Oracle® Solaris Cluster 4.3 Geographic Edition System Administration Guide

Part No: E56741
June 2017

ORACLE

Oracle Solaris Cluster 4.3 Geographic Edition System Administration Guide

Part No: E56741

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit `http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info` or visit `http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs` if you are hearing impaired.

Référence: E56741

Accès aux services de support Oracle

Les clients Oracle qui ont souscrit un contrat de support ont accès au support électronique via My Oracle Support. Pour plus d'informations, visitez le site `http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info` ou le site `http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs` si vous êtes malentendant.

Contents

Figures

Tables

Examples

Using This Documentation

- **Overview** – Provides procedures for administering Oracle Solaris Cluster Geographic Edition (Geographic Edition) software.
- **Audience** – Experienced system administrators with extensive knowledge of Oracle software and hardware.
- **Required knowledge** – Knowledge of the Oracle Solaris operating system, of Oracle Solaris Cluster software, and expertise with the volume manager software that is used with Oracle Solaris Cluster software.

This document is not to be used as a planning or presales guide.

Product Documentation Library

Documentation and resources for this product and related products are available at http://www.oracle.com/pls/topic/lookup?ctx=E56676-01.

Feedback

Provide feedback about this documentation at http://www.oracle.com/goto/docfeedback.

♦ ♦ ♦ CHAPTER 1

Introduction to Administering the Geographic Edition Framework

Oracle Solaris Cluster Geographic Edition (Geographic Edition) framework protects applications from unexpected disruptions by using multiple clusters that are geographically separated. These clusters contain identical copies of the Geographic Edition infrastructure, which manage replicated data between the clusters. Geographic Edition framework is a layered extension of Oracle Solaris Cluster software.

This chapter contains the following sections:

Geographic Edition Administration

Before beginning administration tasks, familiarize yourself with the planning information in the *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide* and the *Oracle Solaris Cluster 4.3 Geographic Edition Overview*. This guide contains the standard tasks that are used to administer and maintain the Geographic Edition configurations.

For general Oracle Solaris Cluster, data service, and hardware administration tasks, refer to the Oracle Solaris Cluster documentation.

You can perform all administration tasks on a cluster that is running the Geographic Edition framework without causing any nodes or the cluster to fail. You can install, configure, start, use, stop, and uninstall the Geographic Edition framework on an operational cluster.

Note - You might be required to take nodes or the cluster offline for preparatory actions, such as installing data replication software and performing Oracle Solaris Cluster administrative tasks. Refer to the appropriate product documentation for administration restrictions.

Geographic Edition Administration Tools

You can perform administrative tasks on a cluster that is running the Geographic Edition framework by using the Oracle Solaris Cluster Manager browser interface or the command-line interface (CLI).

The procedures in this guide describe how to perform administrative tasks by using the CLI.

Oracle Solaris Cluster Manager Browser Interface

The Geographic Edition framework supports Oracle Solaris Cluster Manager, a browser interface tool that you can use to monitor Geographic Edition status and perform various administrative tasks on your cluster. For specific information about how to use Oracle Solaris Cluster Manager, see Chapter 13, “Using the Oracle Solaris Cluster Manager Browser Interface” in *Oracle Solaris Cluster 4.3 System Administration Guide* and the Oracle Solaris Cluster Manager online help.

Note - To administer Oracle Solaris Cluster software by using the Oracle Solaris Cluster Manager browser interface, ensure that the root passwords are the same on all nodes of both clusters in the partnership.

You can use Oracle Solaris Cluster Manager to administer the Geographic Edition framework only after the Geographic Edition infrastructure has been enabled by using the `geoadm start` command. For information about enabling and disabling the Geographic Edition infrastructure, see Chapter 3, “Administering the Geographic Edition Framework”.

Oracle Solaris Cluster Manager does not support creating custom heartbeats outside of a partnership. If you want to specify a custom heartbeat in a partnership join operation, use the CLI to run the `geops join-partnership` command.

To access Oracle Solaris Cluster Manager, do the following:

1. From a browser that is running on a machine that is outside the cluster, go to the following URL from any Java-enabled and Javascript-enabled browser:

   `https://`*node*`:8998/scm`
2. Enter a user name and password that is authorized to connect to the node.

   To use Oracle Solaris Cluster Manager to update or manage the cluster, log in as the `root` role or a role that provides `solaris.cluster.modify` and `solaris.cluster.admin` authorizations.

Command-Line Interface Commands

You can use the following commands to administer the Geographic Edition framework. For more information about each command, refer to the *Oracle Solaris Cluster 4.3 Geographic Edition Reference Manual*.

`geoadm`

Enables or disables the Geographic Edition framework on the local cluster and displays the runtime status of the local cluster

`geohb`

Configures and manages the heartbeat mechanism that is provided with the Geographic Edition framework

`geomg`

Configures and manages multigroups of protection groups

`geops`

Creates and manages the partnerships between clusters

`geopg`

Configures and manages protection groups

`geosite`

Configures and manages cluster sites

♦ ♦ ♦ CHAPTER 2

Overview of Administering a Geographic Edition Configuration

This chapter provides the information you need to know before you begin administering the Geographic Edition framework. It also describes the Oracle Solaris Cluster infrastructure that is required by the Geographic Edition framework. This chapter presents common Oracle Solaris Cluster concepts and tasks you need to understand before administering the Geographic Edition framework. It also provides an example configuration that is used throughout this guide to illustrate the common Geographic Edition administration tasks.

This chapter contains the following sections:

Overview of Oracle Solaris Cluster Administration Concepts

You must be an experienced Oracle Solaris Cluster administrator to administer the Geographic Edition framework.

This section describes the following Oracle Solaris Cluster administration topics that you need to understand before you administer the Geographic Edition framework:

Configuring Resources and Resource Groups

You use either Oracle Solaris Cluster commands or the Oracle Solaris Cluster Manager browser interface to create failover and scalable resource groups.

For more information about administering resources and resource groups in Oracle Solaris Cluster software, see the *Oracle Solaris Cluster 4.3 Data Services Planning and Administration Guide*.

Legal Names and Values of Geographic Edition Entities

This section lists the requirements for legal characters for the names and values of Geographic Edition entities.

This appendix contains the following sections:

Legal Names for Geographic Edition Entities

Geographic Edition entity names consist of the following:

- Host names
- Cluster names, which must follow the naming requirements for host names
- Partnership names
- Protection group names
- Custom heartbeat names

All names must comply with the following rules:

- Must start with a letter
- Must not exceed 255 characters
- Can contain the following:
  - Upper and lowercase letters
  - Digits
  - Dashes (-), except as the last character of a host name or cluster name
  - Underscores (_), except in a host name or cluster name

For more information about host name requirements, see RFC 1123 at http://www.rfcs.org/.

Legal Values for Geographic Edition Entities

The Geographic Edition entity values fall into two categories: property values and description values. Both types of values share the following rules:

- 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 a newline or a semicolon

Administering Rights Profiles

This section describes how to administer role-based access control (RBAC) rights in the Geographic Edition framework. For information about rights for the Geographic Edition framework, see "Planning Security" in *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*.

Modifying a User's Rights

Note - The rights profiles for Oracle Solaris Cluster software, including the Geographic Edition framework,are intended to simplify the assignment of management roles and protect against accidental errors. However, a malicious user would be able to abuse rights-based cluster administration privileges to gain wider system privileges. Therefore, user rights should be assigned with care.

When you grant authorization to users other than the `root` role, you must do so on all nodes of both partner clusters. Otherwise, some operations that have a global scope might fail due to insufficient user rights on one or more nodes in the partnership.

To modify the rights for a user, you must be logged in as the `root` role or assume a role that is assigned the System Administrator rights profile.

For example, you can assign the Geo Management rights profile to the user `admin` as follows:

```
# usermod -P "Geo Management" admin
# profiles admin
Geo Management
Basic Solaris User
#
```

For more information about how to modify the rights properties for a user, refer to Chapter 2, "Oracle Solaris Cluster and User Rights" in *Oracle Solaris Cluster 4.3 System Administration Guide*.

Configuring Logical Hostnames

The logical hostname is a special high-availability (HA) resource. The `geoadm start` command configures the logical hostname that corresponds to the cluster name. You must set up the IP address and host maps for the logical hostname before you run this command.

For more information about using the `geoadm start` command, see "Enabling and Disabling the Geographic Edition Framework" on page 32.

Note - If you are using Availability Suite for data replication, a logical hostname is created for each device group to be replicated. For more information, see Chapter 1, "Replicating Data With the Availability Suite Feature of Oracle Solaris" in *Oracle Solaris Cluster Geographic Edition Data Replication Guide for Oracle Solaris Availability Suite*.

The following table lists the Oracle Solaris Cluster and Geographic Edition components that require IP addresses. Add these IP addresses to the following locations:

- All naming services that are being used
- The local `/etc/inet/hosts` file on each cluster node, after you install the Oracle Solaris OS software

TABLE 1 IP Addresses Required by the Geographic Edition Framework

Component	Number of IP Addresses Needed
Oracle Solaris Cluster administrative console	One per subnet
IP Network Multipathing groups	■ Single-adapter groups – one primary IP address. ■ Multiple-adapter groups – one primary IP address plus one test IP address for each adapter in the group.
Cluster nodes	One per node, per subnet
Domain console network interface	One per domain
Console-access device	One
Logical addresses	One per logical host resource, per subnet
Geographic Edition infrastructure hostname	One logical IP address per cluster infrastructure. For example, if you have two clusters in your Geographic Edition infrastructure, you need two IP addresses.
Replication with Availability Suite from Oracle	One dedicated logical IP address on the local cluster for each device group to be replicated. For example, if you have two clusters in your Geographic Edition infrastructure, you need two IP addresses.

For more information about configuring the IP address and host maps during the installation of Oracle Solaris Cluster software, refer to Chapter 2, “Installing Software on Global-Cluster Nodes” in *Oracle Solaris Cluster 4.3 Software Installation Guide*.

Managing Device Groups

A device group is a hardware resource that is managed by the Oracle Solaris Cluster software. A device group is a type of global device that is used by the Oracle Solaris Cluster software to register device resources, such as disks. A device group can include the device resources of disks and Solaris Volume Manager disk sets.

For information about configuring device groups in Oracle Solaris Cluster software, refer to Chapter 5, “Administering Global Devices, Disk-Path Monitoring, and Cluster File Systems” in *Oracle Solaris Cluster 4.3 System Administration Guide*.

The Geographic Edition framework configures Oracle Solaris Cluster device groups to include replication.

For more information about configuring data replication in the Geographic Edition framework, see the Geographic Edition documentation for the data replication product you use.

Geographic Edition Configuration and Administration Tasks

The following table lists administration tasks for your Geographic Edition configuration.

Note - For procedures to create partnerships, heartbeats, protection groups, sites, and multigroups, see *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide* and the Geographic Edition manual for your data replication product.

TABLE 2 Geographic Edition Administration Tasks

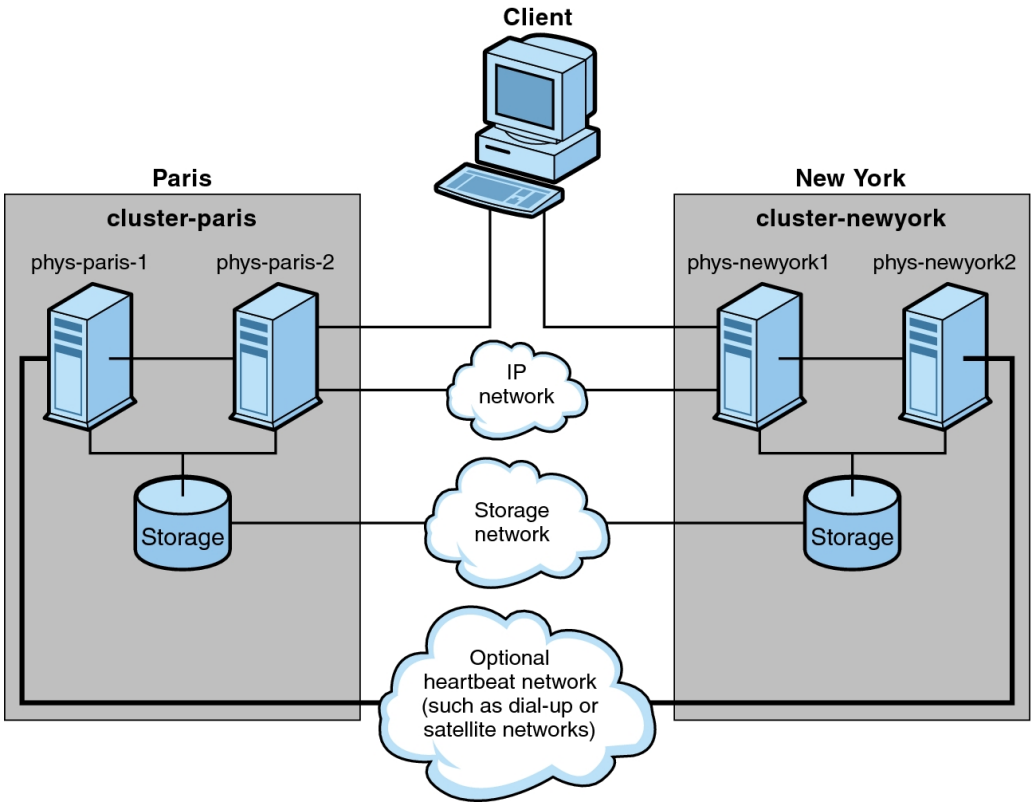
Task	Description and Documentation
Switch over services to the partner cluster. You can switch over individual protection groups or multigroups.	See the procedures to switch over a protection group from primary to secondary in the Geographic Edition manual for your data replication product. You can also use the Oracle Solaris Cluster Manager browser interface.

Task	Description and Documentation
	To switch over multiple protections groups that are configured in a multigroup, see "How to Switch Over a Multigroup" on page 138. You can also use the Oracle Solaris Cluster Manager browser interface.
Take over services from primary to secondary during a disaster. You can take over services for an individual cluster or for a multigroup.	See the procedures to force immediate takeover of services by a secondary cluster in the Geographic Edition manual for your data replication product. You can also use the Oracle Solaris Cluster Manager browser interface. To take over multiple protection groups that are configured in a multigroup, see "How to Take Over a Multigroup" on page 142. You can also use the Oracle Solaris Cluster Manager browser interface.
Recover from a takeover.	■ Perform data recovery and error repair outside of the Geographic Edition infrastructure. See the Geographic Edition documentation for the data replication product you are using. ■ Resynchronize the partner clusters. See "Resynchronizing a Protection Group" on page 84. You can also use the Oracle Solaris Cluster Manager browser interface.
Start an individual protection group or a multigroup.	To start an individual protection group, see "How to Activate a Protection Group" on page 80 or use the Oracle Solaris Cluster Manager browser interface. To start a multigroup, see "How to Start All Protection Groups in a Multigroup" on page 114 or use the Oracle Solaris Cluster Manager browser interface.
Stop an individual protection group or a multigroup.	To stop an individual protection group, see "How to Deactivate a Protection Group" on page 82. You can also use the Oracle Solaris Cluster Manager browser interface. To stop a multigroup, see "How to Stop All Protection Groups in a Multigroup" on page 115. You can also use the Oracle Solaris Cluster Manager browser interface.
Delete a multigroup	See "Deleting a Multigroup" on page 117. You can also use the Oracle Solaris Cluster Manager browser interface.
Delete a site.	See "Deleting a Site" on page 105. You can also use the Oracle Solaris Cluster Manager browser interface.
Delete a protection group or a data replication component.	See "Deleting Protection Groups and Data Replication Components" on page 88. You can also use the Oracle Solaris Cluster Manager browser interface.
Delete a partnership.	See "Leaving or Deleting a Partnership" on page 50. You can also use the Oracle Solaris Cluster Manager browser interface.
Disable the Geographic Edition framework.	See "How to Disable the Geographic Edition Framework" on page 35.
Uninstall the Geographic Edition software.	See *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*.

Example Geographic Edition Cluster Configuration

The following figure describes an Geographic Edition cluster configuration that is used throughout this guide to illustrate the Geographic Edition administration tasks. The primary cluster, `cluster-paris`, contains two nodes, `phys-paris–1` and `phys-paris-2`. The secondary cluster, `cluster-newyork`, also contains two nodes, `phys-newyork-1` and `phys-newyork-2`.

FIGURE 1 Example Cluster Configuration



♦ ♦ ♦ CHAPTER 3

Administering the Geographic Edition Framework

This chapter contains information about enabling your cluster for participation in a partnership. It also contains instructions for disabling the Geographic Edition framework so that your cluster no longer can participate in partnerships.

This chapter contains the following sections:

Geographic Edition Framework Resource Groups

When you enable the Geographic Edition infrastructure, the following Oracle Solaris Cluster resource groups are created:

- `geo-clusterstate` – A scalable resource group that the Geographic Edition framework uses to distinguish between node failover and cluster reboot scenarios. The resource group contains the following resources:
  - `geo-servicetag` - A scalable resource that is started on all nodes of a cluster when Geographic Edition is present. When the Geographic Edition framework is started, this resource checks for the existence of an Oracle Solaris service tag for the running version of Geographic Edition on each node, and creates a service tag if necessary. The service tag indicates that Geographic Edition has been used on the cluster. This service tag is removed from the node when the Geographic Edition packages are removed.
  - `geo-zc-sysevent` - (For zone clusters only) Runs resource methods in the global zone when Geographic Edition is started in a non-global zone. It manages the mechanism which transfers cluster events to subscribers in the zone cluster.

- `geo-infrastructure` – A failover resource group that encapsulates the Geographic Edition infrastructure. The resource group contains the following resources:
  - `geo-cluster-name` – The logical hostname for the Geographic Edition framework. The Geographic Edition framework uses the logical hostname of a cluster for intercluster management communication and heartbeat communication. An entry in the naming services must be the same as the name of the cluster and be available on the namespace of each cluster.
  - `geo-hbmonitor` – Encapsulates the heartbeat processes for the Geographic Edition framework.
  - `geo-failovercontrol` – Encapsulates the Geographic Edition framework itself. The Geographic Edition module uses this resource to load into the common agent container.

These resources are for internal purposes only, so you must not change them.

These internal resources are removed when you disable the Geographic Edition infrastructure.

You can monitor the status of these resources by using the `clresource status` command. For more information about this command, see the `clresource`(1CL) man page.

Enabling and Disabling the Geographic Edition Framework

When you enable the Geographic Edition framework, the cluster is ready to enter a partnership with another enabled cluster. You can use the CLI commands or Oracle Solaris Cluster Manager to create a cluster partnership.

For more information about setting up and installing the Geographic Edition software, see the *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*.

▼ How to Enable the Geographic Edition Framework

This procedure enables the Geographic Edition infrastructure on the local cluster only. Repeat this procedure on all the clusters of your geographically separated cluster.

Before You Begin Ensure that the following conditions are met:

- The cluster is running the Oracle Solaris operating system and the Oracle Solaris Cluster software.
- The Geographic Edition software is installed.

- The cluster has been configured for secure cluster communication by using security certificates, that is, nodes within the same cluster must share the same security certificates. This configuration is done during Oracle Solaris Cluster software installation.
- You must be assigned the Geo Operation rights profile to complete this procedure. For more information, see "Securing Geographic Edition Software" in *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*.

1. **Log in to a cluster node.**

2. **Ensure that the logical hostname, which is the same as the cluster name, is available and defined.**

   ```
   # cluster list
   ```

3. **If the cluster name is not the name you want to use, change the cluster name.**

   If you must change the name of a cluster that is configured in a partnership, do not perform this step. Instead, follow instructions in "Renaming a Cluster That Is in a Partnership" on page 44.

   Follow cluster naming guidelines as described in "Planning Required IP Addresses and Hostnames" in *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*. Cluster names must follow the same requirements as for host names. You cannot rename a cluster that is part of a zone cluster, but you can rename a cluster in a global zone.

   ```
   # cluster rename -c new-cluster-name  old-cluster-name
   ```

   For more information, see the `cluster`(1CL) man page.

   Note - After you have enabled the Geographic Edition infrastructure, you must not change the cluster name while the infrastructure is enabled.

4. **Confirm that the naming service and the local `hosts` files contain a host entry that matches the cluster name.**

   The local host file, `hosts`, is located in the `/etc/inet` directory.

5. **On a node of the cluster, start the Geographic Edition infrastructure.**

   ```
   # geoadm start
   ```

   The `geoadm start` command enables the Geographic Edition infrastructure on the local cluster only. For more information, see the `geoadm`(1M) man page.

6. **Verify that you have enabled the infrastructure and that the Geographic Edition resource groups are online.**

For a list of the Geographic Edition resource groups, see "Geographic Edition Framework Resource Groups" on page 31.

```
# geoadm show
# clresource-group status
# clresource status
```

The output for the `geoadm show` command indicates that the Geographic Edition infrastructure is active from a particular node in the cluster.

The output for the `clresource-group status` and `clresource status` commands displays that the `geo-failovercontrol`, `geo-hbmonitor`, and `geo-cluster-name` resources and the `geo-infrastructure` resource groups are online on one node of the cluster.

For more information, see the `clresourcegroup`(1CL) and `clresource`(1CL) man pages.

Example 1 Enabling the Geographic Edition Infrastructure in a Cluster

This example enables the Geographic Edition framework on the `cluster-paris` cluster.

1. Start the Geographic Edition framework on `cluster-paris`.

   ```
   phys-paris-1# geoadm start
   ```

2. Ensure that the Geographic Edition infrastructure was successfully enabled.

   ```
   phys-paris-1# geoadm show

   --- CLUSTER LEVEL INFORMATION ---
   Oracle Solaris Cluster Geographic Edition is active on cluster-paris from node phys-
   paris-1
   Command execution successful
   phys-paris-1#
   ```

3. Verify the status of the Geographic Edition resource groups and resources.

   ```
   phys-paris-1# clresource-group status
   === Cluster Resource Groups ===

   Group Name            Node Name    Suspended       Status
   ----------            ---------    ---------       ------
   geo-clusterstate      phys-paris-1    No           Online
                         phys-paris-2    No           Online

   geo-infrastructure    phys-paris-1    No           Online
                         phys-paris-2    No           Offline
   ```

```
# clresource status
=== Cluster Resources ===

Resource Name         Node Name      State               Status Message
-------------         ---------      -----               --------------
geo-cluster-name       phys-paris-1   Online              Online - LogicalHostname
 online.
                       phys-paris-2   Offline             Offline

geo-hbmonitor         phys-paris-1   Online              Online - Daemon OK
                      phys-paris-2   Offline             Offline

geo-failovercontrol   phys-paris-1   Online              Online - Service is online
                      phys-paris-2   Offline             Offline

geo-servicetag        phys-paris-1   Online_not_monitored Online_not_monitored
                      phys-paris-1   Offline             Offline
```

Next Steps For information about creating protection groups, see the Geographic Edition data replication guide that corresponds to the type of data replication software you are using.

▼ How to Disable the Geographic Edition Framework

Note - Disabling the Geographic Edition framework removes only the infrastructure resource groups. Resource groups that have been created to support data replication are not removed unless you remove the protection group that the resource groups are supporting by using the `geopg delete` command.

Before You Begin Ensure that all protection groups on the local cluster are offline. For more information, see the following procedures:

- To stop an individual protection group, see "How to Deactivate a Protection Group" on page 82.
- To stop a set of protection groups that are configured in a multigroup, see "How to Stop All Protection Groups in a Multigroup" on page 115.

Caution - If you want to keep the application resource groups online while deactivating a protection group, follow Example 23, “Deactivating a Protection Group While Keeping Application Resource Groups Online,” on page 84 in “How to Deactivate a Protection Group” on page 82.

- You must be assigned the Geo Management rights profile to complete this procedure. For more information, see “Securing Geographic Edition Software” in *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*.

1. **Log in to a cluster node.**

2. **Confirm that all of the protection groups are offline on the local cluster.**

   ```
   phys-paris-1# geoadm status
   ```

   For more information about the `geoadm status` command and its output, see “Monitoring the Runtime Status of the Geographic Edition Framework” on page 119.

3. **Disable the Geographic Edition framework.**

   ```
   phys-paris-1# geoadm stop
   ```

   This command removes the infrastructure resource groups that were created when you enabled the Geographic Edition infrastructure.

   For more information about this command, see the `geoadm`(1M) man page.

4. **Verify that the Geographic Edition framework was disabled and that the Geographic Edition resource groups are no longer displayed.**

   ```
   phys-paris-1# geoadm show
   phys-paris-1# clresource-group status
   ```

   For more information, see the `clresourcegroup`(1CL) man page.

Example 2 Disabling a Cluster

This example disables the `cluster-paris` cluster.

1. Confirm that all protection groups are offline.

   ```
   phys-paris-1# geoadm status

   Cluster: cluster-paris
   ```

```
Partnership "paris-newyork-ps" :OK
   Partner clusters   :cluster-newyork
   Synchronization    :OK
   ICRM Connection    :OK

   Heartbeat "paris-to-newyork" monitoring "cluster-newyork":OK
      Heartbeat plug-in "ping_plugin"   :Inactive
      Heartbeat plug-in "tcp_udp_plugin":OK

Protection group "tcpg"      :OK
    Partnership              :paris-newyork-ps
    Synchronization          :OK

   Cluster cluster-paris    :OK
       Role                 :Primary
       PG activation state  :Deactivated
       Configuration        :OK
       Data replication     :OK
       Resource groups      :OK

   Cluster cluster-newyork  :OK
       Role                 :Secondary
       PG activation state  :Deactivated
       Configuration        :OK
       Data replication     :OK
       Resource groups      :OK
```

2. Disable the Geographic Edition infrastructure.

```
phys-paris-1# geoadm stop
... verifying pre conditions and performing pre remove operations ... done
...removing product infrastructure ... please wait ...
```

3. Confirm that the Geographic Edition infrastructure was successfully disabled.

```
phys-paris-1# geoadm show

--- CLUSTER LEVEL INFORMATION ---
Oracle Solaris Cluster Geographic Edition is not active on cluster-paris

--- LOCAL NODE INFORMATION ---
Node phys-paris-1 does not host active product module.

Command execution successful
```

```
phys-paris-1#
```

4. Verify that Geographic Edition resource groups and resources have been removed.

```
phys-paris-1# clresource-group status
phys-paris-1#
```

Checking the Status of the Geographic Edition Infrastructure

Use the `geoadm show` command to determine whether the Geographic Edition infrastructure is enabled on the local cluster and on which node the infrastructure is active. The Geographic Edition infrastructure is considered active on the node on which the `geo-infrastructure` resource group has a state of `Online`.

You can also use the Oracle Solaris Cluster Manager browser interface to perform this task. Click Partnerships. For Oracle Solaris Cluster Manager log-in instructions, see "How to Access Oracle Solaris Cluster Manager" in *Oracle Solaris Cluster 4.3 System Administration Guide*.

The following example displays information about the `phys-paris-1` node of the `cluster-paris` cluster.

```
phys-paris-1# geoadm show

--- CLUSTER LEVEL INFORMATION ---
Oracle Solaris Cluster Geographic Edition is active on:
node phys-paris-2, cluster cluster-paris

Command execution successful
phys-paris-1#
```

Booting a Cluster

The following events take place when you boot a cluster:

1. After the Oracle Solaris Cluster infrastructure is enabled, the Geographic Edition framework starts automatically. You can verify that the software started successfully by using the `geoadm show` command.
2. The heartbeat framework checks which partners it can reach.

3. Check the current status of the cluster by using the `geoadm status` command. For more information about this command and its output, see "Monitoring the Runtime Status of the Geographic Edition Framework" on page 119.

♦ ♦ ♦ CHAPTER 4

Administering Cluster Partnerships

This chapter describes how to administer partnerships between two Geographic Edition framework-enabled clusters.

This chapter contains the following sections:

Modifying Partnership Properties

This section provides procedures to modify a property of a partnership. Property values must meet the requirements for Geographic Edition entities. See "Legal Values for Geographic Edition Entities" on page 25.

▼ How to Modify Partnership Properties

Note - You can also accomplish this procedure by using the Oracle Solaris Cluster Manager browser interface. Click Partnerships, click the name of the partnership to go to its page, click the Properties tab, and click Edit to access the listed properties for editing. For Oracle Solaris Cluster Manager log-in instructions, see "How to Access Oracle Solaris Cluster Manager" in *Oracle Solaris Cluster 4.3 System Administration Guide*.

Before You Begin You must be assigned the Geo Management rights profile to complete this procedure. For more information, see "Securing Geographic Edition Software" in *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*.

1. **Log in to a cluster node.**

2. **Modify partnership properties.**

```
# geops set-prop -p property-setting [-p…] partnership-name
```

`-p` *property-setting*

Specifies the value of partnership properties with a string of *property=value* pair statements.

Specify a description of the partnership with the `Description` property.

Use the `Notification_emailaddrs` and `Notification_actioncmd` properties to configure heartbeat-loss notification. For more information about configuring heartbeat-loss notification, see "Configuring Heartbeat-Loss Notification" on page 66.

For more information about the properties you can set, see Appendix A, "Standard Geographic Edition Properties".

partnership-name

Specifies the name of the partnership.

For information about the names and values that are supported by the Geographic Edition framework, see "Legal Names and Values of Geographic Edition Entities" on page 24.

For more information about the `geops` command, refer to the `geops`(1M) man page.

3. **Verify that your modification was made correctly.**

```
# geops list
```

Example 3 Modifying the Properties of a Partnership

This example modifies the notification email address for the `cluster-paris` cluster.

```
# geops set-prop -p Notification_emailaddrs=operations@example.com \
paris-newyork-ps
# geops list
```

Adding a New Cluster Node

When you add a new node to a cluster that is in a partnership, you must also make that node an active participant in the Geographic Edition configuration.

▼ How to Add a New Node to a Cluster in a Partnership

Perform all steps from the new node.

1. **Add the new node to the cluster.**

   Follow procedures in Chapter 8, "Administering Cluster Nodes" in *Oracle Solaris Cluster 4.3 System Administration Guide*.

2. **Install Geographic Edition, data replication, and application software on the new node.**

   - To install Geographic Edition software, see *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*.
   - To install data replication and application software, see the appropriate manual for the software that you use.

3. **If the cluster with the new node is the primary for any activated protection groups, remove application resource groups from those protection groups.**

   This step is necessary to avoid application downtime.

   ```
   # geopg remove-resource-group resource-group protection-group
   ```

4. **Deactivate all protection groups that are active on this cluster locally.**

   ```
   # geopg stop -e local protection-group
   ```

5. **Stop the Geographic Edition infrastructure.**

   ```
   # geoadm stop
   ```

6. **Re-enable the Geographic Edition infrastructure.**

   This action re-creates each Geographic Edition resource group and adds all nodes in the cluster, including the new node, to the node list.

```
# geoadm start
```

7. **Reactivate the protection groups that you deactivated in Step 4.**

   ```
   # geopg start -e local protection-group
   ```

8. **Restore any application resource groups that you removed in Step 3.**

   ```
   # geopg add-resource-group resource-group protection-group
   ```

Renaming a Cluster Node

You can rename a node in a Geographic Edition cluster that is in a partnership of an Oracle Solaris Cluster configuration.

If the cluster where you are performing the rename procedure is primary for the protection group and you want to have the application in the protection group online, you can switch the primary group to a secondary during the rename procedure.

For instructions on renaming a node in a Geographic Edition cluster, see "How to Rename a Global Cluster Node" in *Oracle Solaris Cluster 4.3 System Administration Guide*. The procedure includes instructions that pertain to a node in a Geographic Edition configuration.

Renaming a Cluster That Is in a Partnership

When you rename a cluster that is in a partnership, the partnership becomes invalid. You must fully unconfigure the existing partnership and create a new one that uses the new cluster name.

▼ How to Rename a Cluster That Is in a Partnership

This procedure demonstrates how to rename one of the global clusters that is in a partnership. You can rename more than one of the clusters at the same time.

Note - You cannot use this procedure to rename a zone cluster in a partnership.

If the cluster that you rename belongs to more than one partnership, perform each step on all clusters that share a partnership with the cluster that you are renaming, before you proceed to the next step in the procedure.

1. **From one node of the cluster that you are renaming, remove resource groups from each protection group that the cluster belongs to.**

   This task avoids production application downtime.

   ```
   # geopg remove-resource-group application-resource-group protection-group
   ```

2. **From one node of each cluster in a protection group, confirm that application resource groups have been removed.**

   ```
   # geopg list protection-group
   ```

3. **From one node of the cluster that you are renaming, stop each protection group globally.**

   This task stops data replication.

   ```
   # geopg stop protection-group -e global
   ```

4. **From one node of each cluster in a protection group, delete the protection group.**

   ```
   # geopg delete protection-group
   ```

5. **From one node of each cluster in a partnership, leave the partnership.**

   ```
   # geops leave-partnership partnership
   ```

6. **From one node of each cluster, confirm that the protection group and the partnership have been removed.**

   ```
   # geoadm status
   ```

7. **From one node of each cluster, disable the Geographic Edition framework.**

   ```
   # geoadm stop
   ```

8. **From one node of each cluster, confirm that the Geographic Edition framework was disabled.**

   Verify that the `geo-infrastructure`, `geo-clusterstate`, and data replication resource groups are deleted.

   ```
   # clrg list
   ```

```
# geoadm status
```

9. **From one node of the cluster that you are renaming, change the cluster name.**
   Follow cluster naming guidelines as described in "Planning Required IP Addresses and Hostnames" in *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*.

   ```
   # cluster rename -c new-cluster-name
   ```

   Note - The name of the cluster must not include the domain. If a partnership contains clusters that are in different domains, you specify the domain to administrative commands, when necessary, by appending the domain name to the cluster name as *cluster.domain*. Only certain Geographic Edition administrative commands require this fully qualified name when clusters in a partnership are not in the same domain.

10. **Confirm that the cluster name is changed.**

    ```
    # cluster list
    ```

11. **On each node of both clusters, ensure that hostname entries that match the new cluster name are free and are added to the local `/etc/inet/hosts` files.**
    If clusters in the partnership are in different domains, include the domain in the `/etc/hosts` entry for each cluster.

    ```
    # ping new-cluster
        [There should be no response]
    # echo "IP-address  new-cluster" >> /etc/inet/hosts
    ```

12. **From one node of each cluster, start the Geographic Edition framework.**

    ```
    # geoadm start
    ```

    If the Geographic Edition framework fails to start and the failure is not due to problems with the new logical host, restart the common agent container on all nodes by using the `cacaoadm restart` command, then start the Geographic Edition framework.

13. **From one node of each cluster, verify that the Geographic Edition framework is successfully started.**

    ```
    # geoadm status
    ```

14. **From one node of each cluster, add trust between the clusters.**

    ```
    # geops add-trust -c remote-partner-cluster[.domain-name]
    ```

15. **From one node of each cluster, confirm that trust was added successfully.**

Note - Do not specify a domain name to the `verify-trust` subcommand.

```
# geops verify-trust -c remote-partner-cluster
```

16. **Create and join a new partnership between the clusters.**

    a. **From the primary cluster, create the partnership.**

    ```
    # geops create -c remote-partner-cluster[.domain-name] partnership
    ```

    b. **From the secondary cluster, join the partnership.**

    ```
    # geops join-partnership remote-partner-cluster[.domain-name] partnership
    ```

17. **On each cluster, confirm that the new partnership was successfully created and joined.**

    ```
    # geoadm status
    ```

18. **If you did not reboot the nodes of the cluster that you renamed, restart the heartbeats on each node of the renamed cluster.**

    Restarting the heartbeat initiates the heartbeat to read and store the new cluster name.

    ```
    # svcadm disable svc:/system/cluster/gchb_resd:default
    # svcadm enable svc:/system/cluster/gchb_resd:default
    ```

Example 4 Renaming a Cluster in a Partnership

This example renames the cluster `newyork`, in the `paris-newyork-ps` partnership, to `chicago`. The names of the nodes in this cluster are not changed, so `phys-newyork-1` becomes a node in the newly named `chicago` cluster. The `paris-newyork-ps` partnership is first unconfigured. After the cluster is renamed, a new `paris-chicago-ps` partnership is created with the `chicago` cluster as primary and the `paris` cluster as secondary. The two clusters belong to the same domain, so the domain name is not specified in the commands.

```
phys-newyork-1# geopg remove-resource-group app-rg

phys-newyork-1# geopg list examplepg
phys-paris-1# geopg list examplepg

phys-newyork-1# geopg stop examplepg -e global

phys-newyork-1# geopg delete examplepg
phys-paris-1# geopg delete examplepg
```

```
phys-newyork-1# geops leave-partnership paris-newyork-ps
phys-paris-1# geops leave-partnership paris-newyork-ps

phys-newyork-1# geoadm stop
phys-paris-1# geoadm stop

phys-newyork-1# clrg list
phys-newyork-1# geoadm status
phys-paris-1# clrg list
phys-paris-1# geoadm status

phys-newyork-1# cluster rename -c chicago
phys-newyork-1# cluster list

phys-newyork-1# ping chicago
phys-newyork-1# echo "192.168.10.1 chicago" >> /etc/hosts
```

Repeat on each node of the chicago cluster

```
phys-paris-1# ping chicago
phys-paris-1# echo "192.168.20.1 chicago" >> /etc/hosts
```

Repeat on each node of the paris cluster

```
phys-newyork-1# geoadm start
phys-paris-1# geoadm start

phys-newyork-1# geoadm status
phys-paris-1# geoadm status

phys-newyork-1# geops add-trust -c paris
phys-paris-1# geops add-trust -c chicago

phys-newyork-1# geops verify-trust -c paris
phys-paris-1# geops verify-trust -c chicago

phys-newyork-1# geops create -c paris paris-chicago-ps
phys-paris-1# geops join-partnership chicago paris-chicago-ps

phys-newyork-1# geoadm status
phys-paris-1# geoadm status

phys-newyork-1# /etc/init.d/initgchb_resd stop
phys-newyork-1# /etc/init.d/initgchb_resd start
```

Repeat on each node of the chicago cluster

```
phys-paris-1# svcadm disable svc:/system/cluster/gchb_resd:default
phys-paris-1# svcadm enable svc:/system/cluster/gchb_resd:default
```

Repeat on each node of the paris cluster

Next Steps Perform the following tasks by following procedures in the appropriate Geographic Edition data replication guide:

- Create a new protection group and replicate it to partner.
- Add device groups.
- Start globally.
- Add resource groups to the protection group and verify the configuration.

Note - When you create the new protection group, pay close attention to which cluster is the primary and which is the secondary, to ensure that data replication is started in the desired direction.

Removing Trust Between Partner Clusters

This section describe how to remove trust between partner clusters.

▼ How to Remove Trust Between Two Clusters

Perform this procedure to remove trust from between two clusters.

Note - You can also accomplish this procedure by using the Oracle Solaris Cluster Manager browser interface. Click Partnerships, highlight the name of the partnership, and click Remove Partner Trust. For Oracle Solaris Cluster Manager log-in instructions, see "How to Access Oracle Solaris Cluster Manager" in *Oracle Solaris Cluster 4.3 System Administration Guide*.

Before You Begin Ensure that the following conditions are met:

- The cluster on which you want to remove trust is running.
- The cluster name of the partner cluster is known.
- The host information of the partner cluster is defined in the local host file. The local cluster must be able to reach the partner cluster by name.
- You must be assigned the Geo Management rights profile to complete this procedure. For more information, see "Securing Geographic Edition Software" in *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*.

1. **Log in to a cluster node.**

2. **If a partnership is configured between the two clusters, dissolve that partnership.**

   Run the following command on both clusters:

   ```
   # geops leave
   ```

3. **On all nodes of both clusters, remove all keys for the remote cluster from the truststore file on the local node.**

   ```
   # geops remove-trust -c remote-partner-cluster-name
   ```

   Perform this step on all the nodes of the local cluster, and then repeat this step on all nodes of the partner cluster.

   `-c` *remote-partner-cluster-name*

   Specifies the logical hostname of the cluster from which you want to remove the keys. The name for the remote cluster must be identical to the cluster name you specified when adding trust with the `geops add-trust` command. You do not need to specify the fully qualified name if the remote cluster is reachable by partial name.

   When you use this option with the `add-trust` or `remote-trust` subcommand, the option specifies the alias where the public keys on the remote cluster are stored. An alias for certificates on the remote cluster has the following pattern:

   remote-partner-cluster-name`.certificate[0-9]*`

   Only keys that belong to the remote cluster should have their alias match this pattern.

   For more information about the `geops` command, refer to the `geops`(1M) man page.

4. **Repeat the preceding steps on a node of the remote partner cluster.**

Leaving or Deleting a Partnership

This section describes how to leave or delete a partnership. Because the `geops leave-partnership` command destroys the local partnership configuration information, when the last member leaves a partnership, the partnership no longer exists.

You can alternatively use the `geops` command to remove a cluster from a partnership and release all the resources that are associated with the partnership.

How to Leave or Delete a Partnership

Once you complete this procedure for one cluster in the partnership, repeat it for the other partner cluster.

Note - You can also accomplish this procedure by using the Oracle Solaris Cluster Manager browser interface. Click Partnerships, highlight the name of the partnership, and click Leave Partnership. For Oracle Solaris Cluster Manager log-in instructions, see "How to Access Oracle Solaris Cluster Manager" in *Oracle Solaris Cluster 4.3 System Administration Guide*.

Before You Begin Ensure that the following conditions are met:

- The local cluster is a member of the partnership you want to leave.
- This partnership does not contain any protection groups.
- You must be assigned the Geo Management rights profile to complete this procedure. For more information, see "Security Certificates" in *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*.

1. **Log in to a cluster node.**

2. **Verify that the partnership does not contain any protection groups.**

```
# geopg list
```

If the partnership contains protection groups, you can delete them with the `geopg delete` command. For information about deleting protection groups, see "Deleting Protection Groups and Data Replication Components" on page 88.

3. **Remove the partnership on a node of the cluster that is a member of the partnership.**

```
# geops leave-partnership partnership-name
```

partnership-name

Specifies the name of the partnership

Note - The `geops leave-partnership` command deletes the heartbeats configured for the partnership, including custom heartbeats.

For more information, refer to the `geops`(1M) man page.

Example 5 Leaving a Partnership

In this example, the `cluster-paris` cluster leaves the `paris-newyork-ps` partnership.

```
phys-paris-1# geops leave-partnership paris-newyork-ps
```

Example 6 Deleting a Partnership

After the `cluster-paris` cluster leaves the `paris-newyork-ps` partnership, as described in the previous example, the only remaining member of the partnership is the `cluster-newyork` cluster. You can delete the `paris-newyork-ps` partnership by forcing the `cluster-newyork` cluster to leave the partnership.

```
phys-newyork-1# geops leave-partnership paris-newyork-ps
```

Resynchronizing a Partnership

This section provides information about resynchronizing partnership information between partner clusters.

If a cluster that is a member of a partnership fails, when the cluster restarts, it detects whether the partnership parameters have been modified while it was down. You decide which partnership configuration information you want to keep: the information on the cluster that failed or the information on the failover cluster. Then, resynchronize the configuration of the partnership accordingly.

Use the `geoadm status` command or Oracle Solaris Cluster Manager to check whether you need to resynchronize a partnership.

- If the `Configuration` status is `Synchronization Status Error`, you need to synchronize the partnership.
- If the `Local status` is `Partnership Error`, do not resynchronize the partnership. Instead, wait until a heartbeat exchange occurs.

▼ How to Resynchronize a Partnership

Perform this procedure to resynchronize partnership configuration information.

Note - You can also accomplish this procedure by using the Oracle Solaris Cluster Manager browser interface. Click Partnerships, highlight the name of the partnership, and click Update Partnership. For Oracle Solaris Cluster Manager log-in instructions, see "How to Access Oracle Solaris Cluster Manager" in *Oracle Solaris Cluster 4.3 System Administration Guide*.

Before You Begin Ensure that the following conditions are met:

- The local cluster is Geographic Edition enabled.
- The local cluster was an active member of the partnership before failing.
- You must be assigned the Geo Management rights profile to complete this procedure. For more information, see "Securing Geographic Edition Software" in *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*.

Caution - Resynchronizing a partnership overwrites the partnership configuration on the cluster where the command is run with the information from the partner cluster.

1. **Log in to a node on the cluster that needs to be synchronized with the information retrieved from the partner cluster.**

2. **Resynchronize the partnership.**

```
# geops update partnership-name
```

partnership-name

Specifies the name of the partnership

Example 7 Resynchronizing a Partnership

This example resynchronizes the `paris-newyork-ps` partnership.

```
# geops update paris-newyork-ps
```

♦ ♦ ♦ CHAPTER 5

Administering Heartbeats and Heartbeat Plug-Ins

The Geographic Edition framework uses heartbeats over the public network as a way for the individual clusters participating in partnerships to detect cluster failures at partner sites. The heartbeat monitor uses plug-in modules to query the heartbeat status of its partners.

This chapter contains the following sections:

Introduction to Heartbeats and Heartbeat Plug-Ins

A heartbeat in Geographic Edition is a container for a collection of heartbeat plug-ins. A heartbeat has a name and one property that you can tune, `Query_interval`. The `Query_interval` property specifies the delay between heartbeat status requests.

The heartbeat plug-in facilitates the actual physical monitoring activity. The plug-in is defined by a required query command or query library, an optional requester and responder agent, a type, and a `Plugin_properties` string.

The Geographic Edition framework provides a default heartbeat and default heartbeat plug-ins. You can also create a custom heartbeat or a custom heartbeat plug-in.

This section provides the following information:

Overview of the Default Heartbeat and Default Heartbeat Plug-Ins

A default heartbeat that uses the default heartbeat plug-ins is created every time you run `geops create` or `geops join` without specifying a custom heartbeat. The name of the default heartbeat is `hb_`*local-cluster-name~remote-partner-cluster-name*. For more information about the `geops` command, refer to the `geops`(1M) man page.

The Geographic Edition product provides the following default plug-ins:

- `tcp_udp_plugin` - Performs a simple heartbeat check on the cluster logical host IP address. If `tcp_udp_plugin` cannot use UDP port 2084, the plug-in tries to use TCP port 2084.

  Note - The Internet Assigned Numbers Authority (IANA) has officially assigned port number 2084 for use by the Geographic Edition heartbeats.

- `ping_plugin` - Pings the cluster logical hostname on the remote cluster.

Overview of Custom Heartbeats and Custom Heartbeat Plug-Ins

You can create a custom heartbeat plug-in to configure with the default heartbeat or with a new custom heartbeat.

When a heartbeat is created with a custom heartbeat plug-in, the custom heartbeat plug-in is passed the following arguments by the Geographic Edition framework:

query-interval

The value of the `Query_interval` property, which defines the delay in seconds after which a heartbeat status request is declared a failure.

mode

The mode for the plug-in startup, either `Normal` or `Emergency`.

plug-in-property-values

The value of the `Plugin_properties` property that is configured for the heartbeat plug-in, if any.

For more information about the properties you can set, see Appendix A, “Standard Geographic Edition Properties”.

The custom heartbeat plug-in is expected to check the heartbeat on the secondary cluster and return one of the following exit values:

- Zero, if successful – Indicates that the secondary cluster is alive
- Nonzero, on failure – Indicates that the secondary cluster did not respond to the heartbeat check

Guidelines When Creating Custom Heartbeats

Observe the following guidelines when you create custom heartbeats:

- The ability to create custom heartbeats is provided for special circumstances. They require careful configuration. Consult your Oracle specialist for assistance if your system requires the use of custom heartbeats.
- Create the custom heartbeat before you create a partnership. If you create a partnership before you create the custom heartbeat, the default heartbeat that is used by the partnership will prevent the custom heartbeat from being created.
- Ensure that the name of your custom heartbeat is different from the name of the custom heartbeat on the partner cluster.
- Add at least one plug-in to the custom heartbeat, to prevent the partnership from remaining in degraded mode.
- A custom heartbeat prevents the default heartbeat from being used during partnership creation. If you want to use the default heartbeat for your partnership, you must delete the custom heartbeat before you create the partnership with the `geops create` command.

Creating and Deleting Heartbeats

This section describes procedures for creating and deleting a heartbeat.

How to Create a Custom Heartbeat

Follow this procedure to create a custom heartbeat.

Before You Begin

- Review the requirements for custom heartbeats in "Guidelines When Creating Custom Heartbeats" on page 57.
- You must be assigned the Geo Management rights profile to complete this procedure. For more information, see "Securing Geographic Edition Software" in *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*.

1. **Log in to a cluster node.**

2. **Create the heartbeat.**

```
# geohb create -r remote-partner-cluster-name \
[-p property-setting [-p…]] new-heartbeat-name
```

`-r` *remote-partner-cluster-name*

Specifies the name of the remote, secondary partner cluster.

`-p` *property-setting*

Specifies a heartbeat property that is assigned a value by using a *name=statement* pair. Multiple properties might be set at one time by using multiple statements.

For more information about the properties you can set, see Appendix A, "Standard Geographic Edition Properties".

new-heartbeat-name

Specifies an identifier for the new custom heartbeat.

Caution - The name of the custom heartbeat on each cluster in the same partnership must be different. Choose a name that identifies the heartbeat uniquely, such as `paris-to-newyork` on the cluster `cluster-paris` and `newyork-to-paris` on cluster `cluster-newyork`.

When you create a custom heartbeat, you must add at least one plug-in to prevent the partnership from remaining in degraded mode.

For more information about the `geohb` command, refer to the `geohb`(1M) man page.

Example 8 Creating a Custom Heartbeat

This example creates a new heartbeat that is named `paris-to-newyork`.

```
# geohb create -r cluster-newyork paris-to-newyork
```

▼ How to Delete a Heartbeat

Before You Begin You must be assigned the Geo Management rights profile to complete this procedure. For more information, see *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*.

1. **Log in to a cluster node.**
2. **Delete the heartbeat.**

   # geohb delete *heartbeat-name*

   heartbeat-name

   Specifies an identifier for the heartbeat settings.

   For more information about the `geohb` command, refer to the `geohb(1M)` man page.

Example 9 Deleting a Heartbeat

This example deletes a heartbeat that is named `paris-to-newyork`.

```
# geohb delete paris-to-newyork
```

Creating, Modifying, and Deleting Heartbeat Plug-ins

This section contains the following information:

- "How to Create a Custom Heartbeat Plug-in" on page 59
- "How to Modify the Properties of a Heartbeat Plug-in" on page 61
- "How to Delete a Custom Plug-in From a Heartbeat" on page 62

▼ How to Create a Custom Heartbeat Plug-in

This procedure describes how to create a custom heartbeat plug-in and add it to a default or custom heartbeat.

Before You Begin

- If you are creating a custom heartbeat plug-in for a custom heartbeat, ensure that the custom heartbeat already exists.
- You must be assigned the Geo Management rights profile to complete this procedure. For more information, see "Securing Geographic Edition Software" in *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*.

1. **Log in to a cluster node in the primary cluster.**

2. **Create the custom plug-in and add it to a heartbeat.**

```
# geohb add-plugin custom-plug-in-name heartbeat-name [-p property-setting [-p…]]
```

heartbeat-name

Specifies the name of a default or custom heartbeat on the local cluster.

Caution - The presence of a custom heartbeat prevents the default heartbeat from being used during partnership creation. If you want to use the default heartbeat for your partnership, you must delete the custom heartbeat before you create the partnership with the `geops create` command.

custom-plug-in-name

Specifies the name of the custom heartbeat plug-in you are creating.

`-p` *property-setting*

Specifies a heartbeat plug-in property that is assigned a value by using a *name=statement* pair. Multiple properties might be set at one time by using multiple statements.

Specify the path to your custom heartbeat plug-in by using the `Query_cmd` property.

For more information about the properties you can set, see Appendix A, "Standard Geographic Edition Properties".

For more information about the `geohb` command, refer to the `geohb`(1M) man page.

3. **Verify that your changes were made correctly.**

```
# geoadm status
```

4. **Repeat the previous steps on a node of the secondary cluster.**

Example 10 Creating a Custom Heartbeat Plug-in For the Default Heartbeat

The following example adds the custom heartbeat plug-in, `command1`, and adds it to the default heartbeat, `hb_cluster-paris~cluster-newyork`.

```
# geohb add-plugin command1 hb_cluster-paris~cluster-newyork -p Query_cmd=/usr/bin/hb
```

Example 11 Creating a Custom Heartbeat Plug-in For a Custom Heartbeat

This example creates the custom plug-in `command1` and adds it to the `paris-to-newyork` custom heartbeat.

```
# geohb add-plugin command1 paris-to-newyork -p Query_cmd=/usr/bin/hb/
```

How to Modify the Properties of a Heartbeat Plug-in

This procedure describes how to modify heartbeat plug-in properties.

When you modify a heartbeat plug-in property, your changes take effect immediately.

Note - You can also accomplish this procedure by using the Oracle Solaris Cluster Manager browser interface. Click Partnerships, click the partnership name to go to its page, click the heartbeat name to go to its page, click the plug-in name to go to its page, click the Properties tab, and click Edit to access the listed properties for editing. For Oracle Solaris Cluster Manager log-in instructions, see "How to Access Oracle Solaris Cluster Manager" in *Oracle Solaris Cluster 4.3 System Administration Guide*.

Before You Begin You must be assigned the Geo Management rights profile to complete this procedure. For more information, see "Securing Geographic Edition Software" in *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*.

1. **Log in to a cluster node.**
2. **Modify the heartbeat plug-in properties.**

Note - You cannot edit some properties of the default plug-ins, `tcp_upd_plugin` and `ping_plugin`. For more information about the properties you can set, see Appendix A, "Standard Geographic Edition Properties".

 # geohb modify-plugin -p *property-setting* **[-p…]** *plug-in-name heartbeat-name*

heartbeat-name

Specifies an identifier for the heartbeat.

plug-in-name

Specifies the name of the heartbeat plug-in.

`-p` *property-setting*

Specifies a heartbeat plug-in property that is assigned a value by using a *name=statement* pair. Multiple properties might be set at one time by using multiple statements.

For information about the names and values that are supported by the Geographic Edition framework, see "Legal Names and Values of Geographic Edition Entities" on page 24.

For more information about the `geohb` command, refer to the `geohb`(1M) man page.

Example 12 Modifying the Properties of the Heartbeat Plug-in

This example modifies the settings of the default TCP/UDP plug-in, `tcp_udp_plugin`, to use only TCP.

```
# geohb modify-plugin -p Plugin_properties=paris-cluster/TCP/2084 \
tcp_udp_plugin hb_cluster-paris~cluster-newyork
```

How to Delete a Custom Plug-in From a Heartbeat

This procedure describes how to delete a custom heartbeat plug-in.

Caution - Do not delete either of the default heartbeat plug-ins, `tcp_upd_plugin` and `ping_plugin`.

Before You Begin You must be assigned the Geo Management rights profile to complete this procedure. For more information, see "Securing Geographic Edition Software" in *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*.

1. **Log in to a cluster node.**

2. **Remove the plug-in from the heartbeat.**

   # geohb remove-plugin *custom-plug-in-name heartbeat-name*

   custom-plug-in-name

   Specifies the name of the custom heartbeat plug-in to remove.

heartbeat-name

Specifies an identifier for the heartbeat that contains this custom plug-in.

For more information about the `geohb` command, refer to the `geohb(1M)` man page.

Example 13 Deleting a Custom Plug-in From a Heartbeat

This example removes the plug-in `command1` from the heartbeat `paris-to-newyork`.

```
# geohb remove-plugin command1 paris-to-newyork
```

Displaying Heartbeat Configuration Information

This section describes how to display heartbeat configuration information.

▼ How to Display Heartbeat Configuration Information

Note - You can also accomplish this procedure by using the Oracle Solaris Cluster Manager browser interface. Click Partnerships, click the partnership name to go to its page, click the heartbeat name to go to its page, click the Properties tab, and click Edit to access the listed properties for editing. For Oracle Solaris Cluster Manager log-in instructions, see "How to Access Oracle Solaris Cluster Manager" in *Oracle Solaris Cluster 4.3 System Administration Guide*.

Before You Begin You must be assigned the Geo Management rights profile to complete this procedure. For more information, see "Securing Geographic Edition Software" in *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*.

1. **Log in a cluster node.**

2. **Verify that the heartbeat is configured.**

   You can display information for a specific heartbeat or for the whole heartbeat subsystem.

   ```
   # geohb list [heartbeat-name-list]
   ```

heartbeat-name-list

Specifies the names of the heartbeats on the local cluster to list if they are configured. If you do not specify a list of heartbeat names, this command displays a list of all configured heartbeats.

3. **Display the current heartbeat configuration information.**

You can display information for a specific heartbeat or for the whole heartbeat subsystem.

```
# geohb show [heartbeat-name-list]
```

heartbeat-name-list

Specifies the names of the heartbeats on the local cluster for which configuration information should be displayed. If you do not specify a list of heartbeat names, this command displays information about all configured heartbeats.

For more information about the `geohb` command, refer to the `geohb`(1M) man page.

Example 14 Displaying Heartbeat Configuration Information

This example indicates whether the `paris-to-newyork` heartbeat is configured information about the.

```
# geohb list paris-to-newyork
paris-to-newyork
# geohb show paris-to-newyork
    Displays configuration information for paris-to-newyork and its plug-ins
```

Tuning the Heartbeat Properties

You can modify the properties of the default and custom heartbeats by using the `geohb set-prop` command. For more information about this command, refer to the `geohb`(1M) man page.

If you modify the default value of the `Query_interval` property, ensure that the interval is sufficiently long. An interval that is too short causes a timeout and heartbeat-loss event before the logical hostname resource is available. This failover should result in no more than two unanswered heartbeat requests. Setting a default `Query_interval` value of 120 seconds with the default `heartbeat.retries` parameter of 3 enables the peer cluster to be unresponsive for 6 minutes `(120 * 3 )` without having a false failure declared.

The `heartbeat.retries` parameter is specified in the `com.sun.cluster.agent.geocontrol.xml` file. You should not change the `heartbeat.retries` value. If you want to change the default value of the `heartbeat.retries` property, contact an Oracle service representative.

If you adjust the delay setting of the `Query_interval` property, ensure that the following condition is met:

`Query_interval` > *worst-case-logical-host-failover-time* `/ 2`

You must empirically determine the logical-host failover time for the cluster in question. The following must be true to avoid false failures:

`Query_interval` > *worst-case-logical-host-failover-time* `/ 3`

▼ How to Modify the Heartbeat Properties

Note - You can also accomplish this procedure by using the Oracle Solaris Cluster Manager browser interface. Click Partnerships, click the partnership name to go to its page, click the heartbeat name to go to its page, click the Properties tab, and click Edit to access the listed properties for editing. For Oracle Solaris Cluster Manager log-in instructions, see "How to Access Oracle Solaris Cluster Manager" in *Oracle Solaris Cluster 4.3 System Administration Guide*.

Before You Begin You must be assigned the Geo Management rights profile to complete this procedure. For more information, see "Securing Geographic Edition Software" in *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*.

1. **Log in to a cluster node.**

2. **Modify the heartbeat properties.**

   `# geohb set-prop -p` *property-setting* **`[-p…]`** *heartbeat-name*

   `-p` *property-setting*

   Specifies the default properties of the heartbeat.

   A heartbeat property is assigned a value by a *name=statement* pair. Multiple properties can be set at one time by using multiple statements.

   For more information about the properties you can set, see Appendix A, "Standard Geographic Edition Properties".

   For information about the names and values that are supported by the Geographic Edition framework, see "Legal Names and Values of Geographic Edition Entities" on page 24.

heartbeat-name

Specifies an identifier for the heartbeat settings.

For more information about the `geohb` command, refer to the `geohb(1M)` man page.

Example 15 Modifying the Properties of the Default Heartbeat

This example modifies the settings for the default heartbeat between `cluster-paris` and `cluster-newyork`.

```
# geohb set-prop -p Query_interval=60 hb_cluster-paris~cluster-newyork
```

Configuring Heartbeat-Loss Notification

This section contains the following information:

Introduction to Configuring Heartbeat-Loss Notification

You can configure the Geographic Edition framework to send email notification and to run an action script when a heartbeat is lost. You configure heartbeat-loss notification by using the optional `Notification_emailaddrs` and `Notification_actioncmd` properties.

Heartbeat-loss notification occurs if the heartbeat still fails after the interval you configure with the `Query_interval` property of the heartbeat. The heartbeat monitor sends out a heartbeat request to the responder on the logical host every `Query_interval` period. If no response is received within the `Query_interval` period, an internal count is incremented. If the recount reaches the number that is specified in the `heartbeat.retries` property, the heartbeat is deemed to have failed.

For example, you can use the default `Query_interval` value of 120 seconds and the default `heartbeat.retries` of 3. The heartbeat-lost event will be sent a maximum of 10 minutes after the last heartbeat response from the partner cluster.

```
120sec (delay since last query) + 3*120sec (wait for normal response)
+ 120 sec (wait for retry response)
```

Delays can occur between the generation of the heartbeat-loss event and the triggering of the heartbeat-loss notification.

Note - A heartbeat-loss event does not necessarily indicate that the remote cluster has crashed.

The following sections describe how to configure the heartbeat-loss notification properties and how to create a custom action script that the Geographic Edition framework runs after a heartbeat-loss event.

Configuring the Heartbeat-Loss Notification Properties

You can configure heartbeat-loss notification by using two partnership properties, `Notification_emailaddrs` and `Notification_actioncmd`. You specify these properties by using the `geops` command.

You can specify these properties on the default heartbeat during partnership creation. For more information, see "How to Create a Partnership" in *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*. You can also modify these properties by using the procedure that is described in "How to Modify the Heartbeat Properties" on page 65.

If you want to be notified of heartbeat loss by email, set the `Notification_emailaddrs` property. You can specify a list of email addresses, separated by commas. If you want to use email notification, the cluster nodes must be configured as email clients. For more information about configuring mail services, see Chapter 5, "Administering Mail Services" in *Managing sendmail Services in Oracle Solaris 11.3*.

If you want to run a command in response to heartbeat loss, set the `Notification_actioncmd` property.

EXAMPLE 16 Configuring Heartbeat-Loss Notification for an Existing Partnership

This example specifies a notification email address and a custom notification script for the partnership, `paris-newyork-ps`.

```
phys-paris-1# geops set-prop \
```

```
-p Notification_emailaddrs=ops@paris.example.com,ops@newyork.example.com \
-p Notification_actioncmd=/opt/hb_action.sh paris-newyork-ps
```

Creating an Action Shell Script for Heartbeat-Loss

You can create an action shell script that runs when the local cluster detects a heartbeat-loss in the partner cluster. The script runs with root permissions. The file must have root ownership and execution permissions, but the script should not have write permissions.

If you have configured the `Notification_actioncmd` property, the action command runs with arguments that provide information about the event in the following command:

```
# custom-action-command-path -c local-cluster-name -r remote-partner-cluster-name -e 1 \
-n nodename -t time
```

custom-action-command-path

Specifies a path to the action command you have created.

`-c` *local-cluster-name*

Specifies the name of the local cluster.

`-p` *remote-partner-cluster-name*

Specifies the name of the remote partner cluster.

`-e1`

Specifies that `HBLOST=1`, which indicates that a heartbeat-loss event has occurred. The Geographic Edition framework only supports heartbeat-loss notification, so `-e 1` is the only value that can be passed to the action shell script.

`-n`*nodename*

Specifies the name of the cluster node that sent the heartbeat-loss event notification.

`-t` *timestamp*

Specifies the time of the heartbeat-loss event as the number of milliseconds since January 1, 1970, 00:00:00 GMT.

Caution - You can use this script to perform an automatic takeover on the secondary cluster. However, such an automated action is risky. If the heartbeat-loss notification is caused by a total loss of all heartbeat connectivity on both the primary and secondary clusters, such an automated action could lead to a situation where two primary clusters exist.

EXAMPLE 17 How a Notification Action Script Parses the Command-Line Information Provided by the Geographic Edition Framework

This example displays the event information that is provided in the command-line being parsed in a notification action shell script.

```
#!/bin/sh

set -- `getopt abo: $*`
if [ $? != 0]
then
echo $USAGE
exit 2

fi
for i in $*
do

case $i in
-p)      PARTNER_CLUSTER=$1; shift;;
-e)      HB_EVENT=$2; shift;;
-c)      LOCAL_CLUSTER=$3; shift;;
-n)      EVENT_NODE=$4; shift;;
esac
done
```

♦ ♦ ♦ CHAPTER 6

Administering Protection Groups

This chapter contains the following information:

Activating and Deactivating a Protection Group

This section provides the following information:

Guidelines for Activating and Deactivating a Protection Group

When you activate a protection group, the protection group assumes the role that you assigned to it during configuration. When you deactivate a protection group, its application resource groups are also unmanaged.

You can activate or deactivate a protection group in the following ways:

- Globally – Activates or deactivates a protection group on both clusters where the protection group is configured.

- On the primary cluster only – Secondary cluster remains inactive.

  When a protection group is activated on the primary cluster, the application resource groups that are configured for the protection group are also started. The Geographic Edition framework uses the following Oracle Solaris Cluster command on the primary cluster to bring the resource groups online:

  ```
  # clresourcegroup online -eM resource-group-list
  ```
- On the secondary cluster only – Primary cluster remains inactive.

When you activate a protection group, the data replication product that you are using determines the clusters on which data replication can start. The following sections describe additional behaviors when you activate or deactivate a particular data replication product.

Effects of Activating and Deactivating an EMC SRDF Protection Group

Activating and deactivating protection group on a cluster has the following effect on the data replication layer:

- When activated, the data replication configuration of the protection group is validated. During validation, the current local role of a protection group is compared with the configuration of the EMC SRDF device groups.
  - If the EMC SRDF device group is not in a `Failedover` state, the local role of the protection group should match the role of the EMC SRDF device group.
  - If the EMC SRDF device group is in a `Failedover` state, then the local role of the protection group becomes `Secondary` while the role of the EMC SRDF device group remains `Primary`.

  Data replication is started on the data replication device groups that are configured for the protection group, regardless of whether the activation occurs on a primary or secondary cluster. Data is always replicated from the cluster on which the local role of the protection group is `Primary` to the cluster on which the local role of the protection group is `Secondary`.
- Deactivating an EMC SRDF protection group on a cluster has the following effect on the data replication layer:
  - The data replication configuration of the protection group is validated. During validation, the current local role of the protection group is compared with the aggregate device group state. If validation is successful, data replication is stopped.
  - Data replication is stopped on the data replication device groups that are configured for the protection group, regardless of whether the deactivation occurs on a primary or secondary cluster.

Deactivating an EMC SRDF protection group has the following effect on the application layer:

- When a protection group is deactivated on the primary cluster, all of the application resource groups configured for the protection group are stopped and unmanaged.
- When a protection group is deactivated on the secondary cluster, the resource groups on the secondary cluster are not affected. Application resource groups that are configured for the protection group might remain active on the primary cluster, depending on the activation state of the primary cluster.

The EMC SRDF command that is used to stop data replication depends on the RDF state of the EMC SRDF device group.

- If the state is `Synchronized` or `R1Updated`, the `symrdf split` command is run.
- If the state is `Split`, `Suspended`, `Partitioned`, or `Failover`, no command is run because no data is being replicated.

Effects of Activating and Deactivating a Hitachi TrueCopy or Universal Replicator Protection Group

This section provides the following information:

- "Effects of Activating a Hitachi TrueCopy or Universal Replicator Protection Group" on page 73
- "Effects of Deactivating a Hitachi TrueCopy or Universal Replicator Protection Group" on page 76

Effects of Activating a Hitachi TrueCopy or Universal Replicator Protection Group

When you activate a protection group, the protection group assumes the role that you assigned to it during configuration. For more information about activating a protection group, see "Activating a Protection Group" in *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*.

You can activate a protection group in the following ways:

- Globally – Activates a protection group on both clusters where the protection group is configured.
- On the primary cluster only – Secondary cluster remains inactive.
- On the secondary cluster only – Primary cluster remains inactive.

Activating a Hitachi TrueCopy or Universal Replicator protection group on a cluster has the following effect on the data replication layer:

- The data replication configuration of the protection group is validated. During validation, the current local role of a protection group is compared with the aggregate device group state as described in Table 3, “Commands Used to Start Hitachi TrueCopy or Universal Replicator Data Replication,” on page 75. If validation is successful, data replication is started.
- Data replication is started on the data replication device groups that are configured for the protection group, no matter whether the activation occurs on a primary or secondary cluster. Data is always replicated from the cluster on which the local role of the protection group is `Primary` to the cluster on which the local role of the protection group is `Secondary`.

Application handling proceeds only after data replication has been started successfully.

Activating a protection group has the following effect on the application layer:

- When a protection group is activated on the primary cluster, the application resource groups that are configured for the protection group are also started.
- When a protection group is activated on the secondary cluster, the application resource groups are *not* started.

The Hitachi TrueCopy or Universal Replicator command that is used to start data replication depends on the following factors:

- Aggregate device group state
- Local role of the protection group
- Current pair state

If a protection group has a consistency group defined, the fence level is `async` and the device group is in `SMPL` state, then you create the device group with the `paircreate` command when the `geopg start` command is run with the `-f` flag . If a protection group has a consistency group defined, the fence level is not `async` and the device group is in `SMPL` state then you create the device group with the `paircreate` command when you run the `geopg start` command with the `-fg` flags.

On arrays that only support the Hitachi TrueCopy software, the `-fg` fence level option to the `geopg` command is not supported. Thus, on such arrays, the user should only define the `ctgid` on the protection group, if that protection group only has device groups of fence level `async`.

The following table describes the Hitachi TrueCopy or Universal Replicator command that is used to start data replication for each of the possible combinations of factors. In the commands, `dg` is the device group name and `fl` is the fence level that is configured for the device group.

TABLE 3 Commands Used to Start Hitachi TrueCopy or Universal Replicator Data Replication

Aggregate Device Group State	Valid Local Protection Group Role	Hitachi TrueCopy or Universal Replicator Start Command

SMPL	Primary or Secondary	`paircreate -vl -g dg -f fl`
		
		`paircreate -vl -g dg -f fl ctgid`
		
		`paircreate -vr -g dg -f fl`
		
		`paircreate -vr -g dg -f fl ctgid`
		
		All commands require that the `horcmd` process is running on the remote cluster. Device pairs can be started with or without a specified CTGID.

Regular Primary	Primary	If the local state code is 22, 23, 25, 26, 29, 42, 43, 45, 46, or 47, no command is run because data is already being replicated. If the local state code is 24, 44, or 48, then the following command is run: `pairresync -g dg [-l]`. If the local state code is 11, then the following command is run: `paircreate -vl -g dg -f fl`. Both commands require that the `horcmd` process is running on the remote cluster.
Regular Secondary	Secondary	If the local state code is 32, 33, 35, 36, 39, 52, 53, 55, 56, or 57, no command is run because data is already being replicated. If the local state code is 34, 54, or 58, then the following command is run: `pairresync -g dg` If the local state code is 11, the following command is run: `paircreate -vr -g dg -f fl` Both commands require that the `horcmd` process is up on the remote cluster.
Takeover Primary	Primary	If the local state code is 34 or 54, the following command is run: `pairresync -swaps -g`. If the local state code is 11, then the following command is run: `paircreate -vl -g dg -f fl`. The `paircreate` command requires that the `horcmd` process is running on the remote cluster.
Takeover Secondary	Secondary	If the local state code is 24, 44, 25, or 45, the following command is run: `pairresync -swapp -g dg`. If the local state code is 11, the following command is run: `paircreate -vr -g dg -f fl`. Both commands require that the `horcmd` process is running on the remote cluster.

Effects of Deactivating a Hitachi TrueCopy or Universal Replicator Protection Group

You can deactivate a protection group on the following levels:

- Globally – Deactivates a protection group on both clusters where the protection group is configured
- On the primary cluster only – Secondary cluster remains active
- On the secondary cluster only – Primary cluster remains active

Deactivating a Hitachi TrueCopy or Universal Replicator protection group on a cluster has the following effect on the data replication layer:

- The data replication configuration of the protection group is validated. During validation, the current local role of the protection group is compared with the aggregate device group state as described in Table 4, “Commands Used to Stop Hitachi TrueCopy or Universal Replicator Data Replication,” on page 77. If validation is successful, data replication is stopped.
- Data replication is stopped on the data replication device groups that are configured for the protection group, whether the deactivation occurs on a primary or secondary cluster.

Deactivating a protection group has the following effect on the application layer:

- When a protection group is deactivated on the primary cluster, all of the application resource groups that are configured for the protection group are stopped and unmanaged.
- When a protection group is deactivated on the secondary cluster, the resource groups on the secondary cluster are not affected. Application resource groups that are configured for the protection group might remain active on the primary cluster, depending on the activation state of the primary cluster.

The Hitachi TrueCopy or Universal Replicator command that is used to stop data replication depends on the following factors:

- Aggregate device group state
- Local role of the protection group
- Current pair state

The following table describes the Hitachi TrueCopy or Universal Replicator command used to stop data replication for each of the possible combinations of factors. In the commands, `dg` is the device group name.

TABLE 4 Commands Used to Stop Hitachi TrueCopy or Universal Replicator Data Replication

Aggregate Device Group State	Valid Local Protection Group Role	Hitachi TrueCopy or Universal Replicator Stop Command
`SMPL`	`Primary` or `Secondary`	No command is run because no data is being replicated.
`Regular Primary`	`Primary`	If the local state code is 22, 23, 26, 29, 42, 43, 46, or 47, then the following command is run: `pairsplit -g dg [-l]`. If the local state code is 11, 24, 25, 44, 45, or 48, then no command is run because no data is being replicated.
`Regular Secondary`	`Secondary`	If the local state code is 32, 33, 35, 36, 39, 52, 53, 55, 56, or 57, the following command is run: `pairsplit -g dg`. If the local state code is 33 or 53 and the remote state is `PSUE`, no command is run to stop replication. If the local state code is 11, 34, 54, or 58, then no command is run because no data is being replicated.
`Takeover Primary`	`Primary`	No command is run because no data is being replicated.
`Takeover Secondary`	`Secondary`	No command is run because no data is being replicated.

Replication Components Started From Primary Site

Some replication mechanisms can only start or stop replication from the primary site. As such, for data replication to start or stop, activate or deactivate a protection group in one of the following ways:

- Locally from the primary cluster
- Globally from either the primary or the standby cluster

So, if you attempt to activate the protection group locally from the standby cluster, data replication does not start. However, if you activate the protection group globally from the standby cluster, data replication does start.

This applies to the following replication mechanisms:

- Oracle Data Guard
- ZFS Storage Appliance replication
- Availability Suite

Activating a Protection Group

This section provides the following guidelines and instructions to activate a protection group.

Creating a Protection Group Without Starting Replication

If you specify the `-n` option in the `geopg start` command, the command prevents the start of replication when the protection group starts.

If you omit the `-n` option from the `geopg start` command, the replication subsystem starts at the same time as the protection group. The following data replication products have additional behaviors:

- **Availability Suite** – The data replication subsystem starts at the same time as the protection group and the `geopg start` command performs the following operations on each device group in the protection group:
  - Verifies that the role configured for the replication resource is the same as the role of the protection group on the local cluster.
  - Verifies that the role of the volume sets associated with the device group is the same as the role of the protection group on the local cluster.
  - If the role of the protection group on the local cluster is `Secondary`, unmounts the local volumes defined in all volume sets associated with the device group.
  - If the role of the protection group on the local cluster is `Primary`, enables the autosynchronization feature of the Availability Suite remote mirror feature. Also, resynchronizes the volume sets associated with the device group.
- **MySQL** – The `geopg start` command performs the following actions if the role of the protection group is secondary on the local cluster:
  - Starts the MySQL slave threads
  - Prevents modification by an unprivileged database user if this option is configured
  - Prepares the `my.cnf` file to start the database with modifications prevented for an unprivileged database user if this option is configured
- **Oracle Data Guard** – The `geopg start` command performs the following operations on each Oracle Data Guard Broker configuration in the protection group:
  - Verifies that the resource group that is named in the `local_oracle_svr_rg_name` property contains a resource of type `SUNW.scalable_rac_server_proxy` for a scalable resource group or a resource of type `SUNW.oracle_server` for a failover resource group.
  - Verifies that the Oracle Data Guard `dgmgrl` command can connect using the values that are given for `sysdba_username`, `sysdba_password`, and `local_db_service_name`. Or if

the `sysdba_username` and `sysdba_password` properties are null, verifies that the `dgmgrl` command can connect using the Oracle wallet connection format, `dgmgrl /@`*local-db-service-name*.

- Verifies that the role configured for the replication resource is the same as the role of the protection group on the local cluster.
- Verifies that the Oracle Data Guard Broker configuration details match those that are held by Geographic Edition. The details to check include which cluster is primary, the configuration name, the database mode (for both the primary and standby clusters), the replication mode, the standby type, that `FAST_START FAILOVER` is disabled, and that `BystandersFollowRoleChange` is equal to `NONE`.

Bringing Resource Groups Online

The `geopg start` command uses the `clresourcegroup online -eM` *resource-group-list* command to bring resource groups and resources online. For more information about using this command, see the `clresourcegroup`(1CL) man page.

If the role of the protection group is `Primary` on the local cluster, the `geopg start` command brings the application resource groups in the protection group online on the local cluster. For Oracle Data Guard, this includes the shadow Oracle database server resource groups.

For the following data replication products, the `geopg start` command performs additional operations:

Availability Suite

- If the application resource group is a failover type resource group that shares affinities with a device group in the same protection group, the command adds strong positive affinities and failover delegation between the application resource group and the lightweight resource group.

  The application resource group must not have strong, positive affinities with failover delegation. Otherwise, the attempt to add strong positive affinities with failover delegation with the lightweight resource group will fail.
- The command creates strong dependencies between the HAStoragePlus resource in the application resource group and the HAStoragePlus resource in the lightweight resource group for this device group.

MySQL

- Prepares the `my.cnf` file to start the database without the slave threads
- Brings online the application resource groups in the protection group on the local cluster

▼ How to Activate a Protection Group

This procedure activates, or starts, the protection group on the primary and secondary clusters, depending on the scope of the command. When you activate a protection group on the primary cluster, its application resource groups are also brought online.

Note - You can also accomplish this procedure by using the Oracle Solaris Cluster Manager browser interface. Click Partnerships, click the partnership name to go to its page, highlight the protection group name, and click Start Protection Group. For Oracle Solaris Cluster Manager log-in instructions, see "How to Access Oracle Solaris Cluster Manager" in *Oracle Solaris Cluster 4.3 System Administration Guide*.

Before You Begin

- Review the guidelines and special instructions in "Activating a Protection Group" on page 77.
- If you use a role with Geo Management rights, ensure that the `/var/cluster/geo` access control lists (ACLs) are correct on each node of both partner clusters. If necessary, assume the `root` role on the cluster node and set the correct ACLs.

  ```
  # chmod A+user:username:rwx:allow /var/cluster/geo
  ```

  The `/var/cluster/geo` directory must have the correct ACL applied for compatibility between the Geo Management rights profile and data replication software.

  For more information, see "Securing Geographic Edition Software" in *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*.

1. **Assume the `root` role or a role that is assigned the Geo Management rights profile.**

2. **Activate the protection group.**

   `phys-`*node-n*`#` **`geopg start -e`** *scope* **`[-n]`** *protection-group-name*

   -e *scope*

   Specifies the scope of the command.

   If the scope is `local`, then the command operates on the local cluster only. If the scope is `global`, the command operates on both clusters that deploy the protection group.

   Note - The property values `global` and `local` are *not* case sensitive.

   -n

   Prevents the start of replication at protection group startup.

If you omit this option, the replication subsystem starts at the same time as the protection group. See “Activating a Protection Group” on page 77 for information about data replication products with additional behaviors when replication is not started.

protection-group-name

Specifies the name of the protection group.

Example 18 Globally Activating a Protection Group

This example globally activates a protection group.

```
phys-paris-1# geopg start -e global sales-pg
```

Example 19 Locally Activating a Protection Group

This example activates a protection group on a local cluster only. This local cluster might be a primary cluster or a standby cluster, depending on the role of the cluster.

```
phys-paris-1 geopg start -e local sales-pg
```

Troubleshooting If the command fails, the `Configuration` status might be set to `Error` depending on the cause of the failure. The protection group remains deactivated but data replication might be started and some resource groups might be brought online. Run the `geoadm status` command or Oracle Solaris Cluster Manager to obtain the status of your system.

If the `Configuration` status is set to `Error`, revalidate the protection group by using the procedures that are described in “How to Validate a Protection Group” in *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*.

Deactivating a Protection Group

This section provides guidelines and instructions to deactivate a protection group.

The result of deactivating a protection group on the primary or standby cluster depends on the type of data replication that you are using.

Availability Suite Data replication can be stopped only from the primary cluster. So, when you deactivate a protection group on the secondary cluster, this deactivate command does not stop data replication.

If the role of the protection group is `Primary` on the local cluster, the `geopg stop` command disables the autosynchronization of each device group and places the volume sets into logging mode.

EMC SRDF

- The data replication configuration of the protection group is validated. During validation, the current local role of the protection group is compared with the aggregate device group state. If validation is successful, data replication is stopped.
- Data replication is stopped on the data replication device groups that are configured for the protection group, whether the deactivation occurs on a primary or secondary cluster.

Oracle Data Guard

You can stop the Oracle Data Guard configuration from the primary or the standby cluster when the configuration is enabled because the Oracle Data Guard command-line interface (`dgmgrl`) on both clusters still accepts commands.

If the role of the protection group is `Primary` on the local cluster, the `geopg stop` command disables the Oracle Data Guard Broker configuration.

▼ How to Deactivate a Protection Group

This procedure deactivates the protection group on all nodes of the primary and secondary clusters, depending on the scope of the command. When you deactivate a protection group, its application resource groups are also unmanaged.

Note - You can also accomplish this procedure by using the Oracle Solaris Cluster Manager browser interface. Click Partnerships, click the partnership name to go to its page, highlight the protection group name, and click Stop Protection Group. For Oracle Solaris Cluster Manager log-in instructions, see "How to Access Oracle Solaris Cluster Manager" in *Oracle Solaris Cluster 4.3 System Administration Guide*.

You can deactivate, or stop, a protection group in the following ways:

- Globally, meaning you deactivate a protection group on both the primary and the standby cluster where the protection group is configured
- On the primary cluster only
- On the standby cluster only

Before You Begin

- Review the guidelines in "Deactivating a Protection Group" on page 81.
- If you use a role with Geo Management rights, ensure that the `/var/cluster/geo` access control lists (ACLs) are correct on each node of both partner clusters. If necessary, assume the `root` role on the cluster node and set the correct ACLs.

```
# chmod A+user:username:rwx:allow /var/cluster/geo
```

The `/var/cluster/geo` directory must have the correct ACL applied for compatibility between the Geo Management rights profile and data replication software.

For more information, see "Securing Geographic Edition Software" in *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*.

1. **Assume the `root` role or assume a role that is assigned the Geo Management rights profile.**

2. **Deactivate the protection group.**

   When you deactivate a protection group on the primary cluster, its application resource groups are also taken offline.

   ```
   # geopg stop -e scope [-D] protection-group-name
   ```

   -e *scope*

   Specifies the scope of the command.

   If the scope is `Local`, then the command operates on the local cluster only. If the scope is `Global`, the command operates on both clusters where the protection group is deployed.

   Note - The property values, such as `global` and `local`, are *not* case sensitive.

   `-D`

   Specifies that only replication should be stopped and the protection group should be online.

   If you omit this option, the replication subsystem and the protection group are both stopped. If the role of the protection group on the local cluster is `Primary`, omitting the `-D` option also results in taking the application resource groups offline and putting them in an unmanaged state.

   protection-group-name

   Specifies the name of the protection group.

Example 20 Deactivating a Protection Group on All Clusters

This example deactivates a protection group on all clusters.

```
# geopg stop -e global sales-pg
```

Example 21 Deactivating a Protection Group on the Local Cluster

This example deactivates a protection group on the local cluster.

```
# geopg stop -e local sales-pg
```

Example 22 Stopping Remote Replication While Leaving the Protection Group Online

This example stops replication on the local cluster only.

```
# geopg stop -e local -D sales-pg
```

If you decide later to deactivate both the protection group and its underlying replication subsystem, you can rerun the command without the `-D` option:

```
# geopg stop -e local sales-pg
```

Example 23 Deactivating a Protection Group While Keeping Application Resource Groups Online

This example keeps online two application resource groups, `apprg1` and `apprg2`, while deactivating their protection group, `sales-pg`, on both clusters.

1. Remove the application resource groups from the protection group.

   ```
   # geopg remove-resource-group apprg1,apprg2 sales-pg
   ```

2. Deactivate the protection group.

   ```
   # geopg stop -e global sales-pg
   ```

Troubleshooting If the `geopg stop` command fails, run the `geoadm status` command or Oracle Solaris Cluster Manager to obtain the status of each data replication component. For example, the configuration status might be set to `Error` depending on the cause of the failure. The protection group might remain activated even though some resource groups might be unmanaged. The protection group might be deactivated with replication running.

If the configuration status is set to `Error`, revalidate the protection group. See "Validating a Protection Group" in *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*.

Resynchronizing a Protection Group

This section provides information about resynchronizing protection group information between partner clusters.

Partner clusters that become disconnected during a disaster situation might force the administrator to perform a takeover for a protection group that the partners share. When both clusters are brought online again, both partner clusters might report as the primary of the protection group. You must resynchronize the configuration information of the local protection group with the configuration information that is retrieved from the partner cluster.

You decide which protection group configuration information you want to keep: the information on the cluster that failed or the information on the partner cluster. Then, resynchronize the configuration of the protection group accordingly.

You can resynchronize the configuration information of the local protection group with the configuration information retrieved from the partner cluster. You need to resynchronize a protection group when its `Synchronization` status in the output of the `geoadm status` command or in Oracle Solaris Cluster Manager is `Error`. For example, you might need to resynchronize protection groups after booting the cluster. For more information, see "Booting a Cluster" in *Oracle Solaris Cluster 4.3 Geographic Edition System Administration Guide*.

Resynchronizing a protection group updates only entities that are related to Geographic Edition. To update Oracle Solaris Cluster resource groups, resource types, and resources, use the `cluster export -t rg,rt,rs` command to generate an XML cluster configuration file, modify the XML file for the expected configuration on the secondary cluster, and run the `clresource create` command with the `-a` option to apply the configuration updates. For more information, see "How to Configure Oracle Solaris Cluster Software on All Nodes (XML)" in *Oracle Solaris Cluster 4.3 Software Installation Guide* and the `cluster`(1CL) and `clresource`(1CL) man pages.

▼ How to Resynchronize a Protection Group

Note - You can also accomplish this procedure by using the Oracle Solaris Cluster Manager browser interface. Click Partnerships, click the partnership name to go to its page, highlight the protection group name, and click Update Protection Group. For Oracle Solaris Cluster Manager log-in instructions, see "How to Access Oracle Solaris Cluster Manager" in *Oracle Solaris Cluster 4.3 System Administration Guide*.

Before You Begin

- The protection group is be deactivated on the cluster where you run the `geopg update` command.
- You must be assigned the Geo Management rights profile to complete this procedure. For more information, see "Securing Geographic Edition Software" in *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*.

1. **Log in to a cluster node.**

2. **Resynchronize the protection group.**

```
# geopg update protection-group-name
```

protection-group-name

Specifies the name of the protection group

Modifying Protection Groups and Data Replication Components

This section contains the following procedures:

▼ How to Modify a Protection Group

Note - You can also accomplish this procedure by using the Oracle Solaris Cluster Manager browser interface. Click Partnerships, click the partnership name to go to its page, click the protection group name to go to its page, click the Properties tab, then click Edit to access the list of editable properties. For Oracle Solaris Cluster Manager log-in instructions, see "How to Access Oracle Solaris Cluster Manager" in *Oracle Solaris Cluster 4.3 System Administration Guide*.

Before You Begin

- Ensure that the protection group you want to modify exists locally.
- You must be assigned the Geo Management rights profile to complete this procedure. For more information, see "Securing Geographic Edition Software" in *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*.

1. **Log in to one of the cluster nodes.**

2. **Modify the configuration of the protection group.**

This command modifies the properties of a protection group on all nodes of the local cluster. If the partner cluster contains a protection group of the same name, this command also propagates the new configuration information to the partner cluster.

```
# geopg set-prop -p property-setting [-p...] protection-group
```

`-p` *property-setting*

Specifies the properties of the protection group.

For more information about the protection group properties you can set, see Appendix A, "Standard Geographic Edition Properties".

protection-group

Specifies the name of the protection group.

For information about the names and values that are supported by the Geographic Edition framework, see "Legal Names and Values of Geographic Edition Entities" on page 24.

For more information about the `geopg` command, see the `geopg`(1M) man page.

▼ How to Modify a Data Replication Component

Use this procedure to modify a property of a data replication component.

Note - You can also accomplish this procedure by using the Oracle Solaris Cluster Manager browser interface. Click Partnerships, click the partnership name to go to its page, click the protection group name to go to its page, click the data replication component name to go to its page, click the Properties tab, and click Edit to access the list of editable properties. For Oracle Solaris Cluster Manager log-in instructions, see "How to Access Oracle Solaris Cluster Manager" in *Oracle Solaris Cluster 4.3 System Administration Guide*.

Before You Begin

- If you use a role with Geo Management rights, ensure that the `/var/cluster/geo` access control lists (ACLs) are correct on each node of both partner clusters. If necessary, assume the `root` role on the cluster node and set the correct ACLs.

  ```
  # chmod A+user:username:rwx:allow /var/cluster/geo
  ```

  The `/var/cluster/geo` directory must have the correct ACL applied for compatibility between the Geo Management rights profile and data replication software.

  For more information, see "Securing Geographic Edition Software" in *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*.

1. **Assume the `root` role or assume a role that is assigned the Geo Management rights profile.**

2. **Modify the data replication component.**

The following command modifies the properties of a data replication component in a protection group on the local cluster. The command then propagates the new configuration to the partner cluster if the partner cluster contains a protection group with the same name.

```
# geopg modify-replication-component -p  property [-p…] replication-component protection-group
```

`-p` *property*

Specifies the properties of the data replication component.

replication-component

Specifies the name of the data replication component.

protection-group

Specifies the name of the protection group that contains the data replication component.

Example 24 Modifying the Properties of an Oracle ZFS Storage Appliance Remote Replication Component

The following example modifies the `Timeout` property of the remote replication component `trancos` which is part of the appliance protection group `zfssa-p`.

```
# geopg modify-replication-component -p Timeout=215 trancos zfssa-pg
```

Deleting Protection Groups and Data Replication Components

This section provides the following procedures:

- "How to Delete a Protection Group" on page 88
- "How to Remove a Data Replication Component From a Protection Group" on page 90

▼ How to Delete a Protection Group

Use the following procedure to delete a protection group from the local cluster.

Note - You can also accomplish this procedure by using the Oracle Solaris Cluster Manager browser interface. Click Partnerships, click the partnership name to go to its page, highlight the protection group name, and click Delete. For Oracle Solaris Cluster Manager log-in instructions, see "How to Access Oracle Solaris Cluster Manager" in *Oracle Solaris Cluster 4.3 System Administration Guide*.

If you want to delete the protection group everywhere, you must run the `geopg delete` command on each cluster where the protection group exists.

To keep the application resource groups online while deleting the protection group, you must remove the application resource groups from the protection group. See Example 26, "Deleting an EMC SRDF Protection Group While Keeping Application Resource Groups Online," on page 90 for examples of this procedure.

Before You Begin Ensure that the following conditions are met:

- The protection group you want to delete exists locally.
- The protection group is offline on all clusters from which you want to delete it.
- You must be assigned the Geo Management rights profile to complete this procedure. For more information, see "Securing Geographic Edition Software" in *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*.

1. **Log in to one of the nodes on the primary cluster, `cluster-paris`.**

2. **Delete the protection group.**

   The following command deletes the configuration of the protection group from the local cluster. The command also removes the replication resource group for each device group in the protection group.

   ```
   # geopg delete protection-group-name
   ```

   protection-group-name

   Specifies the name of the protection group

3. **To also delete the protection group on the secondary cluster, repeat Step 1 and Step 2 from a node of the secondary cluster.**

Example 25 Deleting a Protection Group

This example deletes a protection group `srdfpg` from both partner clusters. The protection group is offline on both partner clusters.

In this example, `phys-paris-1` is a node of the primary cluster and `phys-newyork-1` is a node of the secondary cluster.

```
# rlogin phys-paris-1 -l root
phys-paris-1# geopg delete srdfpg
# rlogin phys-newyork-1 -l root
phys-newyork-1# geopg delete srdfpg
```

Example 26 Deleting an EMC SRDF Protection Group While Keeping Application Resource Groups Online

This example keeps online two application resource groups, `apprg1` and `apprg2`, while deleting their protection group, `srdfpg`, from both partner clusters. The following commands remove the application resource groups from the protection group, then delete the protection group.

```
phys-paris-1# geopg remove-resource-group apprg1,apprg2 srdfpg
phys-paris-1# geopg stop -e global srdfpg
phys-paris-1# geopg delete srdfpg
phys-newyork-1# geopg delete srdfpg
```

Troubleshooting If the deletion is unsuccessful, the configuration status is set to `Error`. Fix the cause of the error and rerun the `geopg delete` command.

For Oracle Data Guard, unlike other data replication modules, the Oracle database-server resource group is not added to the protection group. Instead, a shadow Oracle database-server resource group is added to represent this resource group. You can add and remove the shadow Oracle database-server resource group to and from the protection group at any time without affecting the Oracle Data Guard data replication. Consequently, the application resource groups that are shown in this example would have no data to replicate in an Oracle Data Guard protection group. Application resource groups that might meet this criteria can be scalable web servers, where their data is static or held on some remote storage that is not controlled by this cluster.

▼ How to Remove a Data Replication Component From a Protection Group

Use the following procedure to remove a data replication component from a protection group.

Note - You can also accomplish this procedure by using the Oracle Solaris Cluster Manager browser interface. Click Partnerships, click the partnership name to go to its page, click the protection group name to go to its page, highlight the data replication component name, and click Remove. For Oracle Solaris Cluster Manager log-in instructions, see "How to Access Oracle Solaris Cluster Manager" in *Oracle Solaris Cluster 4.3 System Administration Guide*.

Before You Begin Ensure that the following conditions are met:

- The protection group is defined on the local cluster.
- The protection group is offline on the local cluster and the partner cluster, if the partner cluster can be reached.
- The data replication component is managed by the protection group.
- If you use a role with Geo Management rights, ensure that the `/var/cluster/geo` access control lists (ACLs) are correct on each node of both partner clusters. If necessary, assume the `root` role on the cluster node and set the correct ACLs.

  ```
  # chmod A+user:username:rwx:allow /var/cluster/geo
  ```

  The `/var/cluster/geo` directory must have the correct ACL applied for compatibility between the Geo Management rights profile and data replication software.

  For more information, see "Securing Geographic Edition Software" in *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*.

1. **Assume the `root` role or assume a role that is assigned the Geo Management rights profile.**

2. **Remove the data replication component.**

   ```
   # geopg remove-replication-component replication-component protection-group
   ```

   replication-component

   Specifies the name of the data replication component.

   protection-group

   Specifies the name of the protection group.

Example 27 Removing a Remote Replication Component From an Oracle ZFS Storage Appliance Protection Group

In the following example, the remote replication component `trancos` is removed from the appliance protection group, `zfssa-pg`.

```
# geopg remove-replication-component trancos zfssa-pg
```

♦ ♦ ♦ CHAPTER 7

Administering Sites

You administer a site from a cluster that is a controller of the site. Changes to the site are propagated by the issuing controller cluster to all other clusters in the site.

This chapter contains the following information about administering a site:

Adding a Cluster to a Site

This section contains the following procedures to add a cluster to a site.

▼ How to Add a Cluster to a Site

Perform this procedure to add one or more clusters to an existing site. You can add a global cluster or a zone cluster. For procedures to create a site, see "How to Create a Site" in *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*.

Note - You can also use the Oracle Solaris Cluster Manager browser interface to perform this task. Click Sites, click the site name to go to its page, and in the Clusters section click Add. For Oracle Solaris Cluster Manager log-in instructions, see "How to Access Oracle Solaris Cluster Manager" in *Oracle Solaris Cluster 4.3 System Administration Guide*.

Before You Begin If you use a role with Geo Management rights, ensure that the `/var/cluster/geo` access control lists (ACLs) are correct on each node of both partner clusters. If necessary, assume the `root` role on the cluster node and set the correct ACLs.

```
# chmod A+user:username:rwx:allow /var/cluster/geo
```

The `/var/cluster/geo` directory must have the correct ACL applied for compatibility between the Geo Management rights profile and data replication software.

For more information, see "Securing Geographic Edition Software" in *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*.

1. **From a node of a site controller cluster, assume the `root` role or assume a role that is assigned the Geo Management rights profile.**

2. **Ensure that all nodes of the site controller cluster are online.**

```
phys-schost-1# cluster status -t node
=== Cluster Nodes ===

--- Node Status ---

Node Name                                       Status
---------                                       ------
phys-schost-2                                   Online
phys-schost-1                                   Online
```

If any node is offline, wait until the node is brought back up before you issue the site invitation. The addition of a new cluster to a site will fail if any node in the issuing cluster is not online.

3. **Invite the cluster to join the site as either a controller or a member.**

site-controller# **`geosite add-member {-c | -m}`** *cluster site*

`-c`

Specifies the site controller role.

`-m`

Specifies the site member role.

cluster

The name of the cluster to add to the site. To add multiple clusters, separate the cluster names with a comma (,). You can use both the `-c` and `-m` options in the same `geosite add-member` command.

site

The name of the site that you are adding the cluster to.

4. **Verify the invitation.**

Command output is similar to the following example, where the new cluster is added as a site controller.

site-controller## **geosite status**

```
Site : site

 Controller "issuing-cluster"
 Configuration               : OK

 Controller "added-cluster"
 Configuration               : OK
 Synchronization             : Unknown
…
```

Troubleshooting If the `geosite add-member` command fails with a timeout, update the site's `Timeout` value to a larger number and retry the command.

Next Steps To complete the addition of the cluster to the site, the invited cluster must join the site. Go to "How to Accept an Invitation to Join a Site" on page 95.

▼ How to Accept an Invitation to Join a Site

Perform this procedure to complete the addition of a cluster to a site.

Note - You can also use the Oracle Solaris Cluster Manager browser interface to perform this task. Click Sites, highlight the site name, and click Join. For Oracle Solaris Cluster Manager log-in instructions, see "How to Access Oracle Solaris Cluster Manager" in *Oracle Solaris Cluster 4.3 System Administration Guide*.

Before You Begin

- Ensure that an invitation to join a site has been issued for the cluster. See "How to Add a Cluster to a Site" on page 93.
- Ensure that all nodes of the cluster that issued the invitation are online.
- Ensure that the common agent container is started on all nodes of both the cluster that issued the invitation and the invited cluster.

1. **From a node of the site cluster that issued the invitation, ensure that all nodes are online.**

```
phys-schost-1# clnode status
=== Cluster Nodes ===

--- Node Status ---

Node Name                                       Status
---------                                       ------
phys-schost-2                                   Online
phys-schost-1                                   Online
```

If any node of the issuing cluster is offline, wait until the node is back online before you accept the site invitation. The acceptance of a site invitation will fail if any node in the issuing cluster is not online.

2. **From a node of the invited cluster, assume the `root` role or assume a role that is assigned the Geo Management rights profile.**

For more information, see "Securing Geographic Edition Software" in *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*.

Note - If you use a role with Geo Management rights, ensure that the `/var/cluster/geo` access control lists (ACLs) are correct on each node of both partner clusters. If necessary, assume the `root` role on the cluster node and set the correct ACLs.

```
# chmod A+user:username:rwx:allow /var/cluster/geo
```

The `/var/cluster/geo` directory must have the correct ACL applied for compatibility between the Geo Management rights profile and data replication software.

---

3. **Ensure that all nodes of the invited cluster are online.**

invited-cluster-node`# cluster status -t node`

4. **Accept the invitation to join the site.**

invited-cluster-node`# geosite join` *issuing-cluster site*

issuing-cluster

The name of the cluster that issued the `geosite` command to add the invited cluster.

site

The name of the site that the invited cluster is joining.

5. **Verify that the cluster is a member of the site.**

   The following command lists all sites that the issuing cluster is a member of.

   ```
   invited-cluster-node# geosite list
   site
   ```

Troubleshooting

If a `geosite join` operation fails due to a failure to verify trust, ensure that the common agent container is running on all nodes of both the invited cluster and the issuing cluster. If a node is not running the common agent container, issue the `/usr/sbin/cacaoadm start` command on that node.

If a `geosite join` operation fails due to a timeout, re-issue the command with a `joinTimeout` property value larger than the default of 30. For example, `geosite join` *cluster* `-p joinTimeout=120` *site*.

Changing the Site Role of a Site Cluster

This section provides the following procedures to change the role of a site cluster:

- "How to Change a Site Membership Role" on page 98
- "How to Forcibly Change a Site Member to a Site Controller" on page 99

A cluster that is part of a site can have one of two roles, controller or member:

- A cluster that has the controller role for a site is authorized to perform operations on the other site members and the multigroups they manage. The controller authorization is based on mutual agreement, any cluster can propose itself as a controller, but it can only control those clusters which have then agreed to accept its authority.
- A cluster that is a simple site member has agreed to accept commands from the site controllers, but cannot issue commands with site-wide effect.

When no existing site controller is accessible, a site member can be forcibly changed to a site controller. The site must have at least one other accessible member.

The site member submits itself or another site member as a candidate for site controller. Each site member cluster must individually accept the candidate cluster as a new controller. No controller operations that are issued by the candidate cluster are obeyed by a site member cluster that has not yet accepted the candidate cluster as a controller.

- If a site controller cluster was unavailable at the time a site member cluster submits itself to be a controller, the controller automatically accepts and synchronizes the site configuration change when it again becomes available.

- If the site member is the only cluster in the site, the controller role is immediately assigned to the cluster. If later another cluster is added to the site, the new cluster must then accept the existing cluster's state as a site controller.

How to Change a Site Membership Role

Perform this procedure to change the role of a site cluster between member and controller.

Note - If the site has no reachable site controller, instead perform "How to Forcibly Change a Site Member to a Site Controller" on page 99.

Note - You can also use the Oracle Solaris Cluster Manager browser interface to perform this task. Click Sites, click the site name to go to its page, highlight the cluster name, and click Set Role. For Oracle Solaris Cluster Manager log-in instructions, see "How to Access Oracle Solaris Cluster Manager" in *Oracle Solaris Cluster 4.3 System Administration Guide*.

Before You Begin

- If you use a role with Geo Management rights, ensure that the `/var/cluster/geo` access control lists (ACLs) are correct on each node of both partner clusters. If necessary, assume the `root` role on the cluster node and set the correct ACLs.

  ```
  # chmod A+user:username:rwx:allow /var/cluster/geo
  ```

  The `/var/cluster/geo` directory must have the correct ACL applied for compatibility between the Geo Management rights profile and data replication software.

  For more information, see "Securing Geographic Edition Software" in *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*.

1. **From a node of a site controller cluster, assume the `root` role or assume a role that is assigned the Geo Management rights profile.**

2. **Change the site role of the cluster.**

   - **To change a site member to a site controller, use the following command.**

     site-controller# **`geosite set-role -c`** *cluster site*

   - **To change a site controller to a site member, use the following command.**

     site-controller# **`geosite set-role -m`** *cluster site*

`-c`

Specifies the site controller role.

`-m`

Specifies the site member role.

cluster

The name of the cluster to change roles. You can specify more than one cluster from the same site in a single `geosite set-role` command by using the `-c` or `-m` option for each cluster that you want to change roles.

site

The name of the site with which the cluster has membership.

▼ How to Forcibly Change a Site Member to a Site Controller

Perform this procedure to change the role of a site member to a site controller when no existing site controller is accessible. The site must have at least one other accessible member.

Note - You can also use the Oracle Solaris Cluster Manager browser interface to perform this task. Click Sites, click the site name to go to its page, highlight the cluster name, click Set Role, and set the force option. For Oracle Solaris Cluster Manager log-in instructions, see "How to Access Oracle Solaris Cluster Manager" in *Oracle Solaris Cluster 4.3 System Administration Guide*.

The site member submits itself or another site member as a candidate for site controller. Each site member cluster must individually accept the candidate cluster as a new controller. No controller operations that are issued by the candidate cluster are obeyed by a site member cluster that has not yet accepted the candidate cluster as a controller.

- If a site controller cluster was unavailable at the time a site member cluster submits itself to be a controller, the controller automatically accepts and synchronizes the site configuration change when it again becomes available.
- If the site member is the only cluster in the site, the controller role is immediately assigned to the cluster. If later another cluster is added to the site, the new cluster must then accept the existing cluster's state as a site controller.

Before You Begin

- If you use a role with Geo Management rights, ensure that the `/var/cluster/geo` access control lists (ACLs) are correct on each node of both partner clusters. If necessary, assume the `root` role on the cluster node and set the correct ACLs.

  ```
  # chmod A+user:username:rwx:allow /var/cluster/geo
  ```

  The `/var/cluster/geo` directory must have the correct ACL applied for compatibility between the Geo Management rights profile and data replication software.

  For more information, see "Securing Geographic Edition Software" in *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*.

1. **From a node of the site member cluster that you want to make a site controller, assume the `root` role or assume a role that is assigned the Geo Management rights profile.**

2. **Submit the request to become a site controller.**

   candidate-cluster-node`#` **`geosite set-role -c`** *candidate-cluster site*

   `-c` *candidate-cluster*

   The name of the cluster that is requesting to be made a site controller. The candidate cluster can be the issuing cluster or another site member cluster.

   site

   The name of the site for which the candidate cluster is requesting the site controller role.

3. **From another member cluster in the site, accept the request from the candidate cluster.**

   A site member cluster in the site must accept the role change request before the change will take effect for that cluster.

   Note - If the site has no other clusters, omit the remaining steps in this procedure.

   site-member-cluster-node`#` **`geosite accept`** *candidate-cluster site*

   candidate-cluster

   The name of the cluster that requested to be made a controller of the specified site.

4. **Verify on the accepting cluster that the candidate cluster is now a site controller.**

   site member-cluster-node`#` **`geosite show -v`** *site*
   …
   candidate-cluster `controller`

5. **Repeat Step 3 and Step 4 for each remaining cluster in the site.**

Troubleshooting If more than one site member cluster proposes itself as a site controller and different site clusters accept a different new site controller, this creates a conflict for control of the site. To avoid such conflicts, promote only one cluster as a site controller and accept it on all accessible site clusters before you promote another cluster as a new site controller.

If a conflict for control of the site does occur, identify which cluster should be the site controller, then issue the `geosite update` *desired-site-controller site* command from each accessible site cluster that is in a synchronization error with that cluster.

If any cluster is down at the time that a site cluster is promoted to site controller, after the site cluster returns to service, ensure that it has no synchronization conflicts and, if necessary, use the `geosite update` command to resolve any conflicts.

Synchronizing Site Configuration Information

Under normal circumstances, a membership change or a change to a site property is automatically propagated to all clusters in the site. This section describes how to manually update site configuration information on the local cluster with configuration information from the remote site cluster. The local site configuration information is overwritten by information from the remote cluster.

▼ How to Synchronize Site Configuration Information

Perform this procedure to update site configuration information on the local cluster.

Note - You can also use the Oracle Solaris Cluster Manager browser interface to perform this task. Click Sites, highlight the site name, and click Validate Site. For Oracle Solaris Cluster Manager log-in instructions, see "How to Access Oracle Solaris Cluster Manager" in *Oracle Solaris Cluster 4.3 System Administration Guide*.

Before You Begin If you use a role with Geo Management rights, ensure that the `/var/cluster/geo` access control lists (ACLs) are correct on each node of both partner clusters. If necessary, assume the `root` role on the cluster node and set the correct ACLs.

```
# chmod A+user:username:rwx:allow /var/cluster/geo
```

The `/var/cluster/geo` directory must have the correct ACL applied for compatibility between the Geo Management rights profile and data replication software.

For more information, see "Securing Geographic Edition Software" in *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*.

1. **From a node of a site controller cluster, assume the `root` role or assume a role that is assigned the Geo Management rights profile.**

2. **Synchronize the site configuration information with the remote site cluster.**

   Site configuration information from the remote site cluster overwrites the site configuration information on the local cluster.

   local-site-cluster-node`#` **`geosite update`** *remote-site-cluster site*

   remote-site-cluster

   The name of the remote site cluster whose site configuration information is being imported to the local site cluster. Both clusters must be configured in the same site.

   site

   The name of the site.

See Also For details about the synchronization statuses, see the Synchronization Status section of the `geoadm`(1M) man page.

Removing a Cluster From a Site

This section contains the following procedures to remove a cluster from a site:

- "How to Remove a Cluster From a Site" on page 102
- "How to Remove an Unreachable Cluster From a Site" on page 104

▼ How to Remove a Cluster From a Site

Perform this procedure to make a cluster remove itself from a site. If the cluster you want to remove from a site is unreachable, instead follow "How to Remove an Unreachable Cluster From a Site" on page 104.

Note - To remove the last cluster from a site, instead follow "How to Delete a Site" on page 105.

Note - You can also use the Oracle Solaris Cluster Manager browser interface to perform this task.

- To remove the local cluster, click Sites, highlight the site name, and click Leave Site.
- To remove another cluster, click Sites, click the site name to go to its page, highlight the cluster name, and click Remove,

For Oracle Solaris Cluster Manager log-in instructions, see "How to Access Oracle Solaris Cluster Manager" in *Oracle Solaris Cluster 4.3 System Administration Guide*.

Before You Begin

If you use a role with Geo Management rights, ensure that the `/var/cluster/geo` access control lists (ACLs) are correct on each node of both partner clusters. If necessary, assume the `root` role on the cluster node and set the correct ACLs.

```
# chmod A+user:username:rwx:allow /var/cluster/geo
```

The `/var/cluster/geo` directory must have the correct ACL applied for compatibility between the Geo Management rights profile and data replication software.

For more information, see "Securing Geographic Edition Software" in *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*.

1. **From a node of the cluster that you want to remove from a site, assume the `root` role or assume a role that is assigned the Geo Management rights profile.**

2. **Detach the issuing cluster from the site.**

   - **If the cluster is not the last controller in the site, use the following command.**

     node-of-cluster-to-remove`# geosite leave` *site*

     site

     The name of the site to remove the cluster from.

   - **If the cluster is the last controller in the site, include the `-f` option in the command.**

     node-of-cluster-to-remove`# geosite leave -f` *site*

- If the site contains other member clusters, the site continues to exist without a site controller.
- If the site has no other member clusters, removal of the last controller also deletes the site.

Note - If the last site member being removed is not a site controller, the remove command fails with an error.

---

3. **From a remaining site cluster, verify that the removed cluster is no longer listed as a site member.**

```
# geosite show -v site

        *** Site "site" is not configured ***
```

How to Remove an Unreachable Cluster From a Site

Perform this procedure to remove from a site a cluster that is unreachable. If a cluster you want to remove is reachable, instead follow "How to Remove a Cluster From a Site" on page 102.

Note - To remove the last cluster from a site, instead follow "How to Delete a Site" on page 105.

Note - You can also use the Oracle Solaris Cluster Manager browser interface to perform this task. Click Sites, click the site name to go to its page, highlight the cluster name, and click Remove, For Oracle Solaris Cluster Manager log-in instructions, see "How to Access Oracle Solaris Cluster Manager" in *Oracle Solaris Cluster 4.3 System Administration Guide*.

Before You Begin

If you use a role with Geo Management rights, ensure that the `/var/cluster/geo` access control lists (ACLs) are correct on each node of both partner clusters. If necessary, assume the `root` role on the cluster node and set the correct ACLs.

```
# chmod A+user:username:rwx:allow /var/cluster/geo
```

The `/var/cluster/geo` directory must have the correct ACL applied for compatibility between the Geo Management rights profile and data replication software.

For more information, see "Securing Geographic Edition Software" in *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*.

1. **From a node of a site controller cluster, assume the `root` role or assume a role that is assigned the Geo Management rights profile.**

2. **Remove the cluster from the site.**

   site-controller-cluster-node`# geosite remove-member` *cluster site*

   cluster

   The name of the cluster to remove from the site. You can remove multiple clusters by specifying each cluster name separated by a comma (,).

3. **From a remaining site cluster, verify that the removed cluster is no longer listed as a site member.**

   `# geosite show -v` *site*

Deleting a Site

This section contains procedures to delete a site. A site is automatically deleted when its last cluster is detached from the site. Each cluster must remove itself from the site. Because the last cluster to remove must be a site controller, first detach all clusters except one site controller. If the last cluster you remove is not a site controller, the command fails with an error.

When a site is deleted, all multigroups that the site references are automatically deleted on all clusters where the site was defined. The deletion of a multigroup has no effect on the protection groups that were configured in the deleted multigroup.

▼ How to Delete a Site

This procedure deletes a site by removing all of its member clusters. Perform this procedure from one site controller cluster.

Note - You can also use the Oracle Solaris Cluster Manager browser interface to perform this task. Click Sites, click the site name to go to its page, in the Clusters section highlight and click Remove for each remote cluster, then highlight the local cluster and click Remove.

For Oracle Solaris Cluster Manager log-in instructions, see "How to Access Oracle Solaris Cluster Manager" in *Oracle Solaris Cluster 4.3 System Administration Guide*.

Before You Begin

- Ensure that the site has at least one site controller cluster. If the site has no site controller cluster, change the role of a site member cluster to site controller before you perform this procedure. See "How to Forcibly Change a Site Member to a Site Controller" on page 99.
- If you use a role with Geo Management rights, ensure that the `/var/cluster/geo` access control lists (ACLs) are correct on each node of both partner clusters. If necessary, assume the `root` role on the cluster node and set the correct ACLs.

  ```
  # chmod A+user:username:rwx:allow /var/cluster/geo
  ```

  The `/var/cluster/geo` directory must have the correct ACL applied for compatibility between the Geo Management rights profile and data replication software.

  For more information, see "Securing Geographic Edition Software" in *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*.

1. **From a node of a site controller cluster, assume the `root` role or assume a role that is assigned the Geo Management rights profile.**

2. **Remove all other clusters from the site.**

   ```
   # geosite remove-member list-of-members site
   ```

   list-of-members

   A comma-separated list of each cluster in the site, excluding the local cluster.

   site

   The name of the site to detach the specified clusters from.

3. **Verify that the local cluster is the only remaining member of the site.**

   ```
   # geosite show -v site
   last-site-controller-cluster controller
   ```

4. **Forcibly detach the local cluster from the site.**

   ```
   # geosite leave -f site
   ```

The site is automatically deleted when the last site cluster is removed.

5. **Verify that the site no longer exists.**

```
# geosite show -v site

        *** Site "site" is not configured ***
```

♦ ♦ ♦ CHAPTER 8

Administering Multigroups

This chapter contains the following information about administering multigroups:

For procedures to create a new multigroup, see "Configuring Sites and Multigroups" in *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*.

For procedures to switch over or take over multigroups, see Chapter 10, "Migrating Services".

Modifying Protection Groups in a Multigroup

When you make a change to a multigroup, the site controller cluster that issues the change propagates the change to all other clusters in the site where the changed multigroup is active.

This section provides the following procedures to add, change, remove, and synchronize protection groups in a multigroup:

▼ How to Add a Protection Group to a Multigroup

Note - You can also use the Oracle Solaris Cluster Manager browser interface to perform this task. Click Sites, click the site name to go to its page, click the multigroup name to go to its page, and click Add Protection Group. For Oracle Solaris Cluster Manager log-in instructions, see "How to Access Oracle Solaris Cluster Manager" in *Oracle Solaris Cluster 4.3 System Administration Guide*.

Before You Begin

- Ensure that a partner cluster of the protection group you are adding is configured in the site.
- If you use a role with Geo Management rights, ensure that the `/var/cluster/geo` access control lists (ACLs) are correct on each node of both partner clusters. If necessary, assume the `root` role on the cluster node and set the correct ACLs.

  ```
  # chmod A+user:username:rwx:allow /var/cluster/geo
  ```

  The `/var/cluster/geo` directory must have the correct ACL applied for compatibility between the Geo Management rights profile and data replication software.

  For more information, see "Securing Geographic Edition Software" in *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*.

1. **From a node of a controller cluster for the site to configure with the new multigroup, assume the `root` role or assume a role that is assigned the Geo Management rights profile.**

2. **Add the protection group to the multigroup.**

   site-controller-cluster-node`# geomg add-protection-group` *protection-group-list multigroup*

   The syntax choices for *protection-group-list* are as follows:

   cluster:*protection-group*

   Specifies a single protection group: The colon (:) separates the cluster name *cluster* from the name of the protection group that is configured in that cluster.

   cluster:*protection-group*/*cluster*:*protection-group*

   Specifies a protection group that has a dependency on another protection group, called a *dependency chain*: The protection group that is specified before the slash (/) in the dependency chain depends on the protection group that is specified after the slash.

   cluster1:*protection-group1*,*cluster1*:*protection-group2*,*cluster2*:*protection-group1*/*cluster3*:*protection-group1*

   The comma (,) separates multiple protection group names in the protection-group list.

(*cluster1*:*protection-group2*,*cluster2*:*protection-group1*)/*cluster3*:*protection-group1*

Specifies that multiple protection groups, *cluster1*:*protection-group2* and *cluster2*:*protection-group1*, all have a dependency on the *cluster3*:*protection-group1* protection group. In this form of dependency chain, parentheses are only used to enclose the multiple protection groups that have a dependency on another, single protection group. Only one protection group can be specified as the depended-on protection group in the dependency chain.

For more information, see the `geomg`(1M) man page.

3. **Verify the addition of the protection groups.**

site-controller-cluster-node# **geomg show** *multigroup*

▼ How to Change the Protection Groups in a Multigroup Dependency Chain

Perform this procedure to change a set of protection groups in a multigroup that are configured in a dependency to or from each other. Such interdependent protection groups are referred to as a *dependency chain*.For more information about configuring dependency chains, see "How to Add a Protection Group to a Multigroup" on page 110.

You must first delete the entire dependency chain of protection groups from the multigroup. You then add back any protection groups in the deleted dependency chain that you want to continue as part of the multigroup.

This procedure does not affect the protection groups themselves.

Note - You can also use the Oracle Solaris Cluster Manager browser interface to perform this task. Click Sites, click the site name to go to its page, click the multigroup name to go to its page, and use the Remove Protection Group and Add Protection Group buttons. For Oracle Solaris Cluster Manager log-in instructions, see "How to Access Oracle Solaris Cluster Manager" in *Oracle Solaris Cluster 4.3 System Administration Guide*.

Before You Begin If you use a role with Geo Management rights, ensure that the `/var/cluster/geo` access control lists (ACLs) are correct on each node of both partner clusters. If necessary, assume the `root` role on the cluster node and set the correct ACLs.

```
# chmod A+user:username:rwx:allow /var/cluster/geo
```

The `/var/cluster/geo` directory must have the correct ACL applied for compatibility between the Geo Management rights profile and data replication software.

For more information, see "Securing Geographic Edition Software" in *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*.

1. **From a node of a site controller cluster, assume the `root` role or assume a role that is assigned the Geo Management rights profile.**

2. **Remove the protection-group dependency chain from the multigroup.**

   Specify each protection-group dependency chain that you want to change.

   `# geomg remove-protection-group` *protection-group-list multigroup*

   protection-group-list

   Name of one or more interdependency protection-groups to remove from the multigroup. Specify the full protection-group dependency that contains the protection group that you want to remove.

   `# geomg remove-protection-group` *dependee-protection-group/depended-on-protection-group multigroup*

   To remove multiple protection groups, with and without a dependency relationship, separate each protection group name with a comma (,).

   `# geomg remove-protection-group` *protection-group,dependee-protection-group/depended-on-protection-group multigroup*

   multigroup

   The name of the multigroup from which you are removing the protection groups.

3. **Add back those protection groups that you want to remain in the multigroup.**

   `# geomg add-protection-group` *protection-group-list multigroup*

▼ How to Remove a Protection Group From a Multigroup

This procedure does not affect the protection group.

Note - If the protection group to remove is part of set of interdependent protection groups, called a dependency chain, do not perform this procedure. Go instead to "How to Change the Protection Groups in a Multigroup Dependency Chain" on page 111.

Note - You can also use the Oracle Solaris Cluster Manager browser interface to perform this task. Click Sites, click the site name to go to its page, click the multigroup name to go to its page, highlight the name of the protection group, and click Remove Protection Group. For Oracle Solaris Cluster Manager log-in instructions, see "How to Access Oracle Solaris Cluster Manager" in *Oracle Solaris Cluster 4.3 System Administration Guide*.

Before You Begin If you use a role with Geo Management rights, ensure that the `/var/cluster/geo` access control lists (ACLs) are correct on each node of both partner clusters. If necessary, assume the `root` role on the cluster node and set the correct ACLs.

```
# chmod A+user:username:rwx:allow /var/cluster/geo
```

The `/var/cluster/geo` directory must have the correct ACL applied for compatibility between the Geo Management rights profile and data replication software.

For more information, see "Securing Geographic Edition Software" in *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*.

1. **From a node of a site controller cluster, assume the `root` role or assume a role that is assigned the Geo Management rights profile.**

2. **Remove the protection group from the multigroup.**

   - **If the protection group to remove is not part of a dependency chain with one or more other protection groups, use the following command:**

     ```
     # geomg remove-protection-group cluster:protection-group multigroup
     ```

   - **If the protection group to remove is part of a dependency chain with one or more other protection groups, use the following command to specify the complete dependency chain of protection groups:**

     ```
     # geomg remove-protection-group protection-group-dependency-chain multigroup
     ```

     cluster

     Name of the cluster where the protection group to remove is configured.

protection-group-dependency-chain

Name of the complete dependency chain of protection groups that contains the protection group that you want to remove. For example, `newyork:pg1/newyork:pg2` is a dependency chain in which the `newyork:pg1` protection group depends on the `newyork:pg2` protection group.

To remove multiple protection groups, separate each protection group or dependency chain name with a comma (,).

3. **If a protection group that you removed was part of a dependency chain, add back any protection groups in the dependency chain that you want to remain in the multigroup.**

```
# geomg add-protection-group protection-group-list multigroup
```

Starting and Stopping Multigroups

This section contains the following procedures:

- "How to Start All Protection Groups in a Multigroup" on page 114
- "How to Stop All Protection Groups in a Multigroup" on page 115

▼ How to Start All Protection Groups in a Multigroup

The protection groups to start can be either those on only the clusters that are specified in the protection-group list of the multigroup or those on both partner clusters where the protection groups are configured.

Note - You can also use the Oracle Solaris Cluster Manager browser interface to perform this task. Click Sites, click the site name to go to its page, highlight the multigroup name, and click Start Protection Groups. For Oracle Solaris Cluster Manager log-in instructions, see "How to Access Oracle Solaris Cluster Manager" in *Oracle Solaris Cluster 4.3 System Administration Guide*.

Before You Begin If you use a role with Geo Management rights, ensure that the `/var/cluster/geo` access control lists (ACLs) are correct on each node of both partner clusters. If necessary, assume the `root` role on the cluster node and set the correct ACLs.

```
# chmod A+user:username:rwx:allow /var/cluster/geo
```

The `/var/cluster/geo` directory must have the correct ACL applied for compatibility between the Geo Management rights profile and data replication software.

For more information, see "Securing Geographic Edition Software" in *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*.

1. **From a node of a site controller cluster, assume the `root` role or assume a role that is assigned the Geo Management rights profile.**
2. **Start all protection groups in the multigroup.**

   - **To start all protection groups on only the clusters specified in the protection-group list of the multigroup, include the `-e local` option in the command.**

     ```
     # geomg start -e local multigroup
     ```

   - **To start all protection groups on both partner clusters where the protection groups are configured, include the `-e global` option in the command.**

     ```
     # geomg start -e global multigroup
     ```

   See the `geomg`(1M) man page for information about additional options for the `start` subcommand.

▼ How to Stop All Protection Groups in a Multigroup

Perform this procedure to stop all protection groups that are configured a multigroup. The protection groups to stop can be either those on only the clusters that are specified in the protection-group list of the multigroup or those on both partner clusters where the protection groups are configured.

Note - You can also use the Oracle Solaris Cluster Manager browser interface to perform this task. Click Sites, click the site name to go to its page, highlight the multigroup name, and click Stop Protection Groups. For Oracle Solaris Cluster Manager log-in instructions, see "How to Access Oracle Solaris Cluster Manager" in *Oracle Solaris Cluster 4.3 System Administration Guide*.

Before You Begin If you use a role with Geo Management rights, ensure that the `/var/cluster/geo` access control lists (ACLs) are correct on each node of both partner clusters. If necessary, assume the `root` role on the cluster node and set the correct ACLs.

```
# chmod A+user:username:rwx:allow /var/cluster/geo
```

The `/var/cluster/geo` directory must have the correct ACL applied for compatibility between the Geo Management rights profile and data replication software.

For more information, see "Securing Geographic Edition Software" in *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*.

1. **From a node of a site controller cluster, assume the `root` role or assume a role that is assigned the Geo Management rights profile.**
2. **Stop all protection groups in the multigroup.**

   - **To stop all protection groups on only the clusters specified in the protection-group list of the multigroup, include the `-e local` option in the command.**

     ```
     # geomg stop -e local multigroup
     ```

   - **To stop all protection groups on both partner clusters where the protection groups are configured, include the `-e global` option in the command.**

     ```
     # geomg stop -e global multigroup
     ```

   See the `geomg`(1M) man page for information about additional options for the `stop` subcommand.

Synchronizing Multigroup Configuration Information

Under normal circumstances, information about a multigroup configuration change is automatically synchronized among all clusters in the multigroup.

This section describes how to manually update the locally known configuration information of a multigroup with the configuration information known to another cluster in the site. Synchronization of a multigroup is normally performed automatically after you add or remove a protection group. You would perform this procedure if synchronization failed for one or more clusters of the multigroup site. The configuration information known to the local cluster is overwritten by information from the specified remote cluster in the site.

▼ How to Synchronize Multigroup Configuration Information

Perform this procedure to manually synchronize multigroup configuration information within a site.

Note - You can also use the Oracle Solaris Cluster Manager browser interface to perform this task. Click Sites, click the site name to go to its page, highlight the multigroup name, and click Validate. For Oracle Solaris Cluster Manager log-in instructions, see "How to Access Oracle Solaris Cluster Manager" in *Oracle Solaris Cluster 4.3 System Administration Guide*.

Before You Begin If you use a role with Geo Management rights, ensure that the `/var/cluster/geo` access control lists (ACLs) are correct on each node of both partner clusters. If necessary, assume the `root` role on the cluster node and set the correct ACLs.

```
# chmod A+user:username:rwx:allow /var/cluster/geo
```

The `/var/cluster/geo` directory must have the correct ACL applied for compatibility between the Geo Management rights profile and data replication software.

For more information, see "Securing Geographic Edition Software" in *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*.

1. **From a node of a controller cluster of the local site, assume the `root` role or assume a role that is assigned the Geo Management rights profile.**

2. **Synchronize the multigroup configuration information with the remote cluster in the site.**

   ```
   # geomg update remote-site-cluster multigroup-name
   ```

   For more information about multigroup synchronization states, see the `geoadm`(1M) man page.

Deleting a Multigroup

This section describes procedures to delete a multigroup. This operation has no effect on the protection groups that are configured in the deleted multigroup.

Alternatively, if you intend to delete a site that references the multigroup you want to delete, instead follow "How to Delete a Site" on page 105. When you delete a site, all multigroups that the site references are automatically deleted as well.

How to Delete a Multigroup

Perform this procedure to delete a multigroup from all clusters of a site where the multigroup exists.

Note - You can also use the Oracle Solaris Cluster Manager browser interface to perform this task. Click Sites, click the site name to go to its page, highlight the multigroup name, and click Delete. For Oracle Solaris Cluster Manager log-in instructions, see "How to Access Oracle Solaris Cluster Manager" in *Oracle Solaris Cluster 4.3 System Administration Guide*.

Before You Begin

If you use a role with Geo Management rights, ensure that the `/var/cluster/geo` access control lists (ACLs) are correct on each node of both partner clusters. If necessary, assume the `root` role on the cluster node and set the correct ACLs.

```
# chmod A+user:username:rwx:allow /var/cluster/geo
```

The `/var/cluster/geo` directory must have the correct ACL applied for compatibility between the Geo Management rights profile and data replication software.

For more information, see "Securing Geographic Edition Software" in *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*.

1. **From a node of a controller cluster of the site where the multigroup exists, assume the `root` role or assume a role that is assigned the Geo Management rights profile.**

2. **Delete the multigroup.**

   controller-cluster-node`# geomg delete` *multigroup*

3. **Verify that the multigroup is deleted.**

   controller-cluster-node`# geomg show` *multigroup*

♦ ♦ ♦ CHAPTER 9

Monitoring and Validating the Geographic Edition Framework

This chapter describes the files and tools that you can use to monitor and validate the Geographic Edition framework.

This chapter contains the following sections:

Also see the applicable Geographic Edition data replication guide for additional procedures to check the runtime status of that particular type of protection group.

Monitoring the Runtime Status of the Geographic Edition Framework

You can display the runtime status of the local Geographic Edition enabled cluster by using the `geoadm status` command. This command displays output that is organized in the following sections:

- **Cluster** – Provides the name of the local cluster
- **Partnership** – Provides information about all partnerships, including the name of the partner cluster, the synchronization state, the local heartbeats, and the local heartbeat plug-in
- **Protection group** – Provides information about the status of protection groups, including information about the local cluster and the remote cluster
- **Site** – Provides information about all sites, including the names of the sites, the name and role of each site cluster, site heartbeat, and site heartbeat plug-in,

- **Multigroup** – Provides information about all multigroups, including the names of the multigroups, the name of associated sites, and synchronization status
- **Pending operations** – Provides status information about any ongoing transaction processes

You must be assigned the Basic Solaris User rights profile to run the `geoadm status` command. For more information, see "Securing Geographic Edition Software" in *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*.

The following example shows sample output when an administrator runs the `geoadm status` command on the `cluster-paris` cluster.

```
phys-paris-1# geoadm status

   Cluster:  cluster-paris

   Partnership "paris-newyork-ps": OK
        Partner clusters        : cluster-newyork
        Synchronization         : OK
        ICRM Connection         : OK

        Heartbeat "paris-to-newyork" monitoring "cluster-newyork": OK
             Plug-in "ping_plugin"      : Inactive
             Plug-in "tcp_udp_plugin"   : OK

   Protection group "tcpg"       : OK
        Partnership             : paris-newyork-ps
        Synchronization         : OK

        Cluster cluster-paris   : OK
             Role               : Primary
             Activation state   : Deactivated
             Configuration      : OK
             Data replication   : OK
             Resource groups    : None

        Cluster cluster-newyork : OK
             Role               : Secondary
             Activation state   : Deactivated
             Configuration      : OK
             Data replication   : OK
             Resource groups    : None

   Protection group "testpg"     : OK
        Partnership             : paris-newyork-ps
        Synchronization         : OK

        Cluster cluster-paris   : OK
             Role               : Primary
```

```
             Activation state   : Deactivated
             Configuration      : OK
             Data replication   : OK
             Resource groups    : None

        Cluster cluster-newyork : OK
             Role               : Secondary
             Activation state   : Deactivated
             Configuration      : OK
             Data replication   : OK
             Resource groups    : None

   Site : site1

        Controller "cluster-paris"
             Configuration       : OK

        Controller "cluster-madrid"
             Configuration       : OK
             Synchronization     : OK

             Heartbeat "paris-to-madrid~site1~cluster-madrid" monitoring "cluster-
madrid": OK
                  Plug-in "tcp_udp_plugin"   : OK

        Member "cluster-london"
             Configuration       : OK
             Synchronization     : OK

             Heartbeat "paris-to-london~site1~cluster-london" monitoring "cluster-
london": OK
                  Plug-in "tcp_udp_plugin"   : OK

   Multigroup "mg2"     :
        Site            : site1
        Configuration   : OK
        Synchronization with cluster cluster-madrid      : OK
        Synchronization with cluster cluster-london     : OK

   Multigroup "mg1"     :
        Site            : site1
        Configuration   : OK
        Synchronization with cluster cluster-madrid      : OK
        Synchronization with cluster cluster-london     : OK

Pending operations:

Protection group "tcpg" operation: Start
```

```
Pending multigroup operations:

Multigroup "mg1" operation: Start
```

The information displayed shows that the protection group, `tcpg`, is started on both the primary cluster, `cluster-paris`, and the secondary cluster, `cluster-newyork`. Data is replicating between the partner clusters and both partners are synchronized.

The following table describes the meaning of the status values.

TABLE 5 Status Value Descriptions

Field	Value Descriptions
`Partnership`	`OK` – The partners are connected. `Error` – The connection between the partner clusters is lost. `Degraded` – The partnership has been successfully created but a connection with the partner cluster has not yet been established. This status value occurs when the partnership has been created and the partner cluster has not been configured.
`Synchronization`	`OK` – The configuration information is synchronized between partner clusters. `Error` – The configuration information differs between the partner clusters. You need to resynchronize the partnership for a partnership synchronization error, or resynchronize the protection group for a protection group synchronization error. ■ For information about resynchronizing a partnership, see "Resynchronizing a Partnership" on page 52. ■ For information about resynchronizing a protection group, see "Resynchronizing a Protection Group" on page 84. `Mismatch` – Configuration information has been created separately on the clusters. The configuration information must be replaced by a copy of the configuration information from the partner cluster. You can synchronize the protection group configuration by using the `geopg get` command. `Unknown` – Information is not accessible because the partners are disconnected or because some components of the protection group cannot be reached.
`ICRM Connection`	`OK` – The Intercluster Resource Management (ICRM) module is running properly. `Error` – The ICRM module on the local cluster is unable to communicate with the ICRM module on the remote cluster.
`Heartbeat`	`OK` – Heartbeat checks are running and the partner cluster responds within the specified timeout and retry periods. `Offline` – Heartbeat checks are not running. `Error` – Heartbeat checks are running but the partner is not responding and retries have timed out.

Field	Value Descriptions
	`Degraded` – Heartbeat checks are running but one of the primary heartbeat plug-ins is degraded or is not running.
`Heartbeat plug-in`	`OK` – Responses are being received from the partner. `Inactive` – Plug-in is not in use but is a standby for retrying to contact the partner if the other plug-ins obtain no response. `No-Response` – Partner cluster is not responding.
`Protection group` (overall protection group state)	`OK` – No component of the protection group on either partner is in the Degraded, Error, or Unknown status, and the protection group configuration is the same on both partner clusters. `Degraded` – Data replication is either not running or is in a partial error state. `Error` – At least one component of the partnership is in an error state on at least one partner, or the protection group configuration is different between the partner clusters. `Unknown` – The status for at least one component of the protection group is unknown, or the status of the protection group is not accessible.
`Protection group` > `Synchronization` (state of protection group configuration information between partner clusters)	`OK` – The configuration is synchronized between partner clusters. `Error` – The configuration on the partner clusters is different. You must synchronize the protection group again. `Mismatch` – The protection group has been configured on each partner cluster individually. You must remove the configuration from one cluster and copy the configuration of the partner cluster. `Unknown` – Information is not accessible because the partners are disconnected.
`Protection group` > `Cluster` (state of protection group on each cluster)	`None` – The data replication or resource group component is not configured in the protection group. `OK` – The state of all the protection group components, such as configuration data, data replication, or resource groups, is `OK`, `NONE`, or `N/A` on the cluster. `Degraded` – The state of one or more of the protection group components is in the `Degraded` state on the cluster. `Error` – The state of some components of the protection group, such as configuration data, data replication, or resource groups, is in `Error`. `Unknown` – The state of some components of the protection group, such as configuration data, data replication, or resource groups, is unavailable.
`Protection group` > `Cluster` > `Role`	`Primary` – The cluster is the primary cluster for this protection group. `Secondary` – The cluster is the secondary cluster for this protection group. `Unknown` – Information is not accessible because the partners are disconnected or because some components of the protection group cannot be reached.

Field	Value Descriptions
`Protection group` > `Cluster` > `Activation state`	`Activated` – The protection group is activated. `Deactivated` – The protection group is deactivated. `Unknown` – Information is not accessible because the partners are disconnected or because some components of the protection group cannot be reached.
`Protection group` > `Cluster` > `Configuration`	`OK` – Protection group configuration has been validated without errors on the cluster. `Error` – Protection group configuration validation resulted in errors on the cluster. You need to revalidate the protection group. For information about validating a protection group, see "Validating a Protection Group" in *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*. `Unknown` – Information is not accessible because the partners are disconnected or because some components of the protection group cannot be reached.
`Protection group` > `Cluster` > `Data replication`	`None` – Data replication is not configured. `OK` – Data replication is running and data is synchronized with the partner cluster when the protection group is activated. Replication is suspended when the protection group is deactivated. This state represents data replication on this cluster and does not reflect the overall state of data replication. This state is mapped from the corresponding state in the data replication subsystem. `Degraded` – Data is not replicated and not synchronized with the partner cluster when the protection group is activated. New writes will succeed but not be replicated. This state represents data replication on this cluster and does not reflect the overall state of data replication. This state is mapped from the corresponding state in the data replication subsystem. `Error` – Data replication from the primary cluster to the secondary cluster is in error if the data replication subsystem reports an error or if data replication is not suspended when the protection group is deactivated. This state represents data replication on this cluster and does not reflect the overall state of data replication. This state is mapped from the corresponding state in the data replication subsystem. `Unknown` – Information is not accessible because the partners are disconnected or because some components of the protection group cannot be reached. `N/A` – The data replication state of the protection group could not be mapped. Data replication is in a valid state on its own but in an `Error` state for the protection group. This state is available only if you are using Availability Suite data replication.
`Protection group` > `Cluster` > `Resource groups`	`None` – No resource group is protected by this protection group. `OK` – If the cluster has the `Primary` role, all resource groups are online when the protection group is activated or unmanaged when the protection group is deactivated. If the cluster has the `Secondary` role, all resource groups are unmanaged.

Field	Value Descriptions
	`Error` – If the cluster has the `Primary` role, not all resource groups are online when the protection group is activated or unmanaged when the protection group is deactivated. If the cluster has the `Secondary` role, not all resource groups are unmanaged. `Unknown` – Information is not accessible because the partners are disconnected or because some components of the protection group cannot be reached.
`Site` > `Controller` or `Member` > `Configuration`	`OK` – The site configuration is correct. `Error` – The site configuration has an error. `Unknown` – The site configuration is not checked.
`Site` > `Controller` or `Member` > `Synchronization`	`DIFFERENT` – Site configuration information on two compared clusters is different but compatible. This is a transitory status. Eventually, the older of the two configurations is automatically synchronized with the newer configuration, at which time the synchronization status becomes OK. If for some reason the status remains at DIFFERENT, use the `geosite validate` subcommand to synchronize the site configurations. `ERROR` – The site configuration on two compared clusters is different and cannot be automatically resolved. Configuration changes from an issuing controller cluster are not accepted by a cluster while it is in the ERROR synchronization status with the issuing controller. If configuration information between two clusters do not resolve automatically and the clusters are in the ERROR status, use the `geosite update` subcommand to resolve site configuration conflicts between the two clusters. `OK` – Site configuration information matches on the compared clusters. `UNKNOWN` – Site configuration information cannot be compared because the Geographic Edition framework cannot reach a cluster. Configuration changes from an issuing controller cluster are not propagated to a cluster while it is in the UNKNOWN synchronization status. If the cluster is not automatically synchronized, use the `geosite validate` subcommand to update the cluster with the latest site configuration information.
`Site` > `Controller` or `Member` > `Heartbeat`	`OK` – Heartbeat monitoring is enabled, and the partner cluster is responding within timeout and retry periods. `Degraded` – Heartbeat checks are running but one of the primary heartbeat plug-ins is degraded or is not running. `Error` – Heartbeat monitoring is running but the partner cluster is not responding and retries have timed out. `Offline` – Heartbeat monitoring is offline.
`Site` > `Controller` or `Member` > `Heartbeat` > `Plug-in`	`OK` – The partner cluster is responding. `Inactive` – The plug-in is not in use. It is a standby plug-in that is used for retrying if other plug-ins do not respond. `No-response` – The partner cluster is not responding.

Field	Value Descriptions
`Multigroup` > `Configuration`	`OK` – The configuration of the multigroup is correct and validated without errors. `Error` – The configuration of the multigroup has errors. Run the `geomg validate` command on the multigroup to identify the error. `Unknown` – The configuration is not accessible.
`Multigroup` > `Synchronization`	`DIFFERENT` – Multigroup configuration information on two compared clusters is different but compatible. This is a transitory status. Eventually, the older of the two configurations is automatically synchronized with the newer configuration, at which time the synchronization status becomes OK. If for some reason the status remains at DIFFERENT, use the `geomg validate` command to synchronize the multigroup configurations. `ERROR` – The multigroup configuration on two compared clusters is different and cannot be automatically resolved. Configuration changes from an issuing controller cluster are not accepted by a cluster while it is in the ERROR synchronization status with the issuing controller. If configuration information between two clusters do not resolve automatically and the clusters are in the ERROR status, use the `geomg update` command to resolve multigroup configuration conflicts between the two clusters. `OK` – Multigroup configuration information matches on the compared clusters. `UNKNOWN` – Multigroup configuration information cannot be compared because the Geographic Edition framework cannot reach a cluster. Configuration changes from an issuing controller cluster are not propagated to a cluster while it is in the UNKNOWN synchronization status. If the cluster is not automatically synchronized after the Geographic Edition framework is started on that cluster, use the `geomg validate` command to update the cluster with the latest multigroup configuration information.

Viewing the Geographic Edition Log Messages

All the Geographic Edition components produce messages that are stored in log files.

Information about the loading, running, and stopping Geographic Edition components in the common agent container is recorded in the following log files. The most recently logged messages are in file 0, then 1, and 2.

- `/var/cacao/instances/default/logs/cacao.0`
- `/var/cacao/instances/default/logs/cacao.1`
- `/var/cacao/instances/default/logs/cacao.2`

System log messages are stored in the `/var/adm/messages` log file.

Each cluster node keeps separate copies of the previous log files. The combined log files on all cluster nodes form a complete snapshot of the currently logged information. The log messages of the Geographic Edition modules are updated on the node where the Geographic Edition framework is currently active. The data replication control-log messages are updated on the node where the data replication resource is currently `Online`.

For data replication modules that are based on script-based plug-ins, you can set the `DEBUG` property to `TRUE` for more verbose messages to aid in troubleshooting.

Displaying Configuration Information for Partnerships and Protection Groups

You can display the current local cluster partnership configuration, including a list of all partnerships that are defined between the local cluster and remote clusters.

You can also display the current configuration of a specific protection group or of all the protection groups that are defined on a cluster.

This section provides the following procedures:

- "How to Display Configuration Information About Partnerships" on page 127
- "How to Display Configuration Information About Protection Groups" on page 128

▼ How to Display Configuration Information About Partnerships

Note - You can also accomplish this procedure by using the Oracle Solaris Cluster Manager browser interface. Click Partnerships, then click the partnership name. For Oracle Solaris Cluster Manager log-in instructions, see "How to Access Oracle Solaris Cluster Manager" in *Oracle Solaris Cluster 4.3 System Administration Guide*.

Before You Begin You must be assigned the Basic Solaris User rights profile to complete this procedure. For more information, see "Securing Geographic Edition Software" in *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*.

1. **Log in to a cluster node.**
2. **Display information about the partnership.**

```
# geops list partnership-name
```

partnership-name

Specifies the name of the partnership. If you do not specify a partnership, then the `geops list` command displays information on all partnerships.

For information about the names and values that are supported by the Geographic Edition framework, see "Legal Names and Values of Geographic Edition Entities" on page 24.

Example 28 Displaying Partnership Configuration Information

This example displays configuration information about the partnership between local `cluster-paris` and remote `cluster-newyork`.

```
# geops list paris-newyork-ps
```

▼ How to Display Configuration Information About Protection Groups

Note - You can also accomplish this procedure by using the Oracle Solaris Cluster Manager browser interface. Click Partnerships, click the partnership name to go to its page, and click the protection group name. For Oracle Solaris Cluster Manager log-in instructions, see "How to Access Oracle Solaris Cluster Manager" in *Oracle Solaris Cluster 4.3 System Administration Guide*.

Before You Begin You must be assigned the Basic Solaris User rights profile to complete this procedure. For more information, see "Securing Geographic Edition Software" in *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*.

1. **Log in to a cluster node.**

2. **Display information about a protection group.**

```
# geopg list [protection-group]
```

protection-group

Specifies the name of a protection group.

If you do not specify a protection group, then the command lists information about all the protection groups that are configured on your system.

Example 29 Displaying Configuration Information About a Protection Group

This example displays configuration information for the `avspg` protection group, which is configured on `cluster-paris`.

```
# geopg list avspg
```

♦ ♦ ♦ CHAPTER 10

Migrating Services

This chapter provides information about detecting cluster failure and moving services to an accessible cluster.

Detecting Cluster Failure

This section describes the internal processes that occur when failure is detected on a primary or a secondary cluster.

Detecting Primary Cluster Failure

When the primary cluster for a protection group fails, the secondary cluster in the partnership detects the failure. The cluster that fails might be a member of more than one partnership, resulting in multiple failure detections.

During a failure, the appropriate protection groups are in the `Unknown` state on the cluster that failed. The following actions take place when a primary cluster failure occurs:

- Heartbeat failure is detected by a partner cluster.
- The heartbeat is activated in emergency mode to verify that the heartbeat loss is not transient and that the primary cluster has failed. The heartbeat remains in the `Online` state during this default timeout interval, while the heartbeat mechanism continues to retry the primary cluster.

This query interval is set by using the `Query_interval` heartbeat property. If the heartbeat still fails after the interval you configured, a heartbeat-lost event is generated and logged in the system log. When you use the default interval, the emergency-mode retry behavior might delay heartbeat-loss notification for about nine minutes. Messages are displayed in the Oracle Solaris Cluster Manager browser interface and in the output of the `geoadm status` command.

For more information about logging, see "Viewing the Geographic Edition Log Messages" on page 126.

- If the partnership is configured for heartbeat-loss notification, then one or both of the following actions occurs:
  - An email is sent to the address specified in the `Notification_emailaddrs` property.
  - The script defined in `Notification_actioncmd` is executed.

  For more information about configuring heartbeat-loss notification, see "Configuring Heartbeat-Loss Notification" on page 66.

Detecting Secondary Cluster Failure

When a secondary cluster for a protection group fails, a cluster in the same partnership detects the failure. The cluster that failed might be a member of more than one partnership, resulting in multiple failure detections.

During failure detection, the following actions take place:

- Heartbeat failure is detected by a partner cluster.
- The heartbeat is activated in emergency mode to verify that the secondary cluster is dead.
- When a failure is confirmed by the Geographic Edition framework, the cluster notifies the administrator. The system detects all protection groups for which the cluster that failed was acting as secondary. The state of the appropriate protection groups is marked `Unknown`.

Migrating Replication Services by Switching Over Protection Groups

Perform a switchover of a protection group when you want to migrate services to the partner cluster in an orderly fashion. You can switch over an individual protection group or switch over a multigroup of multiple protection groups in a single operation.

This section contains the following information:

Actions Performed by the Geographic Edition Framework During a Switchover

A switchover consists of the following actions:

- Application services are unmanaged on the former primary cluster.
- The replication role is reversed and now continues to run from the new primary.
- Application services are brought online on the new primary cluster.

The `geopg switchover` command performs the following actions:

1. The command confirms that the secondary cluster does allow the replication role reversal.
2. The command creates copies of the replicated source projects and exports the shares for use on the target appliance, to ensure that no projects and mount point conflicts can occur.
3. The command destroys the created clones on the target appliance and confirms that the actual reverse replication operation can be performed on the target appliance.

Note - When migrating EMC SRDF replication services, basic Geographic Edition operations such as `geopg switchover` perform a `symrdf swap` operation. The `symrdf swap` operation requires significantly more time for static RDF than dynamic RDF. Therefore, you might need to increase the value of the `Timeout` property of the protection group when using static RDF.

The following data replication products have additional behaviors:

- **Hitachi TrueCopy and Universal Replicator**:

    1. The replication subsystem checks that the Hitachi TrueCopy or Universal Replicator device group is in a valid aggregate device group state.
    2. The replication subsystem checks that the local device group states on the target primary cluster, `cluster-newyork`, are `23`, `33`, `43`, or `53`.

The local device group state is returned by the `pairvolchk -g` *device-group-name* `-ss` command. These values correspond to a `PVOL_PAIR` or `SVOL_PAIR` state.

The Hitachi TrueCopy or Universal Replicator commands that are run on the new primary cluster, `cluster-newyork`, are described in the following table.

Aggregate Device Group State	Valid Device Group State on Local Cluster	Hitachi TrueCopy and Universal Replicator Switchover Commands That Are Run on `cluster-newyork`
SMPL	None	None
Regular primary	`23`, `43`	No command is run, because the Hitachi TrueCopy or Universal Replicator device group is already in the `PVOL_PAIR` state.
Regular secondary	`33`, `53`	`horctakeover -g dg [-t]` The `-t` option is specified when the `fence_level` of the Hitachi TrueCopy or Universal Replicator device group is `async`. The value is calculated as 80% of the `Timeout` property of the protection group. For example, if the protection group has a `Timeout` value of 200 seconds, the value of `-t` used in this command is 80% of 200 seconds, or 160 seconds.
Takeover primary	None	None
Takeover secondary	None	None

- **Oracle Data Guard**:
  1. Before the switchover, the command checks that the remote database is in an `enabled` state in the Oracle Data Guard Broker configuration. The command also confirms that the configuration is healthy by issuing the Oracle Data Guard command-line interface (`dgmgrl`) `show configuration` command to ensure that the command returns a `SUCCESS` state.
  2. If the output from this command indicates that Oracle Data Guard Broker is busy performing its own health check, the Oracle Data Guard command-line interface retries the command until it receives a `SUCCESS` response or until two minutes have passed. If the command-line interface is unable to get a `SUCCESS` response, the command fails.
- **Oracle ZFS Storage Appliance**:
  1. The command creates copies of the replicated source projects and exports the shares for use on the target appliance to ensure that no projects and mount point conflicts can occur.
  2. The command destroys such created clones on the target appliance and confirms that the actual reverse replication operation can be performed on the target appliance.

The command performs the following actions on the original primary cluster:

- For Availability Suite, removes affinities and resource dependencies between all the application resource groups in the protection group and the internal resource group, such as the lightweight resource groups
- Takes offline the application resource groups in the protection group and places them in the `Unmanaged` state.
- Performs a switchover to the secondary cluster for each replication configuration in the protection group.

On the original secondary cluster, the command takes the following actions:

- Runs the script that is defined in the `RoleChange_ActionCmd` property
- Brings online all application resource groups in the protection group

If the command completes successfully, the secondary cluster becomes the new primary cluster for the protection group. The original primary cluster becomes the new secondary cluster. The application resource groups in the protection group are brought online on the new primary cluster and replication from the appliance that is connected from the new primary cluster to the new secondary cluster begins.

Note - For Hitachi TrueCopy or Universal Replicator, after a successful switchover, at the data replication level the roles of the primary and secondary volumes have been switched. The `PVOL_PAIR` volumes that were in place before the switchover become the `SVOL_PAIR` volumes. The `SVOL_PAIR` volumes in place before the switchover become the `PVOL_PAIR` volumes. Data replication will continue from the new `PVOL_PAIR` volumes to the new `SVOL_PAIR` volumes.

The `Local-role` property of the protection group is also switched regardless of whether the application could be brought online on the new primary cluster as part of the switchover operation. On the cluster on which the protection group had a `Local role` of `Secondary`, the `Local-role` property of the protection group becomes `Primary`. On the cluster on which the protection group had a `Local-role` of `Primary`, the `Local-role` property of the protection group becomes `Secondary`.

Note - For Oracle Data Guard, databases that are associated with the Oracle Data Guard Broker configurations of the protection group have their role reversed according to the role of the protection group on the local cluster. For HA for Oracle configurations, the `dataguard_role` resource property is also updated with the status of the new primary and standby clusters. The shadow Oracle database server resource group and any other application resource groups in the protection group are online on the new primary cluster. Data replication from the new primary cluster to the new standby cluster begins.

Troubleshooting a Switchover

The `geopg switchover` command returns an error if any of the switchover operations fails.

- Run the `geoadm status` command to view the status of each component. For example, the Configuration status of the protection group might be set to `Error` depending on the cause of the failure. The protection group might be activated or deactivated.

  If the Configuration status of the protection group is set to `Error`, revalidate the protection group by using the procedures that are described in "Validating a Protection Group" in *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*.

  If the configuration of the protection group is not the same on each partner cluster, you need to resynchronize the configuration by using the procedures that are described in "Resynchronizing a Protection Group" in *Oracle Solaris Cluster 4.3 Geographic Edition System Administration Guide*.

▼ How to Switch Over Replication From the Primary Cluster to the Secondary Cluster

Note - To switch over a set of protection groups that are configured as a multigroup, instead follow procedures in "How to Switch Over a Multigroup" on page 138.

Note - You can also accomplish this procedure by using the Oracle Solaris Cluster Manager browser interface, if you do not need to use the `-f` force option. Click Partnerships, click the partnership name to go to its page, highlight the protection group name, and click Switchover. For Oracle Solaris Cluster Manager log-in instructions, see "How to Access Oracle Solaris Cluster Manager" in *Oracle Solaris Cluster 4.3 System Administration Guide*.

Before You Begin

Before you switch over a protection group from the primary cluster to the secondary cluster, ensure that the following conditions are met:

- For Oracle ZFS Storage Appliance, data replication is active between the primary cluster and the secondary cluster. That is, the replication is enabled from the source to the target appliances.
- For Oracle ZFS Storage Appliance, the Geographic Edition replication resource for this appliance replication shows the `Online` state.
- For Oracle Data Guard, the Oracle Data Guard Broker `show configuration` command shows a `SUCCESS` state. This state is reflected in the state of the Geographic Edition

replication resource for this Oracle Data Guard Broker configuration, which should show the `online` state.

- The Geographic Edition framework is running on both clusters.
- The secondary cluster is a member of a partnership.
- Both cluster partners can be reached.
- The overall state of the protection group is set to `OK`.
- If you use a role with Geo Management rights, ensure that the `/var/cluster/geo` access control lists (ACLs) are correct on each node of both partner clusters. If necessary, assume the `root` role on the cluster node and set the correct ACLs.

  ```
  # chmod A+user:username:rwx:allow /var/cluster/geo
  ```

  The `/var/cluster/geo` directory must have the correct ACL applied for compatibility between the Geo Management rights profile and Oracle ZFS Storage Appliance software.

  For more information, see "Securing Geographic Edition Software" in *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*.

Caution - For EMC SRDF, if you have configured the `Cluster_dgs` property, only applications that belong to the protection group can write to the device groups specified in the `Cluster_dgs` property.

1. **Assume the `root` role or assume a role that is assigned the Geo Management rights profile.**

2. **Initiate the switchover.**

   The application resource groups that are a part of the protection group are stopped and started during the switchover.

   ```
   phys-paris-1# geopg switchover [-f] -m new-primary-cluster protection-group-name
   ```

   `-f`

   Forces the command to perform the operation without asking you for confirmation.

   `-m` *new-primary-cluster*

   Specifies the name of the cluster that is to be the new primary cluster for the protection group.

   protection-group-name

   Specifies the name of the protection group.

Example 30 Forcing a Switchover From the Primary Cluster to the Secondary Cluster

The following example performs a switchover to the secondary cluster `cluster-newyork`.

```
phys-paris-1# geopg switchover -f -m cluster-newyork example-protection-group
```

Next Steps To fail back a protection group to the original cluster, see "Recovering a Protection Group to a Cluster" on page 144.

▼ How to Switch Over a Multigroup

Perform this procedure to switch operation of all protection groups in a multigroup to partner clusters in the specified site.

Note - To switch over an individual protection group, instead see "How to Switch Over Replication From the Primary Cluster to the Secondary Cluster" on page 136.

Before You Begin

- Ensure that the multigroup is started.
- If you use a role with Geo Management rights, ensure that the `/var/cluster/geo` access control lists (ACLs) are correct on each node of both partner clusters. If necessary, assume the `root` role on the cluster node and set the correct ACLs.

  ```
  # chmod A+user:username:rwx:allow /var/cluster/geo
  ```

  The `/var/cluster/geo` directory must have the correct ACL applied for compatibility between the Geo Management rights profile and Oracle ZFS Storage Appliance software.

  For more information, see "Securing Geographic Edition Software" in *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*.

1. **From a node of a controller cluster in the target site, assume the `root` role or assume a role that is assigned the Geo Management rights profile.**

2. **Switch the multigroup to the partner clusters in the specified site.**

   ```
   # geomg switchover -s site multigroup
   ```

   `-s` *site*

   The name of the site that the multigroup is to switch to.

   multigroup

   The name of the multigroup to switch over.

See the geomg(1M) man page for information about additional options for the `switchover` subcommand.

Next Steps To fail back a protection group to the original cluster, see "Recovering a Protection Group to a Cluster" on page 144.

Forcing a Takeover of a Protection Group

Perform a takeover when applications need to be brought online on the secondary cluster, regardless of whether the data is completely consistent between the primary and the secondary. You can take over an individual protection group or take over multiple protection groups that are configured as a multigroup in a single operation.

The information in this section assumes that the protection group has been started.

This section contains the following information:

Actions Performed by the Geographic Edition Framework During a Takeover

A takeover consists of the following actions:

- If the former primary cluster can be reached and the protection group is not locked for notification handling or some other reason, the protection group is deactivated.
- The data of the former primary cluster is taken over by the new primary cluster.

Note - The data on the former primary cluster might not be consistent with the data on the original primary cluster. Data replication from the new primary cluster to the former primary cluster is stopped.

- Application services are brought online on the new primary cluster.
- The protection group is activated without replication enabled.

For Hitachi TrueCopy or Universal Replicator, when a takeover is initiated by using the `geopg takeover` command, the data replication subsystem runs several validations on both clusters. These steps are conducted on the original primary cluster only if the primary cluster can be reached. If validation on the original primary cluster fails, the takeover still occurs.

First, the replication subsystem checks that the Hitachi TrueCopy or Universal Replicator device group is in a valid aggregate device group state. Then, the replication subsystem checks that the local device group states on the target primary cluster, `cluster-newyork`, are not `32` or `52`. These values correspond to a `SVOL_COPY` state, for which the `horctakeover` command fails.

Tip - Order-of-writes on `SVOL` are not maintained during the Hitachi TrueCopy or Universal Replicator `COPY` state. In a situation where `SVOL` has some out-of-order writes, if the user attempts to bring ZFS online, the import might fail. Creating backups provides protection if the primary goes down while in the `COPY` state. The administrator can use the backup to update the `SVOL` and then issue the `geopg takeover` to bring the configuration online on the secondary cluster.

The Hitachi TrueCopy or Universal Replicator commands that are used for the takeover are described in the following table.

TABLE 6 Hitachi TrueCopy and Universal Replicator Takeover Validations on the New Primary Cluster

Aggregate Device Group State	Valid Local State Device Group State	Hitachi TrueCopy and Universal Replicator Takeover Commands That Are Run on `cluster-newyork`
SMPL	All	No command is run.
Regular primary	All	No command is run.
Regular secondary	All Regular secondary states except `32` or `52` For a list of Regular secondary states, refer to "How the State of the Hitachi TrueCopy or Universal Replicator Data Replication Component Is Validated" in *Oracle Solaris Cluster Geographic Edition Data Replication Guide for Hitachi TrueCopy and Universal Replicator*.	`horctakeover -S -g dg [-t]` The `-t` option is given when the `fence_level` of the Hitachi TrueCopy or Universal Replicator device group is `async`. The value is calculated as 80% of the `Timeout` property of the protection group. For example, if the protection group has a `Timeout` of 200 seconds, the value of `-t` used in this command will be 80% of 200 seconds, or 160 seconds.
Takeover primary	All	No command is run.
Takeover secondary	All	`pairsplit -R -g dg pairsplit -S -g dg`

For more details about takeover and the effects of the `geopg takeover` command, see Appendix B, "Disaster Recovery Administration Example," in *Oracle Solaris Cluster 4.3 Geographic Edition System Administration Guide*.

For details about the possible conditions of the primary and secondary cluster before and after a takeover, see Appendix C, "Post-Takeover Conditions," in *Oracle Solaris Cluster 4.3 Geographic Edition System Administration Guide*.

▼ How to Force Immediate Takeover of a Protection Group by a Secondary Cluster

Note - You can also accomplish this procedure by using the Oracle Solaris Cluster Manager browser interface, if you do not need to use the `-f` force option. Click Partnerships, click the partnership name to go to its page, highlight the protection group name, and click Takeover. For Oracle Solaris Cluster Manager log-in instructions, see "How to Access Oracle Solaris Cluster Manager" in *Oracle Solaris Cluster 4.3 System Administration Guide*.

Note - To take over a set of protection groups that are configured as a multigroup, instead see "How to Take Over a Multigroup" on page 142.

Before You Begin

Before you force the secondary cluster to assume the activity of the primary cluster, ensure that the following conditions are met:

- The Geographic Edition framework is up and running on the secondary cluster.
- The secondary cluster is a member of a partnership.
- The `Configuration` status of the protection group is `OK` on the secondary cluster.
- If you are using EMC Symmetrix Remote Data Facility, ensure that no LUNs in the data replication device group are in the `SyncInProg` state. Otherwise, the application might fail to start on the new primary cluster due to data inconsistencies.
- If you use a role with Geo Management rights, ensure that the `/var/cluster/geo` access control lists (ACLs) are correct on each node of both partner clusters. If necessary, assume the `root` role on the cluster node and set the correct ACLs.

  ```
  # chmod A+user:username:rwx:allow /var/cluster/geo
  ```

  The `/var/cluster/geo` directory must have the ACL applied for compatibility between the Geo Management rights profile and the data replication software.

  For more information, see "Securing Geographic Edition Software" in *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*.

Perform this procedure from a node in the secondary cluster.

1. **Assume the `root` role or assume a role that is assigned the Geo Management rights profile.**

2. **Initiate the takeover.**

```
phys-newyork-1# geopg takeover [-f] protection-group-name
```

`-f`

Forces the command to perform the operation without your confirmation, even if the primary cluster for a protection group in the multigroup is reachable and the protection group is active on that cluster.

protection-group-name

Specifies the name of the protection group.

Example 31 Forcing a Takeover by a Secondary Cluster

The following example forces the takeover of the protection group `example-pg` by the secondary cluster `cluster-newyork`.

The node `phys-newyork-1` is the first node of the secondary cluster. For a reminder of which node is `phys-newyork-1`, see "Example Geographic Edition Cluster Configuration" in *Oracle Solaris Cluster 4.3 Geographic Edition System Administration Guide*.

```
phys-newyork-1# geopg takeover -f example-pg
```

Next Steps For information about the state of the primary and secondary clusters after a takeover, see Appendix C, "Post-Takeover Conditions," in *Oracle Solaris Cluster 4.3 Geographic Edition System Administration Guide*.

To fail back a protection group to the original cluster, see "Recovering a Protection Group to a Cluster" on page 144.

▼ How to Take Over a Multigroup

Perform this procedure to transfer all operation of protection groups in a multigroup to partner clusters in the specified site.

Note - To take over a protection groups that is not configured as a multigroup, instead follow procedures in "How to Force Immediate Takeover of a Protection Group by a Secondary Cluster" on page 141.

Caution - To avoid possible loss of replicated data, use instead the `geomg switchover` command. The `geomg takeover` command is intended only for situations where the potential risk of data loss is justified, such as when the primary site is not available or during unplanned downtime.

Before You Begin

If you use a role with Geo Management rights, ensure that the `/var/cluster/geo` access control lists (ACLs) are correct on each node of both partner clusters. If necessary, assume the `root` role on the cluster node and set the correct ACLs.

```
# chmod A+user:username:rwx:allow /var/cluster/geo
```

The `/var/cluster/geo` directory must have the ACL applied for compatibility between the Geo Management rights profile and the data replication software.

For more information, see "Securing Geographic Edition Software" in *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*.

1. **From a node of a controller cluster of the target site, assume the `root` role or assume a role that is assigned the Geo Management rights profile.**

2. **Initiate takeover of the multigroup by the standby site.**

```
# geomg takeover [-f] -s site multigroup
```

`-f`

Forces takeover, even if the primary cluster for a protection group in the multigroup is reachable and the protection group is active on that cluster.

`-s` *site*

The name of the standby site which is to take over the specified multigroup.

multigroup

The name of the multigroup to take over.

See the `geomg`(1M) man page for information about additional options for the `takeover` subcommand.

Next Steps For information about failing back a protection group to the original cluster, see "Recovering a Protection Group to a Cluster" on page 144.

Recovering a Protection Group to a Cluster

After a successful takeover operation, the secondary cluster becomes the primary for the protection group and the services are online on the secondary cluster. After the recovery of the original primary cluster the services can be brought online again on the original primary by using a process called *failback*.

The Geographic Edition framework supports the following kinds of failback:

- **Failback-switchover.** During a failback-switchover, applications are brought online again on the original primary cluster after the data of the original primary cluster was resynchronized with the data on the secondary cluster.
- **Failback-takeover.** During a failback-takeover, applications are brought online again on the original primary cluster and use the current data on the original primary cluster. Any updates that occurred on the secondary cluster while it was acting as primary are discarded.

If you want to leave the new primary as the primary cluster and the original primary cluster as the secondary after the original primary restarts, you can resynchronize and revalidate the protection group configuration without performing a switchover or takeover.

For procedures to perform a failback-switchover or a failback-takeover, see the Geographic Edition documentation for the data replication product you are using.

♦ ♦ ♦ CHAPTER 11

Customizing Switchover and Takeover Actions

This chapter describes how to create a script that runs when the role of a protection group changes from `Secondary` to `Primary`. The chapter contains the following sections:

Creating a Role-Change Action Script

You can configure the Geographic Edition framework to run a command when a cluster within a protection group changes from the `Secondary` to the `Primary` role. This change can happen as a result of either a switchover or takeover operation.

The action command runs during a switchover or takeover on the new primary cluster when the protection group is started on the new primary cluster. The script is invoked on the new primary cluster after the data replication role changes from `Secondary` to `Primary` and before the application resource groups are brought online. If the data replication role change does not succeed, then the script is not called.

Observe the following requirements for the role-change action script:

- The path to this script must be valid on all nodes of all partner clusters that can host the protection group.
- The script must have execute permissions for the user that launches the script. The intended user can be the `root` role, or an administrator with the necessary `solaris.cluster.*` authorization to execute a switchover or takeover operation plus any other actions that the script performs. For more information about user rights for the Geographic Edition framework, see "Administering Rights Profiles" on page 25.
- The script must begin with the path to the shell, such as `#!bin/ksh`.

The following command runs the script:

```
# custom-action-command-path -o primary -c cluster-name \
-s partnership-name protection-group user-arguments
```

custom-action-command-path

Specifies a path to the action command you have created.

`-o primary`

Specifies that the role being assumed by the cluster is `Primary`.

`-c` *cluster-name*

Specifies the name of the secondary cluster that is assuming the new role of primary cluster.

`-s` *partnership-name*

Specifies the name of the partnership that hosts the protection group.

protection-group

Specifies the name of the protection group that is undergoing the role change.

user-arguments

Specifies static arguments that are passed after all the Geographic Edition supplied options.

This free-form string can be parsed by the script as required. For example, you could specify a list of *key=value* pairs, such as `name=oracle.com,ip=10.1.2.3`. You could also specify a sequence of options, such as `-n oracle.com -a 10.1.2.3.4`. The format of these arguments is not restricted by the Geographic Edition framework.

The exit status of the role-change action script is reported as part of the result of the `geopg switchover` or `geopg takeover` command. The exit status is zero if the action script was started successfully. A nonzero exit status indicates an error or failure. The value of the exit status does not affect other aspects of the role-change actions. The switchover or takeover proceeds to bring the application resource groups in the protection group online, regardless of the exit status of the action script.

The exit status of the action script can impact the `geomg switchover` or `geomg takeover` commands. During switchover or takeover of a multigroup, if the action script returns a nonzero exit status for a protection group that has dependents, the switchover or takeover of those dependent protection groups is not performed.

The Geographic Edition framework waits for the script to return before the software processes operations such as bringing online application resource groups. Therefore, you must know

in advance the amount of time required to run the script for the protection group. Setting the timeout period to include enough time for the script to complete avoids switchovers or takeovers timing out and leaving the application resource group offline on the new primary.

EXAMPLE 32 Switchover Action Script for Updating the DNS

This sample script uses the `nsupdate` command to reconfigure the host name to point to a new cluster. For more information, refer to the `nsupdate`(1M) man page.

Clients that try to connect to `companyX.com` are referred by the name service to the address of the primary cluster for a protection group, `cluster-paris`. When the primary cluster fails to respond, the administrator performs a switchover of the protection group to the alternate cluster, `cluster-newyork`.

```
#!/bin/ksh
# sample script to update dns
# Assumes each cluster has an entry with name "lh-paris-1" in /etc/hosts
# but different value for the IP in each cluster
# for forward DNS (A) entry: will delete old entry for "lh-paris-1"
# and add one that is correct for "this cluster"
#
# For reverse (PTR) DNS entry, will just add one for this cluster.
# Will NOT delete PTR record left over from old cluster. So
# eventually you will just have reverse lookup for the IP for both clusters
# doing reverse resolution to the same name (lh-paris-1.odyssey.com)
# This should be fine, as long as the forward resolution stays "correct"
#
# The blank line of input at the end of nsupdate is REQUIRED
#
# A short TTL is put on the new records (600 = 10 minutes)
# but you can't really control what kind of caching goes on on
# the client side


# get IP corresponding to name "lh-paris-1" on THIS Cluster
NEWIP=$(getent hosts lh-paris-1|cut -f1)

# this bit splits out the octets in order to add the reverse PTR entry
IFS=.
set $NEWIP
unset IFS

/usr/sbin/nsupdate <<ENDNSUPDATE
update delete ora-lh.odyssey.com A
update add ora-lh.odyssey.com 600 A $NEWIP
update add $4.$3.$2.$1.in-addr.arpa 600 PTR ora-lh.odyssey.com.
```

```
ENDNSUPDATE
```

Configuring a Protection Group to Run a Script at Switchover or Takeover

After you have created a script, you must configure the protection group to run the script when a switchover or takeover occurs. If a switchover or takeover occurs, the script runs on the cluster that is becoming the new primary cluster.

▼ How to Configure a Protection Group to Run a Script at Switchover or Takeover

Before You Begin You must be assigned the Geo Management rights profile to complete this procedure. For more information, see "Securing Geographic Edition Software" in *Oracle Solaris Cluster 4.3 Geographic Edition Installation and Configuration Guide*.

1. **Log in to a cluster node.**

2. **Configure the `RoleChange_ActionCmd` and `RoleChange_ActionArgs` properties of the protection group.**

```
# geopg set-prop -p RoleChange_ActionCmd=fully-qualified-script \
-p RoleChange_ActionArgs=script-arguments  protection-group-name
```

-p *property-setting*

Specifies the properties of the protection group.

Specify the path to the command by using the `RoleChange_ActionCmd` property. This path should be valid on all nodes of all partner clusters that can host the protection group.

Define the arguments that you want to append to the command when the action command is run by using the `RoleChange_ActionArgs` property.

For more information about the properties you can set, see Appendix A, "Standard Geographic Edition Properties".

protection-group

Specifies the name of the protection group.

Example 33 Configuring a Protection Group to Run a Command at Cluster Switchover or Takeover

This example configures a protection group to run a custom command called `newDNS`.

```
# geopg set-prop -p RoleChange_ActionCmd=/usr/bin/newDNS \
-p RoleChange_ActionArgs=domain=companyx.com,ip=1.2.3.4 avspg
```

♦ ♦ ♦ CHAPTER 12

Script-Based Plug-Ins

This chapter provides information about Geographic Edition script-based plug-ins. It covers the following topics:

Introduction to Geographic Edition Script-Based Plug-Ins

The Geographic Edition framework provides modules to support data replication software products including the Availability Suite feature of Oracle Solaris, Oracle Data Guard, Oracle ZFS Storage Appliance, Oracle MySQL, Hitachi TrueCopy and Universal Replicator, and EMC SRDF. However, the creation of such modules requires detailed knowledge of both the replication software and the internals of the Geographic Edition product. The Geographic Edition framework uses the common agent container with a number of Java management beans (MBeans) that form the interface for the Geographic Edition monitoring and management infrastructure and the replication control software.

By providing a more generic interface module analogous to the Oracle Solaris Cluster Generic Data Service (GDS), the Geographic Edition script-based plug-in enables you to rapidly integrate additional replication technologies by supplying a few interface scripts to fulfill the necessary control functions. This capability frees you from needing to learn the internals of Geographic Edition or needing any knowledge of Java technology or MBeans. Instead, you can focus on the replication technology you need to protect your enterprise data. For more information, see *Oracle Solaris Cluster Generic Data Service (GDS) Guide*.

For simplicity, the term "script" is used throughout this document to represent any compiled binary or script-based executable.

This section provides the following information:

Advantages and Disadvantages of Using Script-Based Plug-Ins

The main advantage of using the script-based plug-in comes from reducing the barriers to implementing new replication mechanisms. Rather than spending time learning about the Java, JMX, MBeans, or common agent container technologies, you can focus on the critical logic needed, for example, to set up a replicated configuration or change the direction of the replication flow.

The disadvantage of this approach stems from the very generic nature of the plug-in that makes it so easy to use. Generic plug-ins lack some of the tight integration that a custom module can offer. For example, the arguments that you supply on the command line to a script-based plug-in configuration are at the script argument level rather than the highly specific replication variable level. So, for example, whereas the Geographic Edition Oracle Data Guard module has separate, specific arguments for properties like `standby_type` and `replication_mode`, an equivalent script-based plug-in version would pass these properties and their value as part of a single anonymous bundle to a script. The script would need to determine the arguments and whether each argument is valid.

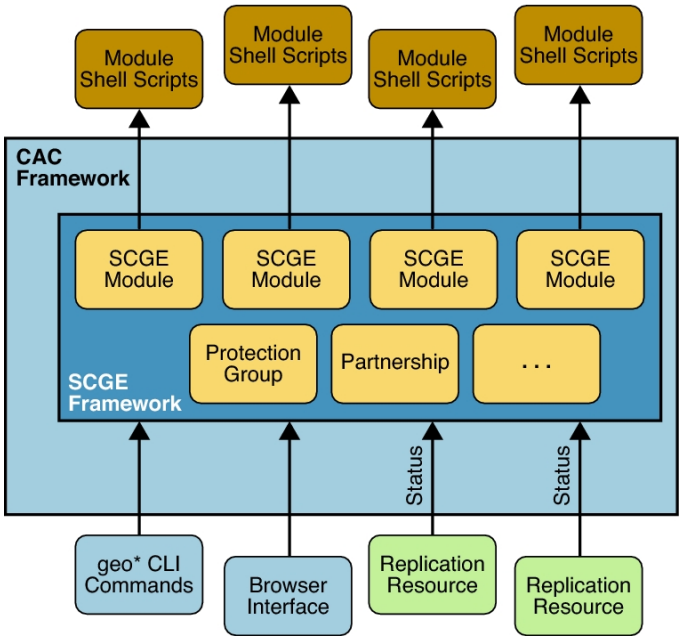
Script-Based Plug-In Architecture

Unlike other replication modules, the script-based plug-in is generic and capable of supporting a wide range of replication technologies. Consequently, the script-based plug-in does not contain a specific set of scripts to control a particular piece of replication software. Instead, it provides a framework for integrating a set of scripts or programs that you, the developer, write, and that a system administrator will later use.

This flexibility means that the script-based plug-in cannot directly enforce the inclusion or exclusion of application resource groups in a protection group. Furthermore, the script-based plug-in cannot even restrict the node lists of these entities, nor the relationship with the replication resource group that contains the replication resource needed to supply the replication status, or indeed any other resource group that is required.

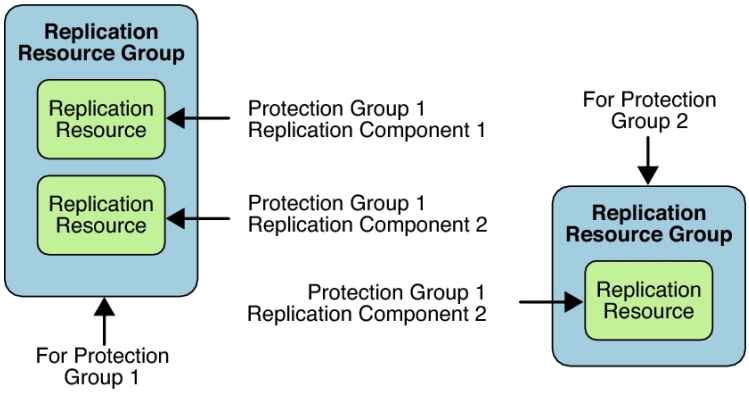
The following figure outlines the relationships between the various components within the Geographic Edition system. Commands issued through the command-line interface (CLI) or the Oracle Solaris Cluster Manager browser interface call the Geographic Edition modules through their relevant common agent container modules. These modules then call out to shell scripts to perform specific tasks. Once a protection group has been instantiated, the replication resource, representing a particular replicated object entity, reports its status back to the module through the event framework. This process enables the overall replication status to be reflected in the Geographic Edition output on the command line or in the browser interface.

FIGURE 2 Script-Based Plug-Ins Framework



The script-based plug-in developer therefore is free to govern the relationships between any or all of these entities: application resource group, data replication resource group, and replication status resource group. As the following figure shows, the only constraints are the requirements to have a named replication resource group per protection group and a named replication resource per device group or replicated component.

FIGURE 3 Script-Based Plug-In Replication Resource Group



The consequence of these requirements is that the administrator must provide a script-based plug-in configuration file for each protection group that is accessible from all cluster nodes and that details which nodes pertain to each script-based plug-in configuration. The purpose of this configuration file is to ensure that any subsequent developer-written scripts are called on one or more nodes on which the service is present.

In addition to the standard protection properties, the script-based plug-in enables the developer to name one or more scripts to perform the actions required by the Geographic Edition framework. These actions fall into two separate groups: those actions that operate at a per protection level and those actions that operate at a per replicated component level.

Restrictions of Script-Based Plug-Ins

There are no inherent restrictions regarding what you can do when creating script-based plug-in modules. However, using the script-based plug-in does not enable you to circumvent or overcome any inherent limitations present in the replication technology you intend to use.

Creating Script-Based Plug-Ins

The scripts can be written using an integrated development environment (IDE), such as the NetBeans IDE. For more information about the NetBeans IDE, see NetBeans IDE (`https://netbeans.org/`).

Property Descriptions for Script-Based Plug-Ins

This section contains the following information:

Protection Group Properties Overview

Table 7, "Protection Group Global Policies," on page 155 lists the protection group properties, along with a brief description, type of property, and default value for each property.

The scripts named by the developer in these properties can reference independent executables, a single common executable, or a combination of the two. No restrictions are placed on the language used to implement these scripts with the exception that the scripts must be able to run by root, from the command line, without a graphical display, and they must return either a zero (success) or nonzero (failure) exit code. The script-based plug-in Mbean returns any error code resulting from a failure. For more information, see Appendix E, "Error Return Codes for Script-Based Plug-Ins".

Protection groups that use script-based plug-in replication have the global properties provided in the following table. Note that all of these properties are tunable when you are offline.

TABLE 7 Protection Group Global Policies

Property Name	Description	Type	Default Value
`add_app_rg_args`	The arguments that are provided to the script, `add_app_rg_script`	Optional	Not applicable
`add_app_rg_script`	The script used to validate and perform tasks relevant for adding an application resource group to a protection group	Required	`/bin/true`
`configuration_file`	The per protection group script-based plug-in configuration file containing details of the nodes pertinent to script-based plug-in replicated components held in the protection group	Required	`/etc/opt/SUNWscgrepsbp/configuration`
`create_config_script`	The script used to create, modify, and validate a script-based plug-in replicated component instance	Required	`/bin/false`
`remove_app_rg_args`	The arguments that are provided to the script, `remove_app_rg_script`	Optional	Not applicable

`remove_app_rg_script`	The script used to validate and perform tasks relevant for removing an application resource group from a protection group	Required	`/bin/true`
`remove_config_script`	The script used to remove a script-based plug-in replicated component instance	Required	`/bin/true`
`start_replication_script`	The script used to start the data replication for a script-based plug-in replicated component instance	Required	`/bin/true`
`stop_replication_script`	The script used to stop the data replication for a script-based plug-in replicated component instance	Required	`/bin/true`
`switchover_script`	The script used to switch over the data replication direction for a script-based plug-in replicated component instance	Required	`/bin/true`
`takeover_script`	The script used to take over the data replication for a script-based plug-in replicated component instance	Required	`/bin/true`

"Protection Group Property Descriptions" on page 157 describes in detail the actions that each script and its associated arguments should perform when called by the script-based plug-in Mbean. "Standardized Script Command-Line Arguments" on page 167 explains how scripts can discriminate between the steps being performed.

Replication Component Properties Overview

Each replication component added to a particular protection group uses the scripts named in "Protection Group Properties Overview" on page 155. Individual replications distinguish themselves by varying the properties passed to these scripts.

The script-based plug-in module provides for two site-specific password properties:

- A local service password property (`local_service_password`)
- A remote service password property (`remote_service_password`)

These properties enable administrators of a script-based plug-in deployment to supply passwords to log in to services or remote systems without having to provide these passwords at switchover or takeover time. For more information, see "How Geographic Edition Handles Password Properties" on page 169.

The script-based plug-in module requires the developer to provide a property naming the replication resource contained in the replication resource group that holds the status of the replication.

Replication components in script-based plug-in protection groups have the optional properties provided in the following table. Note that all of these properties are tunable when you are offline.

TABLE 8 Optional Replication Component Properties

Property Name	Description	Type
`create_config_args`	The arguments passed to the script named by the `create_config_script` protection group property	Global
`local_service_password`	A password that might be needed by the scripts to perform some function on the local system that requires the entry of a password	Local
`remote_service_password`	A password that might be needed by the scripts to perform some function on the remote system that requires the entry of a password	Local
`remove_config_args`	The arguments passed to the script named by the `remove_config_script` protection group property	Global
`start_replication_args`	The arguments passed to the script named by the `start_replication_script` protection group property	Global
`stop_replication_args`	The arguments passed to the script named by the `stop_replication_script` protection group property	Global
`switchover_args`	The arguments passed to the script named by the `switchover_script` protection group property	Global
`takeover_args`	The arguments passed to the script named by the `takeover_script` protection group property	Global

Protection Group Property Descriptions

This section describes protection group properties:

`add_app_rg_script` Property

The script referenced by the `add_app_rg_script` property is responsible for checking that one or more application resource groups selected by the administrator are suitable for addition to the protection group. These checks might require that certain resource types be present or absent. Furthermore, the script must also set up any resource group affinities or dependencies within the confines of what is allowed by Geographic Edition. These affinities or dependencies are needed for the application resource group to produce the correct behavior.

Application resource groups must be in the unmanaged state when they are added to the configuration.

The `add_app_rg_script` is called at other points within the protection group life cycle, not just on the addition of application resource groups, to ensure that application resource groups continue to conform to the required rules. Write the script to ensure that these rules are met at all times.

Resource groups are offline and unmanaged on the standby site so certain application resource groups that represent services with embedded data replication might be unsuitable for addition to the protection group directly. An example is database data replication such as MySQL and Oracle RAC. The `add_app_rg_script` script must accommodate such validation.

The script must also be able to validate the `add_app_rg_args` property supplied to it with the `validate_parameters=true` option without actually performing any of the steps associated with this task. This operation is called only at the time of protection group update and creation, as opposed to at the time of device group update, modification, or validation.

When executed with `validate_parameters=false`, the script must perform any task required to add the resource groups listed in the final comma-separated `rgList` parameter value. These actions might include altering one or more of these resource group properties. The script is called on the local cluster to where the `geopg add-resource-group` command is run and called asynchronously on the remote cluster in response to the internal application resource group table being updated.

For example, if `add_app_rg_script = /var/tmp/addRGs` and `add_app_rg_args = -u root -d /mydir`, the resulting command looks like the following example:

```
# /var/tmp/addRGs -u root -d /mydir function=add_application_rgs \
validate_parameters={true|false} \
currentRole={PRIMARY|SECONDARY} pg=protection-group \
rgList=resource-group1,resource-group2,resource-group3,…
```

where the `rgList` parameter is the comma-separated list of application resource groups that the administrator has opted to add. The script is not responsible for creating these resource groups. Instead, the resource groups must already exist on both clusters. Furthermore, these resource groups must have the `auto_start_on_new_cluster` property set to `false`.

The function name for this step is `add_application_rgs`.

`configuration_file` Property

The `configuration_file` property specifies the file name of the configuration file used to drive the execution of replicated component-level scripts described in “Plug-In Script Functional Requirements” on page 166. Because individual script-based plug-ins inside a protection group might be on disjoint node sets or individual nodes, you should call the user scripts only

on the appropriate cluster node or nodes. For more information, see “Plug-In Script Functional Requirements” on page 166.

The configuration file must exist on all cluster nodes on both the primary and standby clusters. The script-based plug-in module tries to read the file from each node in turn until it finds a readable copy, but makes no effort to determine whether all copies are identical.

The format of the configuration file is as follows:

script-based-plug-in-configuration-name|*nodes-that-must-succeed-running-script*|*comma-separated-node-list*

For example:

```
foo.com|any|phys-node1,phys-node2
bar.com|all|phys-node1,phys-node3
baz.com|any|phys-node4
boo|any|phys-node4
biff|all|phys-node2
```

The script-based plug-in configuration name field must match the name of the replicated component being added to the protection group through the `geopg add-device-group` command.

For `foo.com`, a particular function step is tried on `phys-node1` and then, if it fails, on `phys-node2`. The function step can succeed on either node. This configuration assumes that the service is a multinode service like Oracle RAC.

For `bar.com`, a particular function step must succeed on both `phys-node1` and `phys-node3` for the step to complete. Again, this configuration is relevant only to multinode services like Oracle RAC. This function step enables a script to perform a task on multiple nodes without needing to connect to a remote node using `rsh` or `ssh` between the nodes.

`create_config_script` Property

The script referenced by the `create_config_script` property is responsible for creating, modifying, and validating a script-based plug-in configuration. The script must be able to validate the `create_config_args` property supplied to it with the `validate_parameters=true` option without actually performing the configuration creation.

When executed with `validate_parameters=false`, the script must create a replication group and an associated replication resource for the particular script-based plug-in. There must be only one replication resource group per script-based plug-in protection group and only one replication resource per replicated component. For example, a configuration with two script-

based plug-in protection groups (`hr-pg` and `sales-pg`), each with two replicated components (`hr-west` and `hr-east` for `hr-pg`, and `sales-north` and `sales-south` for `sales-pg`), would have two resource groups (`hr_pg_rep-rg` and `sales_pg-rep-rg`). These resource groups would then have the following two resources:

- `hr_west-rep-rs` and `hr_east-rep-rs` in `hr_pg-rep-rg`
- `sales_north-rep-rs` and `sales_south-rep-rs` in `hr_pg-rep-rg`

When creating the second replicated component or validating either configuration, the script must handle the case where the resource group already exists.

On completion, the script must write the resource group name and resource to standard output. This task is checked by the script-based plug-in framework to both validate that the objects exist and to set up the appropriate notification handling for state change events. The format for the output is as follows:

reprg=*replication-resource-group-name*
reprs=*replication-resource-name*

For example, for the case where the replication resource group is called `hr_pg-rep-rg` and the replication resource is called `hr_west-rep-rs`, the output would be as follows:

```
reprg=hr_pg-rep-rg
reprs=hr_west-rep-rs
```

The script must also write a list of resource groups to standard output that it has either created, or that exist already, or that it considers internal to the protection group. The format of the output must be as follows, with a carriage return at the end of the line:

rglist=*comma-separated-list-of-resource-groups*

For example, for the case where `foo-rg` and `bar-rg` are internal, the output would be as follows:

```
rglist=foo-rg,bar-rg
```

If no resource groups exist, the output would be as follows:

```
rglist=
```

Examples of such internal resource groups are the lightweight resource groups in the Availability Suite module or the shadow RAC proxy server resource groups in the Oracle Data Guard module.

This script is called for each script-based plug-in created in any specific protection group because `create_config_script` is a global protection group property. For example, if a protection group has script-based plug-in configurations `foobar.com` and `baz.com`,

the `create_config_script` script is called once when `foobar.com` is added with the `create_config_args` property given for the `foobar.com` property. The script is later called for `baz.com` when it is added to the protection group with the `baz.com` `create_config_args` property value. This process results in a replication resource group with two resources: one resource monitoring `foobar.com` replication and the other resource monitoring `baz.com`.

If the protection group is known to both the primary and standby sites, then adding the script-based plug-in configuration to the protection group will cause the `create_config_script` script to be executed on the site that the `geopg` command is run from and then on the remote site as a result of the internal Oracle Solaris Cluster Geographic protection group table transfer. The latter step happens asynchronously.

The `create_config_script` script is called with the `create_config_args` property followed by the standard command-line arguments and an additional `isModify` parameter. This parameter is set to `false` when the command has been called as a result of a `geopg create-device-group` or `geopg validate pg`command. This parameter is set to `true` when the command has been called as a result of a `geopg modify-device-group` command.

For example, if `create_config_script = /var/tmp/add` and `create_config_args = "-u root -d /mydir"`, the resulting command looks like the following example:

`/var/tmp/add -u root -d /mydir function=create_configuration \`
`validate_parameters={true|false} currentRole={PRIMARY|SECONDARY}`
`pg=`*protection-group* **`isModify={true|false}`**

The function name for this step is `create_configuration`.

`remove_app_rg_script` Property

The script referenced by the `remove_app_rg_script` property is responsible for removing one or more application resource groups, selected by the administrator, from the protection group. A comma-separated list of resource groups to remove is passed to the script through the `rgList` parameter. The script is called on the local cluster to where the `geopg remove-resource-group` command is run and called asynchronously on the remote cluster in response to the internal application resource group table being updated.

The script must also be able to validate the `remove_app_rg_args` property supplied to it with the `validate_parameters=true` option without actually performing any of the steps associated with this task. This operation is called only at the time of protection group update and creation, as opposed to at the time of device group update, modification, or validation.

For example, if `remove_app_rg_script = /var/tmp/removeRGs` and `remove_app_rg_args = "-u root -d /mydir"`, the resulting command looks like the following example:

```
#/var/tmp/removeRGs -u root -d /mydir\
function=remove_application_rgs \
validate_parameters={true|false} \
currentRole={PRIMARY|SECONDARY} pg=protection-group\
rgList=resource-group1,resource-group2,resource-group3,...
```

where the `rgList` parameter is the comma-separated list of application resource groups that the administrator has opted to remove. The script is not responsible for removing these resource groups, only for making the necessary changes to their properties that might be required as a result of removing them from Geographic Edition protection group control.

The function name for this step is `remove_application_rgs`.

`remove_config_script` Property

The script referenced by the `remove_config_script` property is responsible for reversing the work of the `create_config_script` script. The script must be able to validate the `remove_config_args` property supplied to it with the `validate_parameters=true` option without actually performing the configuration removal.

When executed with `validate_parameters=false`, the script must remove the replication resource (originally named by the `create_config_script` script `reprs=` output for the specific script-based plug-in) from the replication resource group given by the `create_config_script` script `reprg=` output. If the resource is the last in the resource group, the script must also remove the resource group.

For example, if `remove_config_script = /var/tmp/remove and remove_config_args = "-u root -d /mydir"`, the resulting command looks like the following example:

```
# /var/tmp/remove -u root -d /mydir function=remove_configuration \
validate_parameters={true|false} \
currentRole={PRIMARY|SECONDARY} pg=protection-group
```

The function name for this step is `remove_configuration`.

`start_replication_script` Property

The script referenced by the `start_replication_script` property is responsible for starting the data replication process and enabling the replication resource that is used to monitor the replication. The script must also be able to validate the `start_replication_args` property supplied to it with the `validate_parameters=true` option without actually starting the data replication.

When executed with `validate_parameters=false`, the script must start the actual data replication and enable the replication resource that is used to monitor the replication.

For example, if `start_replication_script = /var/tmp/start` and `start_replication_args ="-u root -d /mydir"`, the resulting command looks like the following example:

```
# /var/tmp/start -u root -d /mydir function=start_replication \
validate_parameters={true|false} \
currentRole={PRIMARY|SECONDARY} pg=protection-group
```

The `start_replication_script` script is called on one or both clusters depending on which of the following commands the administrator specifies:

For local clusters only:

```
# geopg start -e local protection-group      # local cluster only
```

For both clusters:

```
# geopg start -e global protection-group     # both clusters
```

The function name for this step is `start_replication`.

`stop_replication_script` Property

The script referenced by the `stop_replication_script` property is responsible for stopping the data replication process and disabling the replication resource that is used to monitor the replication. The script must also be able to validate the `stop_replication_args` property supplied to it with the `validate_parameters=true` option without actually starting the data replication.

When executed with `validate_parameters=false`, the script must stop the actual data replication and disable the replication resource that is used to monitor the replication.

For example, if `stop_replication_script = /var/tmp/stop` and `stop_replication_args = "-u root -d /mydir"`, the resulting command looks like the following example:

```
# /var/tmp/stop -u root -d /mydir function=start_replication \
validate_parameters={true|false} \
currentRole={PRIMARY|SECONDARY} pg=protection-group
```

The `stop_replication_script` script is called on one or both clusters depending on which of the following commands the administrator specifies:

For local cluster only:

```
# geopg stop -e local protection-group      # local cluster only
```

For both clusters:

```
# geopg stop -e global protection-group     # both clusters
```

The function name for this step is `stop_replication`.

`switchover_script` Property

The script referenced by the `switchover_script` property is responsible for two functions:

- Checking that the service is in a position to switch over
- Performing the actual data replication switchover

The second step is performed only if the first step is completed successfully, meaning that the step exits with a zero exit code. In each case, the script is called on both clusters.

The `switchover_script` script is first called on the cluster on which the `geopg switchover`command is executed. Subsequent changes in Geographic Edition status trigger an event on the remote cluster, causing the script to be executed asynchronously on that cluster, too. The arguments for the two calls are different.

For example:

If `switchover_script = /var/tmp/switchover` and `switchover_args = "-u root -d /mydir"`, the resulting command looks like the following example:

```
# /var/tmp/switchover -u root -d /mydir function=check_switchover \
validate_parameters=false currentRole={PRIMARY|SECONDARY} \
pg=protection-group newRole={PRIMARY|SECONDARY}
```

If that step succeeds:

```
# /var/tmp/switchover -u root -d /mydir \
function=perform_switchover \
validate_parameters=false \
currentRole={PRIMARY|SECONDARY} pg=protection-group \
newRole={PRIMARY|SECONDARY}
```

The argument `newRole` is the target role of the cluster after a successful switchover.

The function names for these steps are `check_switchover` and `perform_switchover` and just `switchover` for the `validate_parameter` step, which is called as follows:

```
# developer-switchover-program developer-switchover-program-arguments \
```

```
function=switchover validate_parameters=true \
currentRole={PRIMARY|SECONDARY} pg=protection-group
```

`takeover_script` Property

The script referenced by the `takeover_script` property is responsible for two functions:

- Checking that the service is in a position to be taken over
- Performing the actual data replication takeover

The second step is performed only if the first step is completed successfully, meaning that the step exits with a zero exit code. In each case, the script is called on both clusters. If the original primary cluster is available, the protection group is deactivated on that cluster. Deactivation involves stopping the application resource groups.

The `takeover_script` script must be called on the standby cluster by executing the `geopg takeover` command on that cluster. The arguments for the two calls are different.

For example, if `takeover_script = /var/tmp/switchover` and `takeover_args = "-u root -d /mydir"`, the resulting command looks like the following example:

```
# /var/tmp/switchover -u root -d /mydir function=check_takeover \
validate_parameters=false currentRole={PRIMARY|SECONDARY} \
pg=protection-group newRole={PRIMARY|SECONDARY}
```

Then, if that step succeeds:

```
# /var/tmp/switchover -u root -d /mydir function=perform_takeover \
validate_parameters=false currentRole={PRIMARY|SECONDARY} \
pg=protection-group newRole={PRIMARY|SECONDARY}
```

The argument `newRole` is the target role of the cluster after a successful takeover.

The function names for these steps are `check_takeover` and `perform_takeover` and just `takeover` for the `validate_parameter` step, which is called as follows:

```
# developer-takeover-program developer-takeover-program-arguments \
function=takeover validate_parameters=true \
currentRole={PRIMARY|SECONDARY} pg=protection-group
```

Internals for Script-Based Plug-Ins

This section describes the internals for the script-based plug-ins. It covers the following topics:

Plug-In Script Functional Requirements

A protection group has several global properties that are valid and relevant to both the primary and secondary clusters, and by extension all cluster nodes. Additionally, each replicated component has a set of local and global properties. Together, these properties describe and control the replication pertaining to one or more replicated services.

This section describes the following topics:

Plug-In Script Argument Validation

Each script provided in one of the protection group properties must be capable of validating the arguments with which it has been called in order to determine whether the arguments are complete and acceptable. Validation ensures that scripts such as `switchover_script` and `takeover_script`, that are not called regularly, do not fail because their arguments have become incompatible. Failing to validate the arguments could lead to the inability to switch over or take over in an emergency.

Scripts must therefore be able to validate the arguments defined by the administrator through the Oracle Solaris Cluster Manager browser interface or command-line interface (CLI), and issue a return code of zero, if they are correct. The script must not perform its real function at this stage, for example, to switch over, take over, or create a script-based plug-in configuration. If you do not want to perform these checks, the script must still return without performing any additional work in response to the validate arguments call.

The validate arguments step is denoted by the Geographic Edition script-based plug-in Mbean passing `validate_parameters=true` as one of the command-line arguments. When a script-based plug-in replication component is added to a protection group, all the replicated component-specific scripts listed in "Protection Group Properties Overview" on page 155 are called on to validate their arguments. This call is made on one or more nodes per

cluster depending on the particular script-based plug-in replicated component configuration as defined in the configuration file. For more information, see “configuration_file Property” on page 158 and “Protection Group Properties Overview” on page 155.

The same validation calls are made under the following circumstances:

- When the replication component is modified because the modification might result in program argument changes
- When there are protection group validation calls in response to the `geopg validate` *protection-group* command
- When the Geographic Edition framework is starting and re-creating the initial script-based plug-in replicated component objects that are stored in the Cluster Configuration Repository (CCR)

Two protection group-level program properties, `add_app_rg_script` and `remove_app_rg_script`, have associated protection group argument properties.

Standardized Script Command-Line Arguments

All scripts are called using a standardized command structure. The format of the command is as follows:

```
# developer-program-name administrator-supplied-program-arguments \
function=step-name \
validate_parameters={true|false} \
currentRole={PRIMARY|SECONDARY} \
pg=protection-group \
additional-function-dependent-arguments
```

where *developer-program-name* is the name of one of the externally developed scripts and *administrator-supplied-program-arguments* provides the arguments given for this script by the administrator when setting up a script-based plug-in configuration.

The use of the `function=`*step-name* argument enables scripts to determine what action they are being called on to perform. This function is especially important if a single script has been written to perform one or more tasks. Two scripts in particular need to be concerned with this argument: `switchover_script` and `takeover_script`.

The `currentRole` argument indicates the current role of the local cluster, while the `pg` argument denotes the name of the protection group containing the script-based plug-in configuration. Scripts should be prepared to deal with values in either uppercase or lowercase. The same is true of the `newRole` argument for `switchover_script` and `takeover_script`.

All scripts, if successful, must return a zero exit code. On failure, all scripts must return a nonzero exit code and generate a localized error message on standard error (`stderr`). Any output sent to standard output (`stdout`) is generally ignored (with the exception of `create_config_script`), unless common agent container logging is turned on. In that case, the output is saved in the `/var/cacao/instances/default/logs/cacao.0` log file, along with other common agent container debugging information. Do not save debugging information as a matter of course because the volume of output can be substantial.

Script-Based Plug-In Replication Resource Groups and Resources

The name of the replication resource group for a particular protection group is defined by the value returned by `create_config_script` in the `reprg=` string sent to standard output. This string contains one or more replication resources referenced by individual replication resources named by `create_config_script` in the `reprs=` string sent to standard output. For any one protection group, the value returned by `create_config_script` must be identical.

The function of the replication resources is to monitor the state of the replication associated with the resource and thus the replicated component. The replication resource status, which is set by a probe method, is used to determine the overall status of the protection group. The start and stop methods of the replication resource do not start and stop the actual data replication.

The replication resource must be enabled and disabled by `start_replication_script` and `stop_replication_script`.

Protection Group Status Mapped from Replication Resource Status

The protection group status reflects the aggregated status of all replication resources in the replication resource group created by the developer-written `create_config_script` program.

The following table illustrates the mapping from the status of each replication resource to the protection group status. An X represents any possible status for the resource and demonstrates that the most restrictive status governs the overall status of the protection group.

Unknown	Faulted	Degraded	Online	Protection Group Status

True	X	X	X	UNKNOWN
False	True	X	X	FAULTED
False	False	True	X	DEGRADED
False	False	False	True	ONLINE

How Geographic Edition Handles Password Properties

This section describes the mechanism by which Geographic Edition handles password properties, when the entity added to a protection group (for example, an Oracle Data Guard or script-based plug-in configuration) requires a password property.

The password properties are read during the execution of the `geopg` command. These password properties are recognized by their conformance to the pattern `*_password`. When `geopgi` (a back-end program called by the `geopg` command) parses the protection group properties list, it looks for such arguments. If the password has been supplied in cleartext, as shown in the following example, then `geopg` warns the user that the password is insecure but continues processing the password.

… `-p sysdba_password=`*password* …

For any password properties that have been specified, the `geopgi` program enters non-echo mode and prompts for these passwords, as shown in the following example:

```
… -p local_service_password= -p remote_service_password= …
```

Once all the arguments have been processed, these pairs are written into an internal password file on the local node, which is root readable only. A separate `internalPasswordFile` argument is inserted into the properties list with the value *hostname*:*filename*.

Once in the core Geographic Edition Java code, the `internalPasswordFile` argument is unpacked, and the file is read remotely through an internal common agent container to common agent container call. For security, the passwords are then converted into the hexadecimal representation of their character codes before they are written to the Oracle Solaris Cluster CCR, if the rest of the properties are correct and complete, and the validation succeeds.

When required, the passwords can be queried and converted back from the CCR and supplied to the appropriate programs to achieve the relevant switchovers, takeovers, or status queries.

♦ ♦ ♦ APPENDIX A

Standard Geographic Edition Properties

This appendix provides the standard properties of Geographic Edition heartbeats, heartbeat plug-in, partnerships, protection groups and data replication device groups, sites, and multigroups.

This appendix contains the following sections:

Note - The property names and values, such as `Query_interval`, `True`, and `False`, are *not* case sensitive.

Heartbeat Properties

The following table describes the heartbeat properties that the Geographic Edition framework defines.

TABLE 9 Geographic Edition Heartbeat Properties

Property Name	Description
`Query_interval` (integer)	Specifies the delay in seconds between heartbeat status requests. **Tuning recommendations**: The value of this property is assigned at creation and can be tuned at runtime. **Category**: Optional **Default**: 120 seconds

Heartbeat Plug-in Properties

The following table describes the heartbeat plug-in properties that the Geographic Edition framework defines. To find out how to modify heartbeat plug-in properties, see "How to Modify the Properties of a Heartbeat Plug-in" on page 61.

TABLE 10 Geographic Edition Heartbeat Plug-in Properties

Property	Description
`Plugin_properties` (string)	Specifies a property string specific to the plug-in. **Tuning recommendations**: The value of this property is assigned at creation and can be tuned at runtime. **Category**: Optional **Default**: None except for heartbeats that use the default heartbeat plug-ins, `tcp_udp_plugin` and `ping-plugin`. For the `tcp_udp_plugin` plug-in, the format of this string is predefined as *remote-IP-address*`/UDP/2084/ipsec`, *remote-IP-address*`/TCP/2084/ipsec`. The *remote-IP-address* argument specifies the IP address of the partner cluster. The optional `ipsec` argument specifies whether the plug-in uses IPsec with a Boolean value of `true` or `false`. For the `ping-plugin`, the format of this string is predefined as *remote-IP-address*, where *remote-IP-address* specifies the IP address of the partner cluster.
`Query_cmd` (string)	Specifies the path to the heartbeat status request command. **Tuning recommendations**: The value of this property is assigned at creation and can be tuned at runtime. **Category**: Required property if the plug-in does not specify a predefined plug-in. **Default**: None
`Requester_agent` (string)	Specifies the absolute path to the requester agent. **Tuning recommendations**: The value of this property is assigned at creation and can be tuned at runtime. However, the `Requester_agent` property of the default plug-in should never need to be tuned except for testing purposes. **Category**: Optional **Default**: None
`Responder_agent` (string)	Specifies the absolute path to the responder agent. **Tuning recommendations**: The value is assigned at creation and can be tuned at runtime. However, the `Responder_agent` property of the default plug-in should never need to be tuned except for testing purposes.

Property	Description
	Category: Optional **Default**: None
`Type` (enum)	Designates the type of plug-in. Set to either `Primary` or `Backup`. **Tuning recommendations**: The value of this property is assigned at creation and can be tuned at runtime. **Category**: Required **Default**: None, except for the default heartbeat that is named `ping_plugin`. If using this plug-in, the default value is `backup`.

Partnership Properties

The following table describes the partnership properties that the Geographic Edition framework defines. To find out how to modify a property of a partnership, see "How to Modify Partnership Properties" on page 41.

TABLE 11 Geographic Edition Partnership Properties

Property	Description
`Description` (string)	Describes the partnership. **Tuning recommendations**: The value of this property is assigned at creation and can be tuned at runtime. **Category**: Optional **Default**: Empty string
`Notification_ActionCmd` (string)	Provides the path to the action script that is triggered when heartbeat-loss notification is issued. **Tuning recommendations**: The value of this property is assigned at creation and can be tuned at runtime. **Category**: Optional **Default**: Empty string
`Notification_EmailAddrs` (string array)	Lists the email addresses that are sent email when heartbeat-loss notification is issued. The list is comma delimited. **Tuning recommendations**: The value of this property is assigned at creation and can be tuned at runtime. **Category**: Optional **Default**: Empty string

Protection Group Properties

The following table describes the protection group properties that the Geographic Edition framework defines. To find out how to modify protection group properties, see "How to Modify a Protection Group" on page 86.

TABLE 12 Geographic Edition Protection Group Properties

Property	Description
`Description` (string)	Describes the protection group. **Tuning recommendations**: This property can be tuned at any time. **Category**: Optional **Default**: Empty string
`External_Dependency_Allowed` (Boolean)	Allow dependencies between resource groups and resources that belong to this protection group and resource groups and resources that do not belong to this protection group when set to `true`. **Tuning recommendations**: This property can be tuned at any time. **Category**: Optional **Default**: `false`
`RoleChange_ActionArgs` (string)	Defines a string of arguments that are appended to the end of the command line when the role-change action command, `RoleChange_ActionCmd`, is run. **Tuning recommendations**: This property can be tuned at any time. **Category**: Optional **Default**: Empty string
`RoleChange_ActionCmd` (string)	Specifies the path to an executable command. This script is invoked during a switchover or takeover on the new primary cluster when the protection group is started on the new primary cluster. The script is invoked on the new primary cluster after the data replication role changes from Secondary to Primary and before the application resource groups are brought online. If the data replication role change does not succeed, then the script is not called. This file should be valid on all nodes of all partner clusters that can host the protection group, and have execute permissions for the user that launches the script. **Tuning recommendations**: This property can be tuned at any time. **Category**: Optional **Default**: Empty string
`Timeout` (integer)	Specifies the timeout period for the protection group in seconds. The timeout period is the longest time Geographic Edition waits for a response after you

Property	Description
	run a `geopg` command, such as `geopg start`, `geopg stop`, `geopg switchover`, and `geopg takeover`. If the command does not respond within the timeout period, the Geographic Edition framework reports the operation as timed out, even if the underlying command eventually completes successfully. You should identify the amount of time required to perform a role-reversal of the data replication, and set the timeout value to 150% to 200% of that value to ensure enough time for the role-reversal to complete. To ensure that an operation has finished on the remote cluster, check system status after a timeout before attempting the operation again. For more information, see "Troubleshooting Migration Problems" on page 186. The timeout period applies to operations on a per-cluster basis. An operation with a local scope times out if the operation does not complete after the specified timeout period. An operation with a global scope consists of an action on the local cluster and an action on the remote cluster. The local and remote action are timed separately so that an operation with a global scope times out during one of the following conditions: ■ The local operation does not complete after the specified timeout period. ■ The remote operation does not complete after the specified timeout period. **Tuning recommendations**: This property can be tuned only when the protection group is offline. **Category**: Optional **Range**: 20-1000000 seconds **Default**: 200

Site Properties

The following table describes the site properties that the Geographic Edition framework defines.

TABLE 13 Geographic Edition Site Properties

Property	Description
`Description` (string)	Describes the site. The system sets this property on the local cluster, then propagates the value to the other clusters in the site. **Tuning recommendations**: This property is assigned at creation and tunable at runtime. **Category**: Optional

Property	Description
	Default: None
`joinTimeout` (integer)	Specifies, in seconds, the longest time that the JMX client, the site, waits for the invocation of an MBean-server method to return. If a method does not return by the end of the timeout period, the client moves to its next set of instructions and reports the operation as timed out. By default, a client waits indefinitely for a method to return. If the MBean server is unable to complete an invocation, the JMX client will hang indefinitely. The timeout period is site-wide and applies to operations on a per-cluster basis. The `joinTimeout` property is used when the join operation requires more time to complete than the default setting allows before the `geosite join`command times out. **Tuning recommendations**: This property is assigned at creation,but it is not tunable at runtime. **Category**: Optional **Default**: 30 seconds **Minimum**: 20 seconds **Maximum**: 3600 seconds
`Query_interval` (integer)	Specifies, in seconds, the frequency between heartbeat status requests used by the site clusters. The plug-in enters emergency mode if three `Query_interval` periods pass without response. The plug-in times out and enters error mode if a further `Query_interval` period passes with no response. **Tuning recommendations**: This property is assigned at creation and tunable at runtime. **Category**: Optional **Default**: 120 seconds **Minimum**: 20 seconds **Maximum**: 300 seconds
`Timeout` (integer)	Specifies, in seconds, the longest time that the JMX client, the site, waits for the invocation of an MBean-server method to return. If a method does not return by the end of the timeout period, the client moves to its next set of instructions and reports the operation as timed out. By default, a client waits indefinitely for a method to return. If the MBean server is unable to complete an invocation, the JMX client will hang indefinitely. The timeout period is site-wide and applies to operations on a per-cluster basis. **Tuning recommendations**: This property is assigned at creation and tunable at runtime. **Category**: Optional **Default**: 30 seconds **Minimum**: 30 seconds

Property	Description
	Maximum: 3600 seconds

Multigroup Properties

The following table describes the multigroup properties that the Geographic Edition framework defines.

TABLE 14 Geographic Edition Multigroup Properties

Property	Description
`Description` (string)	Describes the multigroup. The system sets this property on the local cluster, then propagates the value to the other clusters in the site where the multigroup is defined. **Tuning recommendations**: This property is assigned at creation and tunable at runtime. **Category**: Optional **Default**: None
`Timeout` (integer)	Specifies, in seconds, the longest time that the JMX client, the site, waits for the invocation of an MBean-server method to return. If a method does not return by the end of the timeout period, the client moves to its next set of instructions and reports the operation as timed out. By default, a client waits indefinitely for a method to return. If the MBean server is unable to complete an invocation, the JMX client will hang indefinitely. The timeout period is site-wide and applies to operations on a per-cluster basis. The timeout period applies to operations on a per-cluster basis. An operation with a local scope times out if the operation is not completed after the specified `Timeout` period. An operation with a global scope consists of an action on the local cluster and an action on the remote cluster. The local and remote actions are timed separately. So, an operation with a global scope times out if the local operation is not completed after the specified timeout period or if the remote operation is not completed after the specified timeout period. For example, the following command is started with a local scope: `#` **`geomg start -e local`** *multigroup* If you set the `Timeout` property to 3000 seconds, the `geomg start` command times out if the operation does not complete after 3000 seconds. You can start the same command with a global scope as follows: `#` **`geomg start -e global`** *multigroup*

Property	Description
	If the `Timeout` property is set to 3000 seconds, the `geomg start` command times out if the operation is not completed on the local cluster after 3000 seconds or if the operation is not completed on the remote cluster after 3000 seconds. If the local action takes 1500 seconds and the remote action takes 1500 seconds, the operation is not timed out. The multigroup timeout value is an estimated value that is applied to some sub-operations. The timeout value does not apply to the entire operation, so not every operation on a protection group is timed against the timeout period. For example, the time taken to initialize the data structure and to check for the precondition of the operation are not timed in the timeout period. **Tuning recommendations**: This property is assigned at creation and tunable at runtime. **Category**: Optional **Default**: 30 seconds **Minimum**: 20 seconds **Maximum**: 3600 seconds

♦ ♦ ♦ APPENDIX B

Disaster Recovery Administration Example

This appendix provides an example of a disaster recovery scenario and the actions an administrator might perform.

Example Company has two geographically separated clusters, `cluster-paris` in Paris and `cluster-newyork` in New York. These clusters are configured as partner clusters. The cluster in Paris is configured as the primary cluster and the cluster in New York is the secondary.

The `cluster-paris` cluster fails temporarily as a result of power outages during a windstorm. An administrator can expect the following events:

1. The heartbeat communication is lost between `cluster-paris` and `cluster-newyork`. Because heartbeat notification was configured during the creation of the partnership, a heartbeat-loss notification email is sent to the administrator.

   For information about the configuring partnerships and heartbeat notification, see "Modifying Partnership Properties" on page 41.
2. The administrator receives the notification email and follows the company procedure to verify that the disconnect occurred because of a situation that requires a takeover by the secondary cluster. Because a takeover might be consuming time, depending on the requirements of the applications being protected, Example Company does not allow takeovers unless the primary cluster cannot be repaired within two hours.

   For information about verifying a disconnect on a system, see "Detecting Cluster Failure" on page 131.
3. Because the `cluster-paris` cluster cannot be brought online again for at least another day, the administrator runs a `geopg takeover` command on a node in the cluster in New York. This command starts the protection group on the secondary cluster `cluster-newyork` in New York.

   For information about performing a takeover on a system, see "Forcing a Takeover of a Protection Group" on page 139.
4. After the takeover, the secondary cluster `cluster-newyork` becomes the new primary cluster. The failed cluster in Paris is still configured to be the primary cluster. Therefore, when the `cluster-paris` cluster restarts, the cluster detects that the primary cluster was

down and lost contact with the partner cluster. Then, the `cluster-paris` cluster enters an error state that requires administrative action to clear. You might also be required to recover and resynchronize data on the cluster.

For information about recovering data after a takeover, see the Geographic Edition manual for your data replication product.

♦ ♦ ♦ APPENDIX C

Post-Takeover Conditions

This appendix provides details about the state of the primary and secondary clusters after you run the `geopg takeover` command or the `geomg takeover` command.

This appendix contains the following sections:

Results of a Takeover When the Partner Cluster Can Be Reached

This section describes the activation state of the primary and secondary clusters before and after you run the `geopg takeover` command. The results described in this section assume that the partner cluster can be reached.

The following table describes the states of the clusters when you run the `geopg takeover` command on the secondary cluster, `cluster-newyork`.

TABLE 15 Takeover Results of Running the `geopg takeover` Command on the Secondary Cluster

Cluster Role and State Before Takeover	Cluster Role and State After Takeover
`cluster-paris`: primary, deactivated `cluster-newyork`: secondary, deactivated	`cluster-paris`: secondary, deactivated `cluster-newyork`: primary, deactivated
`cluster-paris`: primary, activated `cluster-newyork`: secondary, deactivated	`cluster-paris`: secondary, deactivated `cluster-newyork`: primary, deactivated
`cluster-paris`: primary, deactivated `cluster-newyork`: secondary, activated	`cluster-paris`: secondary, deactivated

Cluster Role and State Before Takeover	Cluster Role and State After Takeover
	`cluster-newyork`: primary, activated, with data replication stopped
`cluster-paris`: primary, activated `cluster-newyork`: secondary, activated	`cluster-paris`: secondary, deactivated `cluster-newyork`: primary, activated, with data replication stopped

The following table describes the states when you run the `geopg takeover` command on the primary cluster, `cluster-paris`.

TABLE 16 Takeover Results of Running the `geopg takeover` Command on the Primary Cluster

Cluster Role and State Before Takeover	Cluster Role and State After Takeover
`cluster-paris`: primary, deactivated `cluster-newyork`: secondary, deactivated	`cluster-paris`: primary, deactivated `cluster-newyork`: secondary, deactivated
`cluster-paris`: primary, activated `cluster-newyork`: secondary, deactivated	`cluster-paris`: primary, activated, with data replication stopped `cluster-newyork`: secondary, deactivated
`cluster-paris`: primary, deactivated `cluster-newyork`: secondary, activated	`cluster-paris`: primary, deactivated `cluster-newyork`: secondary, deactivated
`cluster-paris`: primary, activated `cluster-newyork`: secondary, activated	`cluster-paris`: primary, activated, with data replication stopped `cluster-newyork`: secondary, deactivated

Results of a Takeover When the Partner Cluster Cannot Be Reached

This section describes the activation state of the primary and secondary clusters before and after you run a `geopg takeover` command when the partner cluster cannot be reached or when the protection group on the partner cluster is busy.

The following table describes the states when you run the `geopg takeover` command on the secondary cluster, `cluster-newyork`, and the primary cluster cannot be reached or the protection group on the primary cluster is busy.

Note - The cluster role and state after the takeover, which is given in the table, is available only when the partner cluster can be reached again.

TABLE 17 Takeover Results of Running the `geopg takeover` Command on the Secondary Cluster When the Primary Cluster Cannot Be Reached

Cluster Role and State Before Takeover	Cluster Role and State After Takeover

`cluster-paris`: primary, deactivated, synchronization status `Unknown`
 	
`cluster-newyork`: secondary, deactivated, synchronization status `Unknown`
	`cluster-paris`: primary, deactivated, synchronization status `Error`
	
	`cluster-newyork`: primary, deactivated, synchronization status `Error`

`cluster-paris`: primary, activated, synchronization status `Unknown`
 	
`cluster-newyork`: secondary, deactivated, synchronization status `Unknown`
	`cluster-paris`: primary, activated, synchronization status `Error`
	
	`cluster-newyork`: primary, deactivated, synchronization status `Error`

`cluster-paris`: primary, deactivated, synchronization status `Unknown`
 	
`cluster-newyork`: secondary, activated, synchronization status `Unknown`
	`cluster-paris`: primary, deactivated, synchronization status `Error`
	
	`cluster-newyork`: primary, activated, with data replication stopped, synchronization status `Error`

`cluster-paris`: primary, activated, synchronization status `Unknown`
 	
`cluster-newyork`: secondary, activated, synchronization status `Unknown`
	`cluster-paris`: primary, activated, synchronization status `Error`
	
	`cluster-newyork`: primary, activated, with data replication stopped, synchronization status `Error`

The following table describes the states when you run the `geopg takeover` command on the primary cluster, `cluster-paris`, and the secondary cluster cannot be reached or the protection group on the secondary cluster is busy.

TABLE 18 Takeover Results of Running the `geopg takeover` Command on the Primary Cluster When the Secondary Cluster Cannot Be Reached

Cluster Role and State Before Takeover	Cluster Role and State After Takeover

`cluster-paris`: primary, deactivated, synchronization status `Unknown`
 	
`cluster-newyork`: secondary, deactivated, synchronization status `Unknown`
	`cluster-paris`: primary, deactivated, synchronization status `OK`, `Error`, or `Mismatch`
	
	`cluster-newyork`: secondary, deactivated, synchronization status `OK`, `Error`, or `Mismatch`

`cluster-paris`: primary, activated, synchronization status `Unknown`
 	
`cluster-newyork`: secondary, deactivated, synchronization status `Unknown`
	`cluster-paris`: primary, activated, with data replication stopped, synchronization status `OK`, `Error`, or `Mismatch`
	
	`cluster-newyork`: secondary, deactivated, synchronization status `OK`, `Error`, or `Mismatch`

`cluster-paris`: primary, deactivated, synchronization status `Unknown`
 	
`cluster-newyork`: secondary, activated, synchronization status `Unknown`
	`cluster-paris`: primary, deactivated, synchronization status `OK`, `Error`, or `Mismatch`
	
	`cluster-newyork`: secondary, activated, synchronization status `OK`, `Error`, or `Mismatch`

Cluster Role and State Before Takeover	Cluster Role and State After Takeover

`cluster-paris`: primary, activated, synchronization status `Unknown`
 	
`cluster-newyork`: secondary, activated, synchronization status `Unknown`
	`cluster-paris`: primary, activated, with data replication stopped, synchronization status `OK`, `Error`, or `Mismatch`
	
	`cluster-newyork`: secondary, activated, synchronization status `OK`, `Error`, or `Mismatch`

♦ ♦ ♦ APPENDIX D

Troubleshooting the Geographic Edition Framework

This appendix describes procedures for troubleshooting your application of the Geographic Edition framework.

This appendix contains the following sections:

Troubleshooting Monitoring and Logging

This section provides the following information about setting up logging and about problems that you might encounter with monitoring the Geographic Edition framework:

For information about logging, see "Viewing the Geographic Edition Log Messages" on page 126.

Configuring the Logger File to Avoid Too Many Traces

Configure the logger file, `/etc/cacao/instances/default/private/logger.properties`, as follows depending on the `cmass` messages you want logged:

- To select only `WARNING` and `SEVERE` messages, the first line of the file should read as follows:

  ```
  com.sun.cluster.level=WARNING
  ```

- To enable all `geocontrol` messages, the second line of the file should read as follows:

  ```
  com.sun.cluster.agent.geocontrol.level=ALL
  ```

The enabled traces are copied to the `/var/cacao/instances/default/logs/cacao.0` file.

Configuring the Logger File to Avoid Detailed Messages From the `gcr` Agent

If you want to avoid too detailed messages in your log file from the `gcr` agent, use entries similar to the following in your `/etc/cacao/instances/default/private/logger.properties` logger file:

```
com.sun.cluster.level=WARNING
com.sun.cluster.agent.geocontrol.gcr.level=INFO
com.sun.cluster.agent.geocontrol.level=ALL
```

This property file is updated each time you reinstall the `SUNWscmasa` package.

Configuring the Logger File to Avoid `jmx` Remote Traces

To avoid `jmx` remote traces add the following lines to the beginning of your `logger.properties` file:

```
javax.management.remote.level=OFF
com.sun.jmx.remote.level=OFF
java.io.level=OFF
```

Troubleshooting Migration Problems

This section provides information about problems that you might encounter when services are migrated by using the Geographic Edition framework.

Resolving Problems With Application Resource Group Failover When Communication Is Lost With the Storage Device

When a loss of communication occurs between a node on which the application is online and the storage device, some application resource groups might not failover gracefully to the nodes from which the storage is accessible. The application resource group might result in a `ERROR_STOP_FAILED` state.

Solution or Workaround – The Oracle Solaris Cluster infrastructure does not initiate a switchover when I/O errors occur in a volume or its underlying devices. Because no switchover or failover occurs, the device service remains online on this node despite the fact that storage has been rendered inaccessible.

If this problem occurs, restart the application resource group on the correct nodes by using the standard Oracle Solaris Cluster procedures. Refer to "Clearing the STOP_FAILED Error Flag on Resources" in *Oracle Solaris Cluster 4.3 Data Services Planning and Administration Guide* for information about recovering from the `ERROR_STOP_FAILED` state and restarting the application.

The Geographic Edition framework detects state changes in the application resource group and displays the states in the output of the `geoadm status` command. For more information about using this command, see "Monitoring the Runtime Status of the Geographic Edition Framework" on page 119.

Troubleshooting Cluster Start and Restart

This section provides the following information about troubleshooting problems that you might encounter with starting and restarting the Geographic Edition framework:

Validating Protection Groups in an Error State

After a cluster reboot, the protection group configuration might be in an error state. This problem might be caused by the common agent container process not being available on one of the nodes of the cluster when the protection group is initialized after the reboot.

Solution or Workaround – To fix the configuration error, use the `geopg validate` command on the protection group that is in an error state.

Administering Stopped Protection Groups After a Cluster Restart

If an entire cluster goes down then comes back up, the expected behavior is that Geographic Edition does not restart the protection groups when the local cluster boots back up. After an entire cluster is restarted, the protection groups are deactivated on the local cluster, and the application resource groups in those protection groups will be in unmanaged state on the local cluster. After cluster restart, the administrator must determine whether data replication can be safely restarted and, if so, which cluster should have the `Primary` role.

Solution or Workaround – After a cluster has rebooted, evaluate the state of the clusters and the condition of the storage or data. Then manually reset the roles of the protection groups and restart them, or perform a failback procedure, whichever is appropriate for the situation. If the cluster is a single-node cluster, protection groups must always be manually restarted after a node reboot, because only one node exists in the cluster. For more information, see Chapter 10, “Migrating Services” and see failback and post-takeover procedures in the Geographic Edition guide for the replication component that you are using.

Restarting the Common Agent Container

The Oracle Solaris Cluster software enables the common agent container only during the Oracle Solaris Cluster software installation. Therefore, if you disable the common agent container at any time after the installation, the common agent container remains disabled.

Solution or Workaround – To enable the common agent container after a node reboot, use the `/usr/lib/cacao/bin/cacaoadm enable` command.

Matching the `Nodelist` Property of an Availability Suite Protection Group to Those of Its Device Group and Resource Group

When you add resource groups or Availability Suite device groups to a protection group, or when you run the command `geopg get` on a protection group, the order of the hosts in the `Nodelist` property of each device group and resource group in the protection group must match the order of the hosts in the `Nodelist` property of the protection group, or the operation will fail with a message similar to the following example:

```
Application resource group app-rg must have a nodelist whose physical host components
 match those
of protection group app-pg and the resources it contains.
```

The Geographic Edition framework requires that the entries in the `Nodelist` property of an Availability Suite protection group match those of any device group or resource group added to the protection group. The order of the entries in their `Nodelist` properties must also be identical.

Solution or Workaround – Ensure that the entries, and the order of the entries in the `Nodelist` properties of a protection group, of its device groups, and of its resource groups are identical.

♦ ♦ ♦ APPENDIX E

Error Return Codes for Script-Based Plug-Ins

Error Codes That Are Returned in Script-Based Plug-Ins

The script-based plug-in MBean can return any of the error codes shown in the following table.

Return Code	Error Message	Description
101	E_SBP_PROGRAM_FAILED_TO_READ_CCR	Program {0} failed to read the cluster configuration repository (CCR)
110	E_SBP_PROGRAM_EXITED_NON_ZERO	Program {0} returned a nonzero exit code.
112	E_SBP_UNEXPECTED_ERROR	Unexpected error - {0}.
125	E_SBP_ONE_OR_MORE_RGS_NON_EXISTENT	One or more of the resource groups ({0}) returned by program {1} do not exist.
126	E_SBP_RG_LIST_WRONG_FORMAT,	The output {0} returned by program {1} is invalid. The output must conform to the format `rglist=`*comma-separated-resource-groups*
127	E_SBP_NO_SUCH_FILE	An attempt was made to execute a null or nonexistent command. Check the logs for more details.
128	E_SBP_CANNOT_READ_CONFIG_FILE	Unable to read configuration file {0} from any cluster node. This file must be available on all cluster nodes.
129	E_SBP_ENTRY_NOT_FOUND_IN_CONFIG_FILE	No entry for script-based plug-in configuration {0} exists in configuration file {1}.
130	E_SBP_CONFIG_FILE_FORMAT_ERROR	Field {0} in configuration file {1} must be {2}.
131	E_SBP_CONFIG_FILE_FIELD_FORMAT_ERROR	Configuration file {0} must have three fields per script-based plug-in entry. The fields must be separated by "|".
132	E_SBP_CONFIG_FILE_INVALID_NODE	The entry for script-based plug-in {0} in configuration file {1} contains an invalid cluster node {2} in the node list field.
133	E_SBP_FAILED_TO_CHECK_X_BIT	Failed to check whether {0} is executable on cluster node {1}.
134	E_SBP_SCRIPT_DOES_NOT_EXIST	Script or program {0} does not exist on cluster node {1}.
135	E_SBP_SCRIPT_FILE_IS_NOT_EXECUTABLE	Script or program {0} is not executable on cluster node {1}.
136	E_SBP_INTERRUPTED_OR_TIMED_OUT	The command or probe was interrupted or timed out.

138	E_SBP_COULD_NOT_GET_MBEAN_PROXY	Unable to get MBean proxy for {0) on node {1}.
139	E_SBP_FILE_SECURITY_ACCESS_REFUSED	The Java Security Manager refused access to file {0) on node {1}.
140	E_SBP_NULL_FILE_NAME	File name for property {0} was null.
141	E_SBP_UNABLE_TO_CREATE_SBP_CONFIG	Unable to create script-based plug-in configuration {0}.
142	E_SBP_UNABLE_TO_MODIFY_SBP_CONFIG	Unable to modify script-based plug-in configuration {0}.
143	E_SBP_UNABLE_TO_DELETE_DG	Unable to delete script-based plug-in configuration {0}.
144	E_SBP_UNABLE_TO_CREATE_PROPERTY	Unable to create property {0}.
146	E_SBP_UNABLE_TO_UPDATE_PG_PROPERTY	Unable to update protection group property {0}.
148	E_SBP_UNABLE_TO_UPDATE_PROPERTY	Unable to update property {0}.
150	E_SBP_UNABLE_TO_GET_PROPERTY	Unable to retrieve property.
151	E_SBP_UNABLE_TO_GET_CLUSTER_NODELIST	Unable to get cluster node list.
200	E_SBP_CONFIG_ERROR	Configuration error detected for protection group {0}.
201	E_SBP_SCRIPT_FAILED	The user supplied script-based plug-in command {0} failed with error code {1} on node {2}. The script error message is {3}.
210	E_SBP_INVALID_PROPERTY_FILE	Invalid property file {0}.
221	E_SBP_MISSING_PROPERTY	Property {0} is not set.
222	E_SBP_DUPLICATE_PROPERTY	Duplicate property {0}.
223	E_SBP_INVALID_PROPERTY	Invalid property {0}.
224	E_SBP_INVALID_PROPERTY_VALUE	Invalid value for property {0}.
225	E_SBP_SBP_CONFIG_ALREADY_IN_PG	Script-based plug-in configuration {0} already in protection group {1}.
226	E_SBP_SBP_CONFIG_NOT_FOUND_IN_PG	Script-based plug-in configuration {0} is not found in protection group {1}.
231	E_SBP_UNABLE_TO_NOTIFY_STATUS_CHANGE	Unable to send change notification for data replication status.
233	E_SBP_RG_OFFLINE_EXCEPTION	Failed to take resource group {0} offline.
234	E_SBP_SAME_PROPERTY_VALUE	Property value already set. No modification is needed.
235	E_SBP_UNEXPECTED_EXCEPTION	Unexpected exception - {0}.
236	E_SBP_SERVER_REQUEST_FAILED_DUE_TO_TIMEOUT	Error in running control script on host {0}. Operation timed out after {1} seconds.
237	E_SBP_SERVER_REQUEST_FAILED_WITH_REASON	Error in running control script on host {0} due to system error - {1}\

Index

D

E

F

G

H

R

S

T