a Sun Cluster configuration are as follows:

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

▼ How to Install DNS

1. **Become superuser on a node in the cluster.**

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

   This name should be a host name that is set up when you install the Sun Cluster software. For details on setting up host names, see the *Sun Cluster 3.0 Installation Guide.*

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 `/usr/sbin` before you begin the installation.

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

   Create a `dns` directory and a `named` directory underneath it on a cluster file system, for example, `/global/dns/named`. For information on setting up cluster file systems, see the *Sun Cluster 3.0 Installation Guide.*

```
# 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` to the `/global/dns` directory. Otherwise, create a `named.conf` file in this directory. For information on the types of entries to place in `named.conf` or `named.boot`, see the `in.named`(1M) man page. 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 `named.conf`) under `/global/dns/named`.**

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

   On all the nodes, edit `/etc/resolv.conf` to contain the logical host name. 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 host name `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
```

(continued)

(Continuation)

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

Make the logical host name the first entry after the domain name. DNS attempts to access the server by using the addresses in the order they are listed in `resolv.conf`.

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

---

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

In the following example:

- Replace the *IPaddress* variable with your actual IP address, such as 129.146.87.53.
- Replace the *logicalhostname* variable with your actual logical host name.

```
127.0.0.1       localhost
IPaddress       logicalhostname
```

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.

For example:

```
hosts:    cluster files dns
```

10. Test DNS.

Be sure to stop `in.named` before proceeding. For example:

```
# 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 have already installed the Sun Cluster HA for DNS packages as part of your Sun Cluster installation, go to “Registering and Configuring Sun Cluster HA for DNS” on page 104. Otherwise, go to “Installing Sun Cluster HA for DNS Packages” on page 103.

Installing Sun Cluster HA for DNS Packages

The `scinstall`(1M) utility installs `SUNWscdns`, the Sun Cluster HA for DNS data service package, on a cluster. You can install specific data service packages from the Sun Cluster data service CD by using interactive `scinstall`, or you can install all data service packages on the CD by using the `–s` option to non-interactive `scinstall`. The preferred method is to use interactive `scinstall`, as described in the following procedure.

The data service packages might have been installed as part of your initial Sun Cluster installation. If not, use this procedure to install them now.

▼ How to Install Sun Cluster HA for DNS Packages

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

1. **Load the data service CD into the CD-ROM drive.**

2. **Run `scinstall` with no options.**
   This command starts `scinstall` in interactive mode.

3. **Select the menu option: “Add support for new data service to this cluster node.”**
   You can then load software for any data services that exist on the CD.

4. **Exit `scinstall` and unload the CD from the drive.**

Where to Go from Here

See “Registering and Configuring Sun Cluster HA for DNS” on page 104 to register Sun Cluster HA for DNS and configure the cluster for the data service.

Registering and Configuring Sun Cluster HA for DNS

To register and configure the Sun Cluster HA for DNS data service, use the Cluster Module of Sun Management Center or the following command-line procedure.

▼ 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 Sun Cluster HA for DNS. This name is `SUNW.dns`.
- The names of the cluster nodes that master the data service.
- The logical host name to be used by clients to access the data service. This IP address is normally set up when the cluster is installed. For details, see the section on setting up logical host names in the *Sun Cluster 3.0 Installation Guide*.
- 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.

Perform this procedure on any cluster member.

1. **Become superuser on a node in the cluster.**

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 to be used by logical host names and DNS resources.**

You can optionally select the set of nodes on which the data service can run by using the –h option.

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

–g ***resource-group-name*** 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 -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 logical host names that you will be using have been added to your name service database.

This verification should have been done as part of the Sun Cluster installation. For details, see the planning chapter in the *Sun Cluster 3.0 Installation Guide.*

Note - To avoid any failures because of name service lookup, verify that all logical host names 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 logical host name resources to the resource group.

```
# scrgadm –a –L –g resource-group-name \
–l logical-hostname[,logical-hostname] [–j resource-name] \
[–n network-interface-id-list]
```

–L Specifies the logical host name resources.

–l ***logical-hostname*** Specifies a comma-separated list of logical host names.

–j ***resource-name*** Specifies an optional network resource name. If you do not specify this name, it

	defaults to the first name specified after the `-l` option.
`-n` ***network-interface-id-list***	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 *network-interface-list*. If you do not specify this option, `scrgadm` attempts to discover a net adapter on the subnet identified by the *hostname* list for each node in *nodelist*.

6. Add a DNS application resource to the resource group.

```
# scrgadm -a -j [resource-name] -g resource-group-name \
-t SUNW.dns -y Network_resources_used=network-resource, …\
-y Port_list=port-number/protocol -x DNS_mode=config-file-name \
-x Confdir_list=config-directory
```

`-j` ***resource-name***	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 host names) to be used by DNS. If you do not specify this property, it defaults to all the logical host names 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, it defaults to `53/udp`.
`-x DNS_mode=`***config-file-name***	Specifies the configuration file to use, either `conf(named.conf)` or `boot(named.boot)`. If you do not specify this property, it 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. This is a required extension property for Sun Cluster HA for DNS.

7. Enable the resource and fault monitoring, move the resource group into a managed state, and bring it online.

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

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

–g ***resource-group-name*** — Specifies the name of the resource group.

Example—Registering Failover Sun Cluster HA for DNS

The following example shows how to register Sun Cluster HA for DNS 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: lh-schost-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 lh-schost-1

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

(Add DNS application resources to the resource group.)
# scrgadm -a -j dns-1 -g lh-schost-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 lh-schost-1
```

(continued)

▼ How to Configure SUNW.HAStorage Resource Type

The SUNW.HAStorage resource type synchronizes actions between HA storage and data service. Because Sun Cluster HA for DNS is not disk intensive and not scalable, setting up the SUNW.HAStorage resource type is optional.

For details on the background, see the SUNW.HAStorage(5) man page and "Relationship Between Resource Groups and Disk Device Groups" on page 18. For the procedure, see "How to Set Up SUNW.HAStorage Resource Type for New Resources" on page 168.

Verifying Data Service Installation and Configuration

To verify that Sun Cluster HA for DNS has been installed and configured correctly, run the following command after completing the procedure "How to Register and Configure Sun Cluster HA for DNS" on page 104.

```
# nslookup logical-hostname logical-hostname
```

In this example, *logical-hostname* is the name of the network resource you have configured to service DNS requests, for example, schost-1, as shown in the previous registration example. The output should indicate that the query was answered (served) by the logical host you specified.

Configuring Sun Cluster HA for DNS Extension Properties

The only required extension property for creating a DNS resource is Confdir_list.

See Appendix A for details on all Sun Cluster properties.

▼ How to Configure Sun Cluster HA for DNS Extension Properties

Typically, you configure these properties by using the Cluster Module of Sun Management Center or the command line `scrgadm -x` *parameter=value* at the time you create the Sun Cluster HA for DNS server resource. You can also configure them later by using the procedures described in Chapter 9.

Table 6–2 describes the Sun Cluster HA for DNS extension properties. Some extension properties can be updated dynamically and others only when the resource is created. The Tunable column indicates when the property can be updated.

TABLE 6–2 Sun Cluster HA for DNS Extension Properties

Name/Data Type	Default	Tunable	Description
`Confdir_list` (string array)	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`	At creation	The DNS configuration file to use, either `conf` (`named.conf`) or `boot` (`named.boot`)
`Monitor_retry_count` (integer)	4	Any time	The number of times the fault monitor is to be restarted by the process monitor facility during the time window specified by the `Monitor_retry_interval` property. This property refers to restarts of the fault monitor itself rather than to the resource. Restarts of the resource are controlled by the system-defined properties `Retry_interval` and `Retry_count`.
	Range: 0 – 2,147,483,641 –1 indicates an infinite number of retry attempts.		

TABLE 6–2 Sun Cluster HA for DNS Extension Properties *(continued)*

Name/Data Type	Default	Range	Tunable	Description
`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 fault monitor is not restarted by the process monitor facility.
`Probe_timeout` (integer)	30	0 – 2,147,483,641	Any time	The time-out value (in seconds) used by the fault monitor to probe a DNS instance

CHAPTER 7

Installing and Configuring Sun Cluster HA for Network File System (NFS)

This chapter describes the steps for installing and configuring Sun Cluster HA for Network File System (NFS) on your Sun Cluster servers and the steps for adding Sun Cluster HA for NFS to a system that is already running Sun Cluster.

This chapter contains the following procedures:

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

Use the worksheets in *Sun Cluster 3.0 Release Notes* to plan your resources and resource groups before installing and configuring Sun Cluster HA for NFS.

The mount points for NFS file systems placed under the control of Sun Cluster HA for NFS 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 host names are present in the server'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+. Doing so prevents timing-related errors in this area and ensures that `ifconfig` and `statd` succeed in resolving logical host names.

Note - To avoid "stale file handle" errors on the client during NFS failover, if you are using 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 completing the installation.

Installing and Configuring Sun Cluster HA for NFS

Table 7–1 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 113
Set up and configure Sun Cluster HA for NFS	"Setting Up and Configuring Sun Cluster HA for NFS" on page 113
Configure resource extension properties	"Configuring Sun Cluster HA for NFS Extension Properties" on page 120

Installing Sun Cluster HA for NFS Packages

The `scinstall`(1M) utility installs `SUNWscnfs`, the Sun Cluster HA for NFS data service package, on a cluster. You can install specific data service packages from the Sun Cluster data service CD by using interactive `scinstall`, or you can install all data service packages on the CD by using the `–s` option to non-interactive `scinstall`. The preferred method is to use interactive `scinstall`, as described in the following procedure.

The data service packages might have been installed as part of your initial Sun Cluster installation. If not, use this procedure to install them now.

▼ How to Install Sun Cluster HA for NFS Packages

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

1. **Load the data service CD into the CD-ROM drive.**

2. **Run `scinstall` with no options.**
   This command starts `scinstall` in interactive mode.

3. **Select the menu option: "Add support for new data service to this cluster node."**
   You can then load software for as many data services as exist on the CD.

4. **Exit `scinstall` and unload the CD from the drive.**

Where to Go from Here

See "Setting Up and Configuring Sun Cluster HA for NFS" on page 113 to register Sun Cluster HA for NFS and configure the cluster for the data service.

Setting Up and Configuring Sun Cluster HA for NFS

To register and configure the Sun Cluster HA for NFS data service, use the Cluster Module of Sun Management Center or the following command-line procedure.

Before you set up and configure Sun Cluster HA for NFS, use 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 113 for instructions on how to install the package.

▼ How to Set Up and Configure Sun Cluster HA for NFS

1. **Become superuser on a node in the cluster.**

2. **Verify that all nodes in the cluster are up and running.**

```
# scstat
```

3. **Create a failover resource group to contain the NFS resources.**

   Use the `Pathprefix` standard property to specify a directory to be used for administrative files.

```
# scrgadm -a -g resource-group -y Pathprefix=/global/admin-dir
```

Note - *admin-dir* must be a cluster file system. You must create it yourself.

–a –g ***resource-group***	Adds the named failover resource group.
–y `Pathprefix=`*path*	Specifies a directory on a cluster file system to be used by Sun Cluster HA for NFS administration files. For example, `/global/nfs`. `Pathprefix` must be unique for each resource group you create.

4. **Verify that all logical host names that you will be using have been added to your name service database.**

   You should have done this verification as part of the Sun Cluster installation. For details on this step, see the planning chapter in the *Sun Cluster 3.0 Installation Guide*.

Note - To avoid any failures because of name service lookup, verify that all logical host names 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+. Doing so prevents timing-related errors in this area and ensures that `ifconfig` and `statd` succeed in resolving logical host names.

5. **Add the desired logical host name resources into the failover resource group.**

   You must set up a `LogicalHostname` resource with this step. The host name used with Sun Cluster HA for NFS *cannot* be a `SharedAddress` resource.

```
# scrgadm -a -L -g resource-group-name -l hostname, …
```

–a –L –g ***resource-group-name*** — Specifies the failover resource group into which to place the logical host name resources.

–l *hostname, …* — Specifies the network resources (logical host names) to be added.

6. **From any node of the cluster, create a directory structure for the NFS configuration files.**

   Create the administrative subdirectory directory below the directory specified by the `Pathprefix` property in Step 3 on page 114, for example, `/global/nfs/SUNW.nfs`.

```
# mkdir Pathprefix/SUNW.nfs
```

7. **Create a `dfstab.`*resource-name* file in the `SUNW.nfs` directory created in Step 6 on page 115 and set up share options.**

   For example, create *Pathprefix*`/SUNW.nfs/dfstab.`*resource-name*, 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 you are planning to create (in Step 9 on page 116). A good resource name would refer to the task 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 you have created to be shared. The format of this file is exactly the same as the format used in `/etc/dfs/dfstab`.

```
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. For Sun Cluster, `rw` is recommended. 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 host names used by Sun Cluster, and enables Sun Cluster HA for NFS fault monitoring to operate most efficiently. For details, see `dfstab`(4), `share`(1M), and `share_nfs`(1M).

If you specify a client list in the `share` command, include all physical and logical host names that are associated with the cluster, as well as the host names 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 host names to the appropriate net group.

Note - Do not grant access to the host names 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. In this case, Sun Cluster HA for NFS fault monitoring can still perform monitoring without having write access.

Note - When constructing share options, avoid using the root option and avoid mixing `ro` and `rw` options.

---

8. Register the NFS resource type.

```
# scrgadm -a -t resource-type-name
```

–a –t ***resource-type-name***	Adds the specified resource type. For Sun Cluster HA for NFS, the resource type is `SUNW.nfs`.

9. Create the NFS resource in the failover resource group.

```
# scrgadm -a -j resource-name -g resource-group-name -t resource-type-name
```

–a	Adds a resource.

–j ***resource-name***	Specifies the name of the resource to add, which you defined in Step 7 on page 115. This name can be your choice but must be unique within the cluster.
–g ***resource-group-name***	Specifies the name of a previously created resource group to which this resource is to be added.
–t ***resource-type-name***	Specifies the name of the resource type to which this resource belongs. This name must be the name of a registered resource type.

10. Enable the resource and the resource monitor, manage the resource group, and switch the resource group into the online state.

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

Example—Setting Up and Configuring Sun Cluster HA for NFS

The following example shows how to set up and configure Sun Cluster HA for NFS.

```
(Create a logical host resource group and specify the path to the administrative files used by NFS (Pathprefix).)
# scrgadm -a -g lh-schost-1 -y Pathprefix=/global/nfs

(Add logical hostname resources into the logical host resource group.)
# scrgadm -a -L -g lh-schost-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-name 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 lh-schost-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 lh-schost-1
```

Where to Go from Here

If you need to set share options for your NFS file systems, refer to "How to Change Share Options on an NFS File System" on page 118. If you want to review or set extension properties, refer to "Configuring Sun Cluster HA for NFS Extension Properties" on page 120.

▼ 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 host names to the appropriate `netgroup`. Ideally, grant both read and write access to all the Sun Cluster host names 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-name
```

–M Disables the resource monitor.

3. **Execute your proposed new `share`(1M) command.**

   Before editing the `dfstab.`*resource-name* file with new share options, execute the new `share(1M)` command to verify that the combination of options is valid.

```
# share -F nfs [-o] specific_options [-d ''description''] pathname
```

–F `nfs`	Identifies the file system type as NFS.
–o *specific_options*	Specifies an option. We suggest `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(1M)` command fails, immediately execute another `share(1M)` command with the old options. When the new command executes successfully, go on to the next step.

4. **Edit the `dfstab.`*resource-name* file with the new share options.**

   The format of this file is exactly the same as the format used in `/etc/dfs/dfstab`. Each line consists of a `share(1M)` command.

5. **(Optional) If you are removing a path from the `dfstab.`*resource-name* file, execute the `unshare(1M)` command, then remove the `share`(1M) command for the path from the `dfstab.`*resource-name* file.**

```
# unshare [-F nfs] [-o rw] pathname
# vi dfstab.resource-name
```

`–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-name* file, verify that the mount point is valid, then perform Step 3 on page 118 and Step 4 on page 119.**

7. **Enable fault monitoring on the NFS resource.**

```
# scswitch -e -M -j resource-name
```

▼ How to Tune Sun Cluster HA NFS Method Timeouts

The time HA NFS methods take to finish depends on the number of paths shared by the resources through the `dfstab.`*resource-name* file. The default timeout for these methods is 300 seconds. A simple rule of thumb is to allocate 10 seconds toward the method timeout 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-name* file contains 50 shared paths, the recommended timeout is 500 seconds.

The method timeouts that you should change are:

`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`(1M) `-c` option. For example:

```
% scrgadm -c -j resource-name -y Prenet_start_timeout=500
```

▼ How to Configure `SUNW.HAStorage` Resource Type

The `SUNW.HAStorage` resource type synchronizes actions between HA storage and data service. Because Sun Cluster HA for NFS is disk-intensive, we strongly recommend that you set up `SUNW.HAStorage`.

For details on the background, see the `SUNW.HAStorage`(5) man page and "Relationship Between Resource Groups and Disk Device Groups" on page 18. For the procedure, see "How to Set Up `SUNW.HAStorage` Resource Type for New Resources" on page 168.

Configuring Sun Cluster HA for NFS Extension Properties

You are not required to set any extension properties for Sun Cluster HA for NFS.

See Appendix A for details of all Sun Cluster properties.

▼ How to Configure Sun Cluster HA for NFS Extension Properties

Typically, you configure extension properties by using the Cluster Module of Sun Management Center or the command line `scrgadm -x` *parameter=value* at the time

you create the NFS resource. You can also configure them later by using the procedures described in Chapter 9.

Table 7–2 describes extension properties you can configure for Sun Cluster HA for NFS. Some extension properties can be updated dynamically, others only when the resource is created. The Tunable column indicates when the property can be updated.

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 the fault monitor is to be restarted by the process monitor facility during the time window specified by the `Monitor_retry_interval` property. Note that this property refers to restarts of the fault monitor itself, rather than to the resource. Restarts of the resource are controlled by the system defined properties `Retry_interval`, and `Retry_count`. See the `scrgadm` 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 the fault monitor fails is more than the value specified in the extension property `Monitor_retry_count` within this period, the fault monitor is not restarted by the process monitor facility.
`Mountd_nullrpc_restart` (Boolean)	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`

TABLE 7–2 Sun Cluster HA for NFS Extension Properties *(continued)*

Name/Data Type	Default	Tunable	Description
`Nfsd_nullrpc_restart` (Boolean)	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`
`Rpcbind_nullrpc_reboot` (Boolean)	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`

CHAPTER 8

Installing and Configuring Sun Cluster HA for Oracle Parallel Server

This chapter describes the steps for installing and configuring Sun Cluster HA for Oracle Parallel Server (OPS) on your Sun Cluster servers. It contains the following procedures:

Overview

You install and configure OPS by using the procedures in the Oracle documentation. Although OPS is not registered with or managed by the Sun Cluster Resource Group Manager (RGM), it depends on the RGM to query cluster information.

You can configure OPS to use the shared disk architecture of Sun Cluster. In this configuration, a single database is shared among multiple instances of OPS that access the database concurrently. Conflicting access to the same data is controlled by means of the Oracle UNIX Distributed Lock Manager (UDLM). If a process or a node crashes, the UDLM is reconfigured to recover from the failure.

In the event of a node failure in an OPS environment, you can configure Oracle clients to reconnect to the surviving server without the use of the IP failover used by

Sun Cluster failover data services. This failover process is described in the *Sun Cluster 3.0 Concepts* document.In an OPS environment, multiple Oracle instances cooperate to provide access to the same shared database. The Oracle clients can access the database by using any of the instances. Thus, if one or more instances have failed, clients can continue to access the database by connecting to a surviving instance.

Installing and Configuring Sun Cluster HA for Oracle Parallel Server

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" on page 124
Install Sun Cluster HA for Oracle Parallel Server packages	"How to Install Sun Cluster HA for Oracle Parallel Server Packages" on page 125
Install the UNIX Distributed Lock Manager and Oracle software	"Installing the Oracle Software" on page 126

Installing Volume Management Software

Two possible configurations for Sun Cluster HA for Oracle Parallel Server disks are:

- The VERITAS Volume Manager (VxVM) cluster functionality
- Sun StorEdge™ A3x00 with hardware RAID support

How to Install Volume Management Software

If you are using VxVM, you must first install and configure the VxVM software on the cluster nodes. For details, see the VxVM appendix in the *Sun Cluster 3.0*

Installation Guide and your VxVM documentation. A separate license is required for VxVM cluster operations; the Oracle documentation describes the licensing requirements.

Caution - Failure to install the VxVM cluster license correctly might result in a panic if OPS support is installed without VxVM functioning properly. Prior to installing the OPS packages, run the `vxlicense` check command to ensure that a valid cluster license is installed.

If you are using the Sun StorEdge A3x00 with hardware RAID support, you have no volume management software to install. In that case, you must install RAID Manager software for A3x00.

Installing Sun Cluster HA for Oracle Parallel Server Packages

Use this procedure to install the packages needed to run Sun Cluster HA for OPS.

▼ How to Install Sun Cluster HA for Oracle Parallel Server Packages

To complete this procedure, you need the Sun Cluster data services CD. Perform this procedure on all cluster nodes that can run Sun Cluster HA for Oracle Parallel Server.

1. **Load the data services CD into the CD-ROM drive.**

2. **Install the Sun Cluster HA for Oracle Parallel Server packages.**

   The packages vary, depending on whether you are using hardware RAID or VERITAS Volume Manager for your volume manager.

   a. **If you are using hardware RAID as your volume manager, install as follows:**

   ```
   # pkgadd -d . SUNWscucm SUNWudlm SUNWudlmr SUNWschwr
   ```

   b. **If you are using 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 126 to install the UDLM and Oracle software.

Installing the Oracle Software

Use the procedures in this section to do the following:

- 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, you must ensure that:

- 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.

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 Sun Cluster HA for Oracle Parallel Server. For example:

   ```
   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:

```
# useradd -u 120 -g dba -d /orahome oracle
```

   Ensure that the user IDs are the same on all the nodes that run Sun Cluster HA for Oracle Parallel Server.

Where to Go from Here

After setting up the cluster environment for OPS, install the UDLM software on each cluster node. For instructions, see your OPS installation documentation.

▼ How to Install the UDLM Software

You must install the UDLM software on the local disk of each node.

1. **Become superuser on a node in the cluster.**

2. **Install the UDLM software.**

   Refer to 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
```

(continued)

(Continuation)

```
forceload: sys/shmsys
forceload: sys/semsys
forceload: sys/msgsys
```

4. Shut down and reboot all the nodes.

Caution - Before rebooting, 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.

Refer to `scshutdown`(1M) for details.

First, shut down all of the nodes by typing the following command on one node:

```
phys-schost-1# scshutdown -g0 -y
```

For each node, type 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. For instructions, see the OPS installation documentation.

▼ How to Install the Oracle RDBMS Software

Refer to your OPS installation documentation for instructions on installing the RDBMS software.

Where to Go from Here

When installing the Oracle RDBMS software, create your Oracle database by using the instructions in the Oracle documentation.

CHAPTER 9

Administering Data Service Resources

This chapter describes the procedures used to manage resources, resource groups, and resource types within the cluster. If a procedure can be completed through the Sun Management Center GUI, that is noted before the procedure.

This chapter contains the following procedures:

- "How to Clear the STOP_FAILED Error Flag on Resources" on page 160
- "How to Re-register Preregistered Resource Types" on page 162
- "How to Add a Node to a Resource Group" on page 163
- "How to Remove a Node from a Resource Group" on page 165
- "How to Set Up SUNW.HAStorage Resource Type for New Resources" on page 168

For overview information about resource types, resource groups, and resources, see Chapter 1 and the *Sun Cluster 3.0 Concepts* document.

Administering Data Service Resources

Table 9–1 lists the sections that describe the administration tasks for data service resources.

TABLE 9–1 Task Map: Data Service Administration

Task	**For Instructions, Go To ...**
Register a resource type.	"How to Register a Resource Type" on page 133
Create failover or scalable resource groups.	"How to Create a Failover Resource Group" on page 134 "How to Create a Scalable Resource Group" on page 136

TABLE 9–1 Task Map: Data Service Administration *(continued)*

Task	For Instructions, Go To …
Add logical host names or shared addresses and data service resources to resource groups.	"How to Add a Logical Host Name Resource to a Resource Group" on page 138 "How to Add a Shared Address Resource to a Resource Group" on page 140 "How to Add a Failover Application Resource to a Resource Group" on page 142 "How to Add a Scalable Application Resource to a Resource Group" on page 144
Enable resources and resource monitors, manage the resource group, and bring it and its associated resources online.	"How to Bring a Resource Group Online" on page 146
Remove resource types from the cluster.	"How to Remove a Resource Type" on page 147
Remove resource groups from the cluster.	"How to Remove a Resource Group" on page 149
Remove resources from resource groups.	"How to Remove a Resource" on page 150
Switch the primary for a resource group.	"How to Switch the Current Primary of a Resource Group" on page 151
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 153
Display resource type, resource group, and resource configuration information.	"How to Display Resource Type, Resource Group, and Resource Configuration Information" on page 155

TABLE 9–1 Task Map: Data Service Administration *(continued)*

Task	For Instructions, Go To …
Change resource type, resource group, and resource properties.	"How to Change Resource Type Properties" on page 156 "How to Change Resource Group Properties" on page 157 "How to Change Resource Properties" on page 158
Clear error flags for failed Resource Group Manager (RGM) processes.	"How to Clear the `STOP_FAILED` Error Flag on Resources" on page 160
Re-register the built-in resource types `LogicalHostname` and `SharedAddress`	"How to Re-register Preregistered Resource Types" on page 162
Update the network interface 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 163
Remove a node from a resource group.	"How to Remove a Node from a Resource Group" on page 165
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 168

Configuring and Administering Sun Cluster Data Services

Configuring a Sun Cluster data service is a single task composed of several procedures. These procedures enable you to register a resource type, create resource groups, add resources into the resource groups, and bring the resources online.

You can use the Sun Management Center GUI or the command-line interface to initially configure Sun Cluster data services. For information on using the GUI to configure data services, refer to the *Sun Cluster 3.0 System Administration Guide*. The command-line interface is described in the installation and configuration chapter for each data service in this document.

To update your data service configuration after the initial configuration, use the procedures in this chapter or the Sun Management Center GUI.

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. For more information on resource types, see Chapter 1.

▼ 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. For the mapping between the names and the license certificate names, see the *Sun Cluster 3.0 Release Notes*.

Perform this procedure from any cluster node.

Refer to `scrgadm`(1M) for additional information.

1. **Become superuser on a node in the cluster.**

2. **Register the resource type.**

```
# scrgadm –a –t resource-type-name
```

`–a`	Adds the specified resource type.
`–t` ***resource-type-name***	Specifies name of the resource type to add. To determine the predefined name to supply, see the *Sun Cluster 3.0 Release Notes*.

3. **Verify that the resource type has been registered.**

```
# scrgadm -pv -t resource-type-name
```

Example–Registering Resource Types

The following example registers Sun Cluster HA for iPlanet Web Server (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. For details, see the following section, “Creating a Resource Group” on page 134.

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; a scalable resource group can be online on multiple nodes simultaneously.

You can create resource groups through the Sun Management Center GUI or by using the command line as shown in the following procedure. For conceptual information on resource groups, see Chapter 1 and the *Sun Cluster 3.0 Concepts* document.

▼ 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.

Refer to scrgadm(1M) for additional information.

Perform this procedure from any cluster node.

1. **Become superuser on a node in the cluster.**

2. **Create the failover resource group.**

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

–a	Adds the specified resource group.
–g ***resource-group-name***	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-name
```

Example–Creating a Failover Resource Group

This example shows the addition of a failover resource group (lh-rg-1) that can be mastered by two nodes (phys-schost-1 and phys-schost-2).

```
# scrgadm –a –g lh-rg-1 –h phys-schost1,phys-schost-2
# scrgadm –pv -g lh-rg-1
Res Group name:                                 lh-rg-1
  (lh-rg-1) Res Group RG_description:           <NULL>
  (lh-rg-1) Res Group management state:         Unmanaged
  (lh-rg-1) Res Group Failback:                 False
  (lh-rg-1) Res Group Nodelist:                 phys-schost-1 phys-schost-2
  (lh-rg-1) Res Group Maximum_primaries:        1
  (lh-rg-1) Res Group Desired_primaries:        1
  (lh-rg-1) Res Group RG_dependencies:          <NULL>
  (lh-rg-1) Res Group mode:                     Failover
  (lh-rg-1) Res Group network dependencies:     True
  (lh-rg-1) Res Group Global_resources_used:    All
  (lh-rg-1) Res Group Pathprefix:
```

Where to Go from Here

After creating a failover resource group, you can add application resources to this resource group. For the procedure, see “Adding Resources to Resource Groups” on page 138.

▼ 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 allows 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.

Refer to `scrgadm`(1M) for additional information.

Perform this procedure from any cluster node.

1. **Become superuser on a node in the cluster.**

2. **Create the failover resource group that holds the shared addresses to be used by the scalable resource.**

3. **Create the scalable resource group.**

```
# scrgadm -a -g ss-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 ***ss-resource-group-name***	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, it defaults to all nodes.

4. Verify that the scalable resource group has been created.

```
# scrgadm -pv -g resource-group-name
```

Example–Creating a Scalable Resource Group

This example shows the addition of a scalable resource group (`ss-rg-1`) to be hosted on two nodes (`phys-schost-1`, `phys-schost-2`). The scalable resource group depends on the failover resource group (`fo-rg-1`) that contains the shared addresses.

```
# scrgadm –a –g ss-rg-1 \
–y Maximum_primaries=2 \
–y Desired_primaries=2 \
–y RG_dependencies=fo-rg-1 \
–h phys-schost-1,phys-schost-2
# scrgadm –pv -g ss-rg-1
Res Group name:                                 ss-rg-1
  (ss-rg-1) Res Group RG_description:           <NULL>
  (ss-rg-1) Res Group management state:         Unmanaged
  (ss-rg-1) Res Group Failback:                 False
  (ss-rg-1) Res Group Nodelist:                 phys-schost-1 phys-schost-2
  (ss-rg-1) Res Group Maximum_primaries:        2
  (ss-rg-1) Res Group Desired_primaries:        2
  (ss-rg-1) Res Group RG_dependencies:          fo-rg-1
  (ss-rg-1) Res Group mode:                     Scalable
  (ss-rg-1) Res Group network dependencies:     True
  (ss-rg-1) Res Group Global_resources_used:    All
  (ss-rg-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 144 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 they can be managed by the RGM. Three types of resources are described in this section: logical host name resources or shared address resources and data service (application) resources.

Logical host name 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 host name resources and the application resources for the data service. Scalable resource groups contain only the application resources for scalable services. The shared addresses 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.

You can add resources to resource groups through the Sun Management Center GUI or by using the command line, as shown in this section.

For more information on resources, see the *Sun Cluster 3.0 Concepts* document and Chapter 1.

▼ How to Add a Logical Host Name 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 host names you are adding to the resource group

Refer to `scrgadm`(1M) for additional information.

Perform this procedure from any cluster node.

1. Become superuser on a node in the cluster.

2. Add the logical host name resource to the resource group.

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

–a Adds a logical host name resource.

–L	Specifies the logical host name resource form of the command.
–j ***resource-name***	Specifies an optional resource name of your choice. If you do not specify this option, the name defaults to the first host name specified with the –l option.
–g ***resource-group-name***	Specifies the name of the resource group in which this resource resides.
–l ***hostname, ...***	Specifies a comma-separated list of UNIX host names (logical host names) 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 scrgadm(1M) 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 identified by the *hostname* list for each node in *nodelist*.

3. Verify that the logical host name resource has been added.

```
# scrgadm -pv –j resource-name
```

The resource addition action cause the resource to be validated by the Sun Cluster software. If the validation succeeds, the resource can be enabled and the resource group can be moved into the state where it is managed by the RGM. If the validation fails, scrgadm produces an error message to that effect and exits. In that case, 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 Host Name Resource to a Resource Group

This example shows the addition of logical host name resource (lh-r-1) to a resource group (lh-rg-1).

```
# scrgadm -a -L -j lh-r-1 -g lh-rg-1 -l schost-1
# scrgadm -pv -j lh-r-1
RG Name: lh-rg-1
```

(continued)

```
(lh-rg-1) Res name:                          lh-r-1
  (lh-rg-1:lh-r-1) Res R_description:
  (lh-rg-1:lh-r-1) Res resource type:        SUNW.LogicalHostname
  (lh-rg-1:lh-r-1) Res resource group name:  lh-rg-1
  (lh-rg-1:lh-r-1) Res enabled:              False
  (lh-rg-1:lh-r-1) Res monitor enabled:      True
```

Where to Go from Here

After adding logical host name resources, use the procedure “How to Bring a Resource Group Online” on page 146 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 host names you are adding to the resource group.

Refer to `scrgadm`(1M) for additional information.

Perform this procedure from any cluster node.

1. Become superuser on a node in the cluster.

2. Add the shared address resource to the resource group.

```
# scrgadm –a –S [–j resource-name] -g resource-group-name –l hostname, … \
[–X auxnode-list] [–n netiflist]
```

–a	Adds shared address resources.
–S	Specifies the shared address resource form of the command.
–j ***resource-name***	Specifies an optional resource name of your choice. If you do not specify this option, the name defaults to the first host name specified with the –l option.

–g ***resource-group-name***	Specifies the resource group name.
–l ***hostname, ...***	Specifies a comma-separated list of shared address host names.
–X ***auxnode-list***	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 *network-interface-list*. See `scrgadm`(1M) 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 identified by the *hostname* list for each node in *nodelist*.

3. Verify that the shared address resource has been added and validated.

```
# scrgadm -pv -j resource-name
```

The resource addition action causes the resource to be validated by the Sun Cluster software. If the resource is successfully validated, it can be enabled and the resource group can be moved into the state where it is managed by the RGM. If the validation fails, `scrgadm` produces an error message to this effect and exits. In that case, 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 (`sa-r-1`) to a resource group (`sa-rg-1`).

```
# scrgadm -a -S -j sa-r-1 -g sa-rg-1 -l schost-1
# scrgadm -pv -j sa-r-1
(sa-rg-1) Res name:                             sa-r-1
    (sa-rg-1:sa-r-1) Res R_description:
    (sa-rg-1:sa-r-1) Res resource type:         SUNW.SharedAddress
```

(continued)

```
(sa-rg-1:sa-r-1) Res resource group name:  sa-rg-1
(sa-rg-1:sa-r-1) Res enabled:              False
(sa-rg-1:sa-r-1) Res monitor enabled:      True
```

Where to Go from Here

After adding a shared resource, enable it by following the procedure “How to Bring a Resource Group Online” on page 146.

▼ How to Add a Failover Application Resource to a Resource Group

A failover application resource is an application resource that uses logical host names 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 host name resources used by the application resource, which are the logical host names previously included in the same resource group

Refer to `scrgadm`(1M) for additional information.

Perform this procedure from any cluster node.

1. Become superuser on a node in the cluster.

2. Add a failover application resource to the resource group.

```
# scrgadm –a –j resource-name –g resource-group-name –t resource-type-name \
[–x Extension_property=value, …] [-y Standard_property=value, …]
```

–a	Adds a resource.
–j ***resource-name***	Specifies your choice of the name of the resource to add.
–g ***resource-group-name***	Specifies the name of the failover resource group created previously.

–t ***resource-type-name***	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 it is required.
–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 it is required.

Note - You can set additional properties. For details, see Appendix A and the chapter in this book on installing and configuring your failover data service.

---

3. Verify that the failover application resource has been added and validated.

```
# scrgadm -pv -j resource-name
```

The resource addition action causes the resource to be validated by the Sun Cluster software. If the validation succeeds, the resource can be enabled and the resource group can be moved into the state where it is managed by the RGM. 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 (`fo-r-1`) to a resource group (`fo-rg-1`). The resource depends on logical host name resources (`schost-1`, `schost-2`), which must reside in the same failover resource groups defined previously.

```
# scrgadm -a -j fo-r-1 -g fo-rg-1 -t rt-1 \
-y Network_resources_used=schost-1,schost2 \
# scrgadm -pv -j fo-r-1
(fo-rg-1) Res name:                          fo-r-1
    (fo-rg-1:fo-r-1) Res R_description:
    (fo-rg-1:fo-r-1) Res resource type:      rt-1
    (fo-rg-1:fo-r-1) Res resource group name:  fo-rg-1
```

(continued)

```
(fo-rg-1:fo-r-1) Res enabled:              False
(fo-rg-1:fo-r-1) Res monitor enabled:      True
```

Where to Go from Here

After adding a failover application resource, enable it by following the procedure "How to Bring a Resource Group Online" on page 146.

▼ 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 used by the scalable service resource, which are the shared addresses previously included in a failover resource group

Refer to `scrgadm`(1M) for additional information.

Perform this procedure from any cluster node.

1. Become superuser on a node in the cluster.

2. Add a scalable application resource to the resource group.

```
# scrgadm –a –j resource-name –g resource-group-name –t resource-type-name \
–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-name***	Specifies your choice of the name of the resource to add.
–g ***resource-group-name***	Specifies the name of a scalable service resource group created previously.

`-t` ***resource-type-name*** Specifies the name of the resource type for this resource.

`-y Network_resources_used=`***network-resource*`[,`*`network-resource`*`...]` Specifies the 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 it is required.

`-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 it is required.

Note - You can set additional properties. For information on other configurable properties, see Appendix A and the chapter in this book on installing and configuring your scalable data service. Specifically for scalable services, you would normally set the `Port_list`, `Load_balancing_weights`, and `Load_balancing_policy` properties, which are described in Appendix A.

---

3. Verify that the scalable application resource has been added and validated.

```
# scrgadm -pv -j resource-name
```

The resource addition action causes the resource to be validated by the Sun Cluster software. If the validation succeeds, the resource can be enabled and the resource group can be moved into the state where it is managed by the RGM. 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 (`ss-r-1`) to a resource group (`ss-rg-1`). Note that `ss-rg-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 defined previously.

```
# scrgadm -a -j ss-r-1 -g ss-rg-1 -t rt-1 \
-y Network_resources_used=schost-1,schost-2 \
–y Scalable=True
# scrgadm -pv -j ss-r-1
(ss-rg-1) Res name:                          ss-r-1
    (ss-rg-1:ss-r-1) Res R_description:
    (ss-rg-1:ss-r-1) Res resource type:      rt-1
    (ss-rg-1:ss-r-1) Res resource group name:  ss-rg-1
    (ss-rg-1:ss-r-1) Res enabled:            False
    (ss-rg-1:ss-r-1) Res monitor enabled:    True
```

Where to Go from Here

After adding a scalable application resource, enable it by following the procedure “How to Bring a Resource Group Online” on page 146.

Bringing Resource Groups Online

To allow 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 one-step procedure below. For details, see `scswitch(1M)`.

Perform this procedure from any cluster node.

▼ How to Bring a Resource Group Online

1. **Become superuser on a node in the cluster.**

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-name
   ```

   –Z — Brings a resource group online by first enabling its resources and their monitors.

   –g ***resource-group-name*** — 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 it 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 (`rg-1`) online and verify its status.

```
# scswitch -Z -g rg-1
# scstat -g
```

Where to Go from Here

After a resource group has been brought online, it is configured and ready to use. In the event of resource or node failure, the RGM maintains availability of the resource group by automatically switching the resource group online on alternate nodes.

Removing Resource Types

You need not remove resource types that are not being used. However, if you want to remove a resource type, you can use this procedure to do so.

For additional information, see `scrgadm`(1M) and `scswitch`(1M).

Perform this procedure from any cluster node.

▼ How to Remove a Resource Type

Before removing 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 node in the cluster.**

2. **Disable each resource of the resource type to be removed.**

```
# scswitch –n –j resource-name
```

–n	Disables the resource.
–j ***resource-name***	Specifies the name of the resource to disable.

3. **Remove each resource of the resource type to be removed.**

```
# scrgadm –r –j resource-name
```

–r Removes the specified resource.

–j Specifies the name of the resource to remove.

4. **Remove the resource type.**

```
# scrgadm –r –t resource-type-name
```

–r Removes the specified resource type.

–t ***resource-type-name*** 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.

Perform this procedure from any cluster node.

For additional information, see `scrgadm`(1M) and `scswitch`(1M).

▼ How to Remove a Resource Group

1. **Become superuser on a node in the cluster.**

2. **Run the following command to take the resource group offline.**

```
# scswitch –F –g resource-group-name
```

–F — Switches a resource group offline.

–g ***resource-group-name*** — 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-name
```

–n — Disables the resource.

–j ***resource-name*** — 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:

   - Remove the resources
   - Remove the resource group

```
# scrgadm –r –j resource-name
# scrgadm –r –g resource-group-name
```

–r — Removes the specified resource or resource group.

–j ***resource-name*** — Specifies the name of the resource to be removed.

–g ***resource-group-name*** — 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 (`rg-1`) after you have removed its resource (`resource-1`).

```
# scswitch –F –g rg-1
# scrgadm -r -j resource-1
# scrgadm -r -g rg-1
```

Removing Resources

Disable the resource before removing it from a resource group.

For additional information, see `scrgadm`(1M) and `scswitch`(1M).

Perform this procedure from any cluster node.

▼ How to Remove a Resource

1. **Become superuser on a node in the cluster.**

2. **Disable the resource for the resource that you want to remove.**

```
# scswitch –n –j resource-name
```

–n — Disables the resource.

–j ***resource-name*** — Specifies the name of the resource to disable.

3. **Remove the resource.**

```
# scrgadm –r –j resource-name
```

–r — Removes the specified resource.

–j ***resource-name*** 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.

For additional information, see `scrgadm`(1M) and `scswitch`(1M).

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 node in the cluster.

2. Switch the primary to a potential primary.

```
# scswitch –z –g resource-group-name –h nodelist
```

–z	Switches the specified resource group online.
–g ***resource-group-name***	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 (`rg-1`) from its current primary (`phys-schost-1`) to the potential primary (`phys-schost-2`). You 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:          rg-1
  Status
    Node Name:                phys-schost-1
    Status:                   Online

    Node Name:                phys-schost-2
    Status:                   Offline
...
phys-schost-1# scswitch –z –g rg-1 –h phys-schost-2
phys-schost-1# scstat –g
...
Resource Group Name:          rg-1
  Status
    Node Name:                phys-schost-2
    Status:                   Online

    Node Name:                phys-schost-1
    Status:                   Offline
...
```

(continued)

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.

For additional information, see `scrgadm`(1M) and `scswitch`(1M).

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 node in the cluster.**

2. **Disable the resource.**

   Repeat this step for all resources in the resource group.

   ```
   # scswitch –n –j resource-name
   ```

   –n — Disables the resource.

   –j ***resource-name*** — Specifies the name of the resource to disable.

3. **Run the following command to take the resource group offline.**

```
# scswitch –F –g resource-group-name
```

–F — Switches a resource group offline.

–g ***resource-group-name*** — Specifies the name of the resource group to take offline.

4. Bring the resource group into the unmanaged state.

```
# scswitch -u -g resource-group-name
```

–u — Puts the specified resource group in the unmanaged state.

–g ***resource-group-name*** — 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-name
```

Example–Disabling a Resource and Moving the Resource Group Into the Unmanaged State

This example shows how to disable the resource (`r-1`) and then move the resource group (`rg-1`) into the unmanaged state.

```
# scswitch -n -j r-1
# scswitch -F -g rg-1
# scswitch -u -g rg-1
# scrgadm -pv -g rg-1
Res Group name:                              rg-1
  (rg-1) Res Group RG_description:           <NULL>
  (rg-1) Res Group management state:         Unmanaged
  (rg-1) Res Group Failback:                 False
  (rg-1) Res Group Nodelist:                 phys-schost-1 phys-schost-2
  (rg-1) Res Group Maximum_primaries:        2
  (rg-1) Res Group Desired_primaries:        2
  (rg-1) Res Group RG_dependencies:          <NULL>
  (rg-1) Res Group mode:                     Failover
  (rg-1) Res Group network dependencies:     True
  (rg-1) Res Group Global_resources_used:    All
  (rg-1) Res Group Pathprefix:

  (rg-1) Res name:                           r-1
    (rg-1:r-1) Res R_description:
```

(continued)

(Continuation)

```
(rg-1:r-1) Res resource type:            SUNW.apache
(rg-1:r-1) Res resource group name:      rg-1
(rg-1:r-1) Res enabled:                  True
(rg-1:r-1) Res monitor enabled:          False
(rg-1:r-1) Res detached:                 False
```

Displaying Resource Type, Resource Group, and Resource Configuration Information

Before performing administrative procedures on resources, resource groups, or resource types, use this procedure to view the current configuration settings for these objects.

You can display resource type, resource group, and resource configuration information through the Sun Management Center GUI or by using the command line as shown in this section.

For additional information, see `scrgadm`(1M) and `scswitch`(1M).

Perform this procedure from any cluster node.

▼ How to Display Resource Type, Resource Group, and Resource Configuration Information

The `scrgadm` command provides three levels of configuration status information, as follows:

- 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
```

For details, see the `scrgadm`(1M) man page.

Changing Resource Type, Resource Group, and Resource Properties

Resource groups and resources have standard configuration properties you can change. Resources also have extension properties, some of which are predefined by the data service developer, which you cannot change. See the individual data service chapters in this document for a list of the extension properties for each data service.

For information on the standard configuration properties for resource groups and resources, see `scrgadm`(1M).

You can change resource group and resource properties through the Sun Management Center GUI or by using the command line, as shown in this section. For the set of properties you can change through the GUI, see the Sun Management Center online help.

▼ 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.

Perform this procedure from any cluster node.

1. **Become superuser on a node in the cluster.**

2. **Use 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-name –h installed-node-list
```

–c	Changes the specified resource type property.
–t ***resource-type-name***	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-name
```

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. For a complete list of resource group properties, see Appendix A.

Perform this procedure from any cluster node.

1. **Become superuser on a node in the cluster.**

2. **Change the resource group property.**

```
# scrgadm –c –g resource-group-name –y property=new-value
```

–c — Changes the specified property.

–g ***resource-group-name*** — 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-name
```

Example–Changing a Resource Group Property

This example shows how to change the `Failback` property for the resource group (`ss-rg-1`).

```
# scrgadm -c -g ss-rg-1 -y Failback=True
# scrgadm -pv -g ss-rg-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. For a complete list of resource group properties, see Appendix A.

Perform this procedure from any cluster node.

1. **Become superuser on a node in the cluster.**

2. **Use the `scrgadm -pvv` command to view the current resource property settings.**

```
# scrgadm -pvv -j resource-name
```

3. Change the resource property.

```
# scrgadm –c –j resource-name –y property=new-value | –x extension-property=new-value
```

–c	Changes the specified property.
–j ***resource-name***	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 installing and configuring the individual data services.

4. Verify that the resource property has been changed.

```
# scrgadm pvv -j resource-name
```

Example–Changing a Standard Resource Property

This example shows how to change the system-defined `Start_timeout` property for the resource (`r-1`).

```
# scrgadm -c -j r-1 -y start_timeout=30
# scrgadm -pvv -j r-1
```

Example–Changing an Extension Resource Property

This example shows how to change an extension property (`Log_level`) for the resource (`r-1`).

```
# scrgadm -c -j r-1 -x Log_level=3
# scrgadm -pvv -j r-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 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

For additional information, see scswitch(1M).

Perform this procedure from any cluster node.

1. **Become superuser on a node in the cluster.**

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-name –f STOP_FAILED
```

–c	Clears the flag.
–h ***nodelist***	Specifies the node names on which the resource was running.

–j ***resource-name*** 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 on page 160. It should now be** OFFLINE **or** ONLINE**.**

```
# scstat –g
```

If the resource group remains in the ERROR_STOP_FAILED state, as shown by scstat –g, take the resource group offline on those nodes where it is still in the ERROR_STOP_FAILED state by using the following scswitch command.

```
# scswitch –F –g resource-group-name
```

–F Takes the resource group offline on all nodes that can master the group.

–g ***resource-group-name*** 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 page 160 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 host name and shared address resources use these resource types. You never need to register these two resource types, but it is possible to accidentally delete them. If you have deleted resource types inadvertently, use the following procedure to re-register them.

For additional information, see scrgadm(1M).

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 two procedures:

- To configure a cluster node to be an additional master of a resource group
- 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 are used by both failover and scalable services. Each IP subnetwork connected to the cluster has its own network resource specified and included in a failover resource group. This is either a logical host name 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 used by the scalable resource.

For additional information, see `scrgadm`(1M).

Run either of these procedures from any cluster node.

▼ How to Add a Node to a Resource Group

To complete this procedure, you must supply the following information:

- 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:

- Be sure to verify that the new node is already a cluster member.
- For failover resource groups, perform all the steps below.
- For scalable resource groups, you must do the following:
  1. For each network resource used by a scalable resource in the resource group, make the resource group where the network resource is located run on the new node (Steps 1 through 4 below).
  2. Add the new node to the list of nodes that can master the scalable resource group (the `nodelist` resource group property) (Step 3 below).
  3. (Optional) Update the Load_balancing_weights property of the scalable resource to assign a weight to the node you wish to add to the resource group. Otherwise, the weight defaults to 1. See `scrgadm`(1M).

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 one for `netiflist` identifies them by node ID.

---

2. Update `netiflist` for the network resources affected by the node addition.

This step overwrites the previous value of *netiflist*, so you must include all NAFO groups here. Also, you must input nodes to `netiflist` by node ID. To find the node ID, use `scconf -p`.

```
# scrgadm -c -j network-resource -x netiflist=netiflist
```

–c	Changes a network resource.
–j ***network-resource***	Specifies the name of the network resource (logical host name 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` so 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 (`rg-1`), which contains a logical host name resource (`schost-2`).

```
# scrgadm -pvv -g rg-1 | grep -i nodelist
(rg-1) Res Group Nodelist:    phys-schost-1 phys-schost-3
# scrgadm -pvv -g rg-1 | grep -i netiflist
(rg-1:schost-2) Res property name: NetIfList
(rg-1:schost-2:NetIfList) Res property class: extension(rg-1:schost-2:NetIfList) List of NAFO int
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 rg-1 -h phys-schost-1,phys-schost-2,phys-schost-3
# scrgadm -pvv -g rg-1 | grep -i nodelist
(rg-1) Res Group Nodelist:     phys-schost-1 phys-schost-2 phys-schost-
3
# scrgadm -pvv -g rg-1 | grep -i netiflist
(rg-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:

- Be sure to verify that the resource group is *not* mastered on the node you will remove. If that's not the case, use `scswitch`(1M) to take the resource group offline on the node you want to remove.
- For failover resource groups, perform all the steps below.
- For scalable resource groups, you must do the following:
  1. Remove the node from the list of nodes that can master the scalable resource group (the `nodelist` resource group property) (Step 1 below).
  2. (Optional) For each network resource used by a scalable resource in the resource group, update the resource group where the network resource is located to *not* be mastered on the removed node (Steps 1 through 4 below).
  3. (Optional) Update the `Load_balancing_weights` property of the scalable resource to remove the weight of the node you wish to remove from the resource group. See `scrgadm`(1M).

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 configured for each resource in the resource group.

```
# scrgadm -pvv -g resource-group | grep -i netiflist
```

Note - The output of the above command lines identifies the nodes by node ID.

---

3. Update `netiflist` for network resources affected by the removal of the node.

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. To find the node ID, use `scconf -p`.

```
# scrgadm -c -j network-resource -x netiflist=netiflist
```

–c	Changes a network resource.
–j ***resource-group***	Specifies the name of the network resource (logical host name 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*.

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 (rg-1), which contains a logical host name resource (schost-1).

```
# scrgadm -pvv -g rg-1 | grep -i nodelist
(rg-1) Res Group Nodelist:       phys-schost-1 phys-schost-2 phys-schost-3
# scrgadm -c -g rg-1 -h phys-schost-1,phys-schost-2
# scrgadm -pvv -g rg-1 | grep -i netiflist
(rg-1:schost-1) Res property name: NetIfList(rg-1:schost-1:NetIfList) Res property class: extension(rg-1:schost-1:NetIfL
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 rg-1 | grep -i nodelist
(rg-1) Res Group Nodelist:       phys-schost-1 phys-schost-2
# scrgadm -pvv -g rg-1 | grep -i netiflist
(rg-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 may take a while 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 used by the data service 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, you should 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 `rg-1` contains three data services:

- iWS, which depends on `/global/rg-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 `rg-1`, do the following:

1. **Become superuser on a node in the cluster.**

2. **Create the resource group** `rg-1`**.**

```
# scrgadm -a -g rg-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 rg-1 -t SUNW.HAStorage \  -x ServicePaths=/global/rg-1,/dev/globa
```

`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** `rg-1` **and set their dependency to** `hastorage-1`**. For example, for iWS, type:**

```
# scrgadm -a -j resource-name -g rg-1 -t SUNW.iws \

-x Confdir_list=/global/iws/schost-1 \  -y Scalable=False -y Network_resources_used=schost-1 \
```

7. **Set** `rg-1` **to the managed state and bring it online.**

```
# scswitch -Z -g rg-1
```

`SUNW.HAStorage` 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`. For details, see the `SUNW.HAStorage`(5) man page.

▼ How to Set Up `SUNW.HAStorage` Resource Type for Existing Resources

To create a `SUNW.HAStorage` resource for existing resources, do the following:

1. **Register the resource type.**

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

2. **Create the** `SUNW.HAStorage` **resource** `hastorage-1`**, for example:**

```
# scrgadm -a -g resource-group-name -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-name -y Resource_Dependencies=hastorage-1
```

CHAPTER 10

Understanding Data Service Fault Monitors

This chapter describes the fault monitors supplied with each Sun Cluster data service and how each monitor operates. It contains the following sections:

Sun Cluster Data Service Fault Monitors

The data services supplied by Sun 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 fault monitor is invoked by the RGM 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 two functions:

- Monitors the abnormal exit of the data service server process.
- 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 process. On abnormal exit, the PMF invokes an action script supplied by the data service to communicate the failure to the data service fault monitor.

This communication between the PMF action script and the probe occurs over a UNIX domain socket. The only communication intended to take place through the UNIX domain socket is when the PMF informs the probe, through the action script, that the data service has exited abnormally. This event is considered a total failure of the data service.

The data service fault probe runs in an infinite loop and sleeps for an adjustable amount of time set by the resource property `Thorough_probe_interval`. While sleeping, the probe polls for messages from the PMF action script. If the server process exits abnormally during this interval, the PMF action script informs the probe.

The probe then updates the status of the data service as “Service daemon not running” and takes action. The action can involve just restarting the data service locally or failing over the data service to a secondary cluster node. The probe decides whether to restart or to fail over the data service by checking 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.

If no messages have been received on the control socket, the probe, after sleeping for an interval specified by `Thorough_probe_interval`, checks the health of the data service. The logic followed by the probe is as follows:

1. Sleep (`Thorough_probe_interval`).
2. Perform health checks under a time-out property `Probe_timeout`. This is a resource extension property of each data service that you can set.
3. If the result of Step 2 is a success, that is, the service is healthy, update the success/failure history by purging 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. It then computes the total number of times the health check failed.

The result of the health check can range from a total 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 it 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. In that case, a data service fault monitor can decide not to treat this "partial" failure as fatal. Instead, the monitor can just track this failure as a nonfatal probe of the service. These partial failures are still accumulated over the interval specified in `Retry_interval`.

However, if the probe cannot connect to the server at all, it can be considered a fatal failure. Partial failures lead to incrementing the failure count by a fractional amount. A fatal (total) failure always increments the failure count by 1. Every time the failure count increases by 1 (either by a fatal failure or by accumulation of partial failures), the probe attempts to correct the situation either by restarting or failing over the data service.

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 `scha_control` request issued in the previous step can be denied by the Sun Cluster framework because of various reasons; the reason is identified by the return code of `scha_control`. The probe checks the return code. If the `scha_control` is denied, the probe resets the failure/success history and starts afresh. The reason for this action is that because the number of failures is already above `Retry_count`, the fault probe would attempt to issue `scha_control` in each subsequent iteration (which is to be denied again). This request would place additional load on the system and increase the likelihood of further service failures in the case where they have been triggered by an overloaded system. The probe then returns to Step 1.

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, it 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 set by the resource property `Probe_timeout` 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 receives 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 Apache data service and records the failure in its history. An Apache probe failure can be a total failure or a partial failure.

   Probe failures that are considered total failures are:

   - Failure to connect to the server, as flagged by the error message: `Failed to connect to %s port %d`, with `%s` being the host name and `%d` the port number.
   - 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 flagged by the error message: `Failed to communicate with server %s port %d: %s`, with the first `%s` being the host name, `%d` the port number, and the second `%s` further details about the error.

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

     - 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 flagged by the error message: `Failed to communicate with server %s port %d: %s`, with the first `%s` being the host name and `%d` the port number; the second `%s` further details about the error.
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 172.

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 by using the time-out value specified by the resource property `Probe_timeout`.

   The result of this `nslookup` command can be either failure or success. If the `nslookup` query was successfully replied to by DNS, 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 total failure

2. Based on the success/failure history, a failure can cause a local restart or a data service failover. This action is further described in "Health Checks of the Data Service" on page 172.

Sun Cluster HA for NFS Fault Monitor

The Sun Cluster HA for NFS fault monitor contains two parts. One is NFS system fault monitoring, which involves monitoring the NFS daemons (`nfsd`, `mountd`, `statd`, and `mountd`) and taking appropriate action when they have problems. The other part is specific to each NFS resource. The fault monitor of each resource monitors the file systems exported by the resource by checking the status of each shared path.

Fault Monitor Startup

The NFS system fault monitor is started by a NFS resource start method. 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. It also stops the NFS system fault monitor if no other NFS resource fault monitor is running on the local node.

NFS System Fault Monitor Process

The system fault monitor probes `rpcbind`, `statd`, `lockd`, `nfsd`, and `mountd` by checking for the presence of the process and its response to a null `rpc` call. 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`

For a description of these properties, see Chapter 7.

Each system fault monitor probe cycle does the following:

1. Sleeps for `Cheap_probe_interval`.
2. Probes `rpcbind`.

   If the process dies, 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 dies, restarts both.

   If a null `rpc` call fails, logs a message to syslog but does not restart.
4. Probe `mountd` and `mountd`.

   If the process dies, restart it.

   If a null `rpc` call fails, restart `mountd` if the PXFS device is available and the extension property `Mountd_nullrpc_restart=True`.

If any of the NFS daemons fails 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 does the following:

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 doing `stat()` of the path.

If any path is bad, the resource status is set to `FAULTED`. If all paths are working, the resource status is set to `ONLINE` again.

Sun Cluster HA for Oracle Fault Monitor

The two fault monitors for Sun Cluster HA for Oracle 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 two processes: a main fault monitor process and database client fault probe. The main process performs error lookup and `scha_control` actions. The database client fault probe 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. Some possible actions are: switchover, stopping and restarting the server, and stopping and restarting the resource group.

The probe uses the time-out value 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 it finds.

The server fault monitor is started through `pmfadm` to make it highly available. If the monitor is killed for any reason, it is automatically restarted by `pmf`.

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 it highly available. If it is killed, it is automatically restarted by `pmf`.

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

Sun Cluster HA for iPlanet Web Server Fault Monitor

The probe for Sun Cluster HA for iPlanet Web Server (iWS) 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 two configurations of iWS: the secure instance and 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 set by the resource property `Probe_timeout` to limit the time spent trying to successfully probe iWS. For details on this resource property, see Appendix A.

The `Network_resources_used` resource property setting on the iWS resource determines the set of IP addresses that are used by the Web server. The `Port_list` resource property setting determines the list of port numbers in use by iWS. The fault monitor assumes that the Web server is listening on all combinations of IP and port. If you are customizing 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 connects to the Web server by using the specified IP address and port combination. If the connection is not successful, the probe concludes that a total failure has occurred. The probe then records the failure and takes appropriate action.
2. If the probe successfully connects, it checks to see if the Web server is being run in a secure mode. If so, the probe just 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 a 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 be listening 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 are creating 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 Sun Cluster HA for iPlanet Web Server. The failure is recorded in the probe's history.

A probe failure can be a total or partial failure. Probe failures that are considered total failures are:

- Failure to connect to the server, as flagged by the error message: `Failed to connect to %s port %d`, with `%s` being the host name and `%d` the port number.
- 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 flagged by the error message: `Failed to communicate with server %s port %d: %s`, with the first `%s` being the host name and `%d` the port number; the second `%s` further details about the error.

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

- 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 flagged by the error message: `Failed to communicate with server %s port %d: %s`, with the first `%s` being the host name and `%d` the port number; the second `%s` further details about the error.

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 172.

Sun Cluster HA for Netscape Directory Server Fault Monitor

The probe for Sun Cluster HA for Netscape Directory Server accesses particular IP addresses and port numbers. The IP addresses are those from network resources listed in the `Network_resources_used` resource property. The port is the one

listed in the `Port_list` resource property. For a description of these properties, see Appendix A.

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, it 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 total failure.

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

- `Probe_timeout` duration is exceeded while probing the set of IP addresses for the port. Potential causes are:
  - 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) being monitored.
- A problem other than timeout occurs while invoking `ldapsearch`. Note that this scenario does not cover the case where `ldapsearch` is invoked successfully but returns an error.

APPENDIX **A**

Standard Properties

This appendix describe the standard resource type, resource group, and resource properties. It also describes the resource property attributes available for changing system-defined properties and creating extension properties.

The following is a list of the information in this appendix:

Note - The property values, such as `True` and `False`, are *not* case sensitive.

Resource Type Properties

Table A–1 describes the resource type properties defined by Sun Cluster. 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 it 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 it and it 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 it and it 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 it isn't, the RGM creates it 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

Property Name	Description	Updatable	Category
`API_version` (integer)	The version of the resource management API used by this resource type implementation. 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

TABLE A–1 Resource Type Properties *(continued)*

Property Name	Description	Updatable	Category
`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
`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

TABLE A–1 Resource Type Properties *(continued)*

Property Name	Description	Updatable	Category
`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. It must be a complete path, that is, it 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 *(continued)*

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)

TABLE A–1 Resource Type Properties *(continued)*

Property Name	Description	Updatable	Category
`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)
`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 defined by Sun Cluster. 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** — Cannot be set directly by an administrative tool.

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

Property Name	Description	Updatable	Category
`Cheap_probe_interval` (integer)	The number of seconds between invocations of a quick fault probe of the resource. This property is only created by the RGM and available to the administrator if it 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

TABLE A–2 Resource Properties *(continued)*

Property Name	Description	Updatable	Category
`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 *(continued)*

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 it is coming to. The default value is `Lb_weighted`.	At creation	Conditional Optional

TABLE A–2 Resource Properties *(continued)*

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

TABLE A–2 Resource Properties *(continued)*

Property Name	Description	Updatable	Category
`Monitored_switch` (enum)	Set to `Enabled` or `Disabled` by the RGM 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 it is enabled again. If the resource does not have a monitor callback method, this property does not exist. The default is `Enabled`.	Never	Query-only
`Network_resources_used` (string array)	A list of logical host name or shared address network resources used by the resource. 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 host name 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 it 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)	Set to `Enabled` or `Disabled` by the RGM if the cluster administrator enables or disables the resource with an administrative utility. If disabled, a resource has no callbacks invoked until it is enabled again. The default is `Disabled`.	Never	Query-only

TABLE A–2 Resource Properties *(continued)*

Property Name	Description	Updatable	Category
`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 being used by that port, 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 it 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 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
`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

TABLE A–2 Resource Properties *(continued)*

Property Name	Description	Updatable	Category
`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
`Retry_count` (integer)	The number of times a monitor attempts to restart a resource if it fails. This property is created by the RGM only and available to the administrator if it is declared in the RTR file. It 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 *(continued)*

Property Name	Description	Updatable	Category
`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`. This property is created by the RGM only and available to the administrator if it is declared in the RTR file. It 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 *(continued)*

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 the following scalable service properties for resources of that type: `Network_resources_used`, `Port_list`, `Load_balancing_policy`, and `Load_balancing_weights`. These properties have their default values unless they are explicitly declared in the RTR file. The default for `Scalable`—when it 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)	Set by the resource monitor. 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 it is brought offline.	Never	Query-only
`Status_msg`: on each cluster node (string)	Set by the resource monitor at the same time as the `Status` property. This property is settable per resource per node. The RGM sets it to the empty string when the resource is brought offline.	Never	Query-only

TABLE A–2 Resource Properties *(continued)*

Property Name	Description	Updatable	Category
`Thorough_probe_interval` (integer)	The number of seconds between invocations of a high-overhead fault probe of the resource. This property is created by the RGM only and available to the administrator if it is declared in the RTR file. It 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 defined by Sun Cluster. 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** — Cannot be set directly by an administrative tool.

The Updatable column shows whether the property is updatable (Y) or not (N) after it 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 cluster file systems are used by any resource in this resource group. 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 host name 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) used by the RGM to determine where to bring the resource group online in the event of a reconfiguration or as the result of an `scha_control giveover` command or function being executed. In the event of 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

TABLE A–3 Resource Group Properties *(continued)*

Property Name	Description	Updatable	Category
`RG_description` (string)	A brief description of the resource group. The default is the empty string.	Y	Optional
`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 the group can be mastered by multiple nodes 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)	Set by the RGM 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 it 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 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 defined by the resource type implementation. 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.

TABLE A–4 Resource Property Attributes *(continued)*

Property	Description
`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

RGM names fall into 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 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, or semicolon.