Oracle® Solaris Cluster Geographic Edition Reference Manual

ORACLE®

Contents

Preface

The *Oracle Solaris Cluster Geographic Edition Reference Manual* provides reference information about commands that are supplied with Oracle Solaris Cluster Geographic Edition software. This book is intended for experienced system administrators with extensive knowledge of Oracle software and hardware. This book is not to be used as a planning or presales guide. The information in this book assumes knowledge of the Oracle Solaris Operating System and expertise with the data replication software that is used with Oracle Solaris Cluster Geographic Edition software.

Both novice users and those familiar with the Oracle Solaris Operating System can use online man pages to obtain information about their SPARC based system or x86 based system and its features.

A man page is intended to answer concisely the question "What does this command do?" The man pages in general comprise a reference manual. They are not intended to be a tutorial.

Note - Oracle Solaris Cluster software runs on two platforms, SPARC and x86. The information in this book pertains to both platforms unless otherwise specified in a special chapter, section, note, bulleted item, figure, table, or example.

Overview

The following contains a brief description of each man page section and the information it references.

- Section 1HA describes, in alphabetical order, Geographic Edition API commands.
- Section 1M describes, in alphabetical order, commands that are used chiefly for system maintenance and administration purposes.

The following is a generic format for man pages. The man pages of each manual section generally follow this order, but include only needed headings. For example, if no bugs can be reported, no BUGS section is included. See the intro pages for more information and detail about each section, and man(1) for general information about man pages.

NAME   This section gives the names of the commands or functions that are
                    documented, followed by a brief description of what they do.

SYNOPSIS This section shows the syntax of commands or functions. If a command or file does not exist in the standard path, its full path name is shown. Options and arguments are alphabetized, with single-letter arguments first, and options with arguments next, unless a different argument order is required.

The following special characters are used in this section:

[ ] Brackets. The option or argument that is enclosed in these brackets is optional. If the brackets are omitted, the argument must be specified.

… Ellipses. Several values can be provided for the previous argument, or the previous argument can be specified multiple times, for example, "`filename ...`".

| Separator. Only one of the arguments separated by this character can be specified at a time.

{ } Braces. The options and/or arguments enclosed within braces are interdependent. All characters within braces must be treated as a unit.

PROTOCOL This section occurs only in subsection 3R and indicates the protocol description file.

DESCRIPTION This section defines the functionality and behavior of the service. Thus it describes concisely what the command does. DESCRIPTION does not discuss OPTIONS or cite EXAMPLES. Interactive commands, subcommands, requests, macros, and functions are described under USAGE.

IOCTL This section appears on pages in Section 7 only. Only the device class that supplies appropriate parameters to the `ioctl`(2) system call is called `ioctl` and generates its own heading. `ioctl` calls for a specific device are listed alphabetically (on the man page for that specific device). `ioctl` calls are used for a particular class of devices. All these calls have an `io` ending, such as `mtio`(7I).

OPTIONS This section lists the command options with a concise summary of what each option does. The options are listed literally and in the order they appear in the SYNOPSIS section. Possible arguments to options are discussed under the option, and where appropriate, default values are supplied.

OPERANDS   This section lists the command operands and describes how they affect the actions of the command.

OUTPUT   This section describes the output – standard output, standard error, or output files – generated by the command.

RETURN   If the man page documents functions that return values, this section lists
VALUES   these values and describes the conditions under which they are returned. If a function can return only constant values, such as 0 or –1, these values are listed in tagged paragraphs. Otherwise, a single paragraph describes the return values of each function. Functions that are declared void do not return values, so they are not discussed in RETURN VALUES.

ERRORS   On failure, most functions place an error code in the global variable `errno` that indicates why they failed. This section lists alphabetically all error codes a function can generate and describes the conditions that cause each error. When more than one condition can cause the same error, each condition is described in a separate paragraph under the error code.

USAGE   This section lists special rules, features, and commands that require in-depth explanations. The subsections that are listed here are used to explain built-in functionality:

                    Commands
                    Modifiers
                    Variables
                    Expressions
                    Input Grammar

EXAMPLES   This section provides examples of usage or of how to use a command or function. Wherever possible, a complete example, which includes command-line entry and machine response, is shown. Whenever an example is given, the prompt is shown as `example%`, or if the user must be a `root` role, `example#`. Examples are followed by explanations, variable substitution rules, or returned values. Most examples illustrate concepts from the SYNOPSIS, DESCRIPTION, OPTIONS, and USAGE sections.

ENVIRONMENT   This section lists any environment variables that the command or
VARIABLES   function affects, followed by a brief description of the effect.

EXIT STATUS   This section lists the values the command returns to the calling program or shell and the conditions that cause these values to be returned. Usually, zero is returned for successful completion, and values other than zero are returned for various error conditions.

FILES

This section lists all file names that are referred to by the man page, files of interest, and files created or required by commands. Each file name is followed by a descriptive summary or explanation.

ATTRIBUTES

This section lists characteristics of commands, utilities, and device drivers by defining the attribute type and its corresponding value. See `attributes`(5) for more information.

SEE ALSO

This section lists references to other man pages, in-house documentation, and outside publications.

DIAGNOSTICS

This section lists diagnostic messages with a brief explanation of the condition that caused the error.

WARNINGS

This section lists warnings about special conditions that could seriously affect your working conditions. WARNINGS is not a list of diagnostics.

NOTES

This section lists additional information that does not belong anywhere else on the page. NOTES covers points of special interest to the user. Critical information is never covered here.

BUGS

This section describes known bugs and, wherever possible, suggests workarounds.

OSC4GEO 1HA

Oracle Solaris Cluster Geographic Edition Reference Manual • July 2014, E39677-01

Name

scha_devicegroup_get — get information about a Geographic Edition device group or other data replication component

scha_devicegroup_get -G *protection-group* -D *device-group* -p *property* [-d] [-r]

The `scgeo_devicegroup_get` command accesses information about an Oracle Solaris Cluster Geographic Edition device group or other data replication component.

Use this command in shell script implementations of the callback methods for partnerships.

Information is output by this command to the standard output (`stdout`) in formatted strings, as described in the `scha_cmds`(1HA) man page. Output is a string or several strings that are output on separate lines. You can store the output in shell variables. You can also parse the output by using the `awk`(1) command or other shell commands for further use by the script.

You must be root to use this command.

The following options are supported:

-D *device-group*

Indicates the data replication component information to be accessed.

-d

Specifies debug mode.

-G *protection-group*

Specifies the name of the protection group.

-p *property*

Specifies the name of a data replication component property. See the `geopg`(1M) on page 55 man page for information about properties for each data replication product.

-r

Selects the remote value for non-global properties.

EXAMPLE 1   Using the `scgeo_devicegroup_get` Command to Query a Property of a Data Replication Component

Use the following command to retrieve the value of the `switchover_args` property of the `mysql-dg` data replication component in the `paris-london` protection group.

```
# /usr/cluster/lib/geo/lib/scgeo_devicegroup_get -G paris-london -D mysql-dg \
-p switchover_args
```
list of arguments

The following exit values are returned:

`0`   The command completed successfully.

`1`   A general usage error occurred.

`100`   An error occurred.

                    Failure error codes are described scha_calls(3HA).

See attributes(5) for descriptions of the following attributes:

ATTRIBUTE TYPE	ATTRIBUTE VALUE
Availability	`ha-cluster/geo/geo-framework`
Interface Stability	Stable

awk(1), scha_cmds(1HA), geopg(1M) on page 55, scha_calls(3HA), attributes(5)

Name

scgeo_partnership_get — get information about a Geographic Edition partnership

scgeo_partnership_get -P *partnership* -p *property* [-d]

The scgeo_partnership_get command accesses information about an Oracle Solaris Cluster Geographic Edition partnership.

Use this command in shell script implementations of the callback methods for partnerships.

The property names do not constitute a fixed interface. Some knowledge of the data stored in the Cluster Configuration Repository (CCR) is required, in particular the /etc/cluster/ccr/global/pts_*partnership-name* file.

Information is output by this command to the standard output (stdout) in formatted strings, as described in the scha_cmds(1HA) man page. Output is a string or several strings that are output on separate lines. You can store the output in shell variables. You can also parse the output by using the awk(1) command or other shell commands for further use by the script.

You must be root to use this command.

The following options are supported:

-d

Specifies debug mode.

-P *partnership*

Indicates the partnership information to be accessed.

-p *property*

Specifies the name of a partnership property. See the geops(1M) on page 79 man page for information about partnership properties.

However, property names do not constitute a fixed interface. Some knowledge of the data stored in the Cluster Configuration Repository (CCR) is required, in particular the /etc/cluster/ccr/global/pts_*partnership-name* file.

EXAMPLE 2 Using the scgeo_partnership_get Command to Query the clusterNamePartner1 Value of a Partnership

Use the following command to retrieve the name of the primary cluster in the paris-london partnership, which is paris.

```
# /usr/cluster/lib/geo/lib/scgeo_partnership_get -P paris-london -p clusterNamePartner1
paris
```

EXAMPLE 3   Using the `scgeo_partnership_get` Command to Query a Property of a Partnership

Use the following command to retrieve the value of the `requestTimeout` property of the `paris-london` partnership.

```
# /usr/cluster/lib/geo/lib/scgeo_partnership_get -P paris-london -p requestTimeout
30000
```

The following exit values are returned:

`0`	The command completed successfully.
`1`	A general usage error occurred.
`100`	An error occurred.
	Failure error codes are described `scha_calls`(3HA).

`/etc/cluster/ccr/global/pts_`*partnership-name*

     Contains CCR configuration information about the *partnership-name* partnership.

See `attributes`(5) for descriptions of the following attributes:

ATTRIBUTE TYPE	ATTRIBUTE VALUE
Availability	`ha-cluster/geo/geo-framework`
Interface Stability	Stable

`awk`(1), `scha_cmds`(1HA), `geops`(1M) on page 79, `scha_calls`(3HA), `attributes`(5)

Name

scgeo_protectiongroup_get — get information about a Geographic Edition protection group

```
scgeo_protectiongroup_get -G protection-group -p property [-d] [-r]
```

The `scgeo_protectiongroup_get` command accesses information about an Oracle Solaris Cluster Geographic Edition protection group.

Information is output by this command to the standard output (`stdout`) in formatted strings, as described in the `scha_cmds`(1HA) man page. Output is a string or several strings that are output on separate lines. You can store the output in shell variables. You can also parse the output by using the `awk`(1) command or other shell commands for further use by the script.

You must be root to use this command.

The following options are supported:

`-d`

> Specifies debug mode.

`-G` *protection-group*

> Indicates the protection-group information to be accessed.

`-p` *property*

> Specifies the name of a protection-group property. The following *property* values retrieve the corresponding value of the named property.
>
> ```
> APPLICATION_RG_LIST
> APPLICATION_RG_NODE_LIST
> CLUSTER_NAME
> CLUSTER_ROLE
> DESCRIPTION
> DEVICEGROUP_ENTRY_LIST
> EXTERNAL_DEPENDENCY_ALLOWED
> LOCAL_APPLICATION_RG_NODE_LIST
> LOCAL_CLUSTER_NAME
> LOCAL_CLUSTER_ROLE
> PARTNERSHIP_NAME
> PG_CREATION_NODENAME
> PG_VALIDATED
> PG_VALIDATED_LOCALLY
> PG_VALIDATED_REMOTELY
> REMOTE_APPLICATION_RG_NODE_LIST
> REMOTE_CLUSTER_NAME
> REMOTE_CLUSTER_ROLE
> ```

```
ROLE_CHANGE_ACTION_ARGS
ROLE_CHANGE_ACTION_COMMAND
TIMEOUT
```

Additionally, you can specify property names that are specific to a data replication product.

-r

Selects the remote value for non-global properties.

EXAMPLE 4   Using the `scgeo_protectiongroup_get` Command to Query a Property of a Protection Group

Use the following command to retrieve the name of the MySQL configuration script `create_config_script` that is used in the `sbp-pg` protection group.

 # /usr/cluster/lib/geo/lib/scgeo_protectiongroup_get -G sbp-pg -p create_config_script

`/var/tmp/create.sh`

The following exit values are returned:

`0`   The command completed successfully.

`1`   A general usage error occurred.

`100`   An error occurred.

                      Failure error codes are described scha_calls(3HA).

See attributes(5) for descriptions of the following attributes:

ATTRIBUTE TYPE	ATTRIBUTE VALUE
Availability	`ha-cluster/geo/geo-framework`
Interface Stability	Stable

awk(1), scha_cmds(1HA), su(1M), scha_calls(3HA), attributes(5)

OSC4GEO 1M

Name

geoadm — enable or disable the Oracle Solaris Cluster Geographic Edition framework on the local cluster

```
/usr/cluster/bin/geoadm [subcommand] -?

/usr/cluster/bin/geoadm -V

/usr/cluster/bin/geoadm show

/usr/cluster/bin/geoadm start

/usr/cluster/bin/geoadm status

/usr/cluster/bin/geoadm stop [-f | -i]
```

The geoadm command enables or disables the Oracle Solaris Cluster Geographic Edition (Geographic Edition) framework on the local cluster. You can also use this command to check whether the Geographic Edition framework is enabled on the cluster.

After installation, the Geographic Edition framework must be enabled by running the command geoadm start. This command enables the cluster to participate in partnerships and to host protection groups. The geoadm start command configures the highly available infrastructure that unlocks the Geographic Edition framework on the cluster.

The geoadm stop command stops the highly available infrastructure that leaves the Geographic Edition framework locked on the cluster. Run the geoadm stop command before you remove product packages.

Note - The geoadm stop command can successfully disable the Geographic Edition framework on a cluster only under one condition. All local states of the protection groups in which the cluster is involved must be Offline or Unknown.

Running the geoadm start or the geoadm stop command on one node of the cluster affects the entire cluster.

To use the geoadm command to enable or disable the Geographic Edition infrastructure, you must be assigned the proper role-based access control (RBAC) rights profile.

If you have root access, you have permissions to perform any operation. If you do not have root access, the following RBAC rights apply:

- **Basic Solaris User.** You can read information about Geographic Edition entities by running commands such as geoadm show and geoadm status.
- **Geo Management.** You can perform all the read operations that someone with Basic Solaris User authorizations can perform. You can also perform administrative and configuration tasks such as geoadm start and geoadm stop.

For more information, see the rbac(5) man page and "Planning Security" in "Oracle Solaris Cluster Geographic Edition Installation Guide ".

The general form of this command is as follows:

geoadm [*subcommand*] [*options*] [*operands*]

You can omit *subcommand* only if *options* specifies the option -?, -o, -V, or -v.

Each option of this command has a long form and a short form. Both forms of each option are given with the description of the option in the OPTIONS section of this man page.

The following subcommands are supported:

show

   Display whether the Geographic Edition framework is enabled on the cluster.

start

   Configure and enable the Geographic Edition infrastructure on the cluster.

status

   Display the runtime status of the Geographic Edition entities on the local cluster.

   The Geographic Edition framework must be installed on the local cluster before you can run the status subcommand. You can run the status subcommand whether or not the cluster has been enabled for partnership.

   The status subcommand displays the following information:

   - Whether the local cluster is enabled for partnership.
   - Whether the local cluster is involved in a partnership. If the cluster is involved in a partnership, the status subcommand lists all partnership members.
   - Heartbeat status.
   - Protection group status.
   - Site and multigroup status.
   - Status of ongoing transactions.

   See the EXTENDED DESCRIPTION section for details about the possible values for each status.

stop

   Disable the Geographic Edition framework and configuration on the cluster.

   Caution - All protection groups on the cluster must be in the Offline state to run the stop subcommand successfully.

The `stop` subcommand removes the cluster state and Geographic Edition framework resource groups on the cluster, but the `stop` subcommand does not remove data replication resource groups.

The Geographic Edition framework is enabled and configured again the first time you run the `start` subcommand after running the `stop` subcommand.

The following options are supported:

`-?`
`--help`

Displays help information. You can specify this option without a subcommand or with a subcommand.

- If you specify this option without a subcommand, the list of subcommands for this command is displayed.
- If you specify this option with a subcommand, the usage options for the subcommand are displayed.

The question mark might be interpreted as a special character by some shells. Use quotes (`-"?"`) or an escape character to avoid pattern matching.

`-f`
`--force`

Indicates that you want to bypass the command confirmation questions while the Geographic Edition framework is being disabled.

`-i`
`--interactive`

Indicates that you want to display the interactive command confirmation questions while the Geographic Edition framework is being disabled.

`-V`
`--version`

Displays the version of the command.

Do not specify this option with subcommands, operands, or other options. The subcommands, operands, or other options are ignored. The `-V` option only displays the version of the command. No other operations are performed.

The following sections list the status descriptions.

Partnership Runtime Status

This section describes the types of status that are reported for partnerships, including heartbeat status.

- **Partnership**– Displays partnership status for the local cluster only. The partnership status can be one of the following:

  OK   The partner clusters are connected.

  Degraded   The partnership is successfully created but connection with the partner cluster is not yet established. This status value usually occurs when the partner cluster is not yet configured in the partnership.

  Error   The partner clusters are disconnected.

- **(Partnership) Synchronization** – Displays partnership synchronization status for all the clusters in a partnership. The partnership synchronization status can be one of the following:

  OK   The configuration is synchronized between partner clusters.

  Error   The configuration on the partner clusters is different. You must synchronize the partnership again.

  Mismatch   The partner clusters have been configured individually. Therefore, you must remove the configuration on one cluster and copy the configuration of the partner cluster.

  Unknown   Information is not accessible because the partners are disconnected.

- **(Partnership) ICRM Connection** – Displays Intercluster Resource Management (ICRM) Connection status for the cluster management agent that is running on the remote cluster. The ICRM connection status can be one of the following:

  OK   The ICRM module on each of the partner clusters can communicate with each other.

  Error   The ICRM module on the local cluster is unable to communicate with the ICRM module on the remote cluster.

- **(Partnership) Heartbeat** – Displays heartbeat status for the local cluster only. The heartbeat status can be one of the following:

  OK   Heartbeat monitoring is enabled, and the partner cluster is responding within timeout and retry periods.

  Degraded   Heartbeat checks are running ut 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.

- **(Partnership) Plug-in** – Displays heartbeat plug-in status for the local cluster only. The plug-in status can be one of the following:

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.

Protection Group Runtime Status

This section describes the types of status that are reported for a protection group, including data replication and resource group status.

- **Protection group** – Displays the overall protection group status for all the clusters in the protection group. The overall protection group status can be one of the following:

OK	No component of the protection group on either partner cluster is in the Degraded, Error, or Unknown status, and the protection group configuration is the same on both partner clusters. The protection group can be either online or offline.
Degraded	Data replication is either not running or is in a partial error state.
Error	At least one component of the partnership, such as configuration, data replication, or resource groups, is in an error state on at least one partner cluster, or the protection group configuration is different between the partner clusters.
Unknown	The status for at least one component of the protection group, such as configuration, data replication, or resource groups, is unknown, or the status of the protection group on the cluster is not accessible.

- **(Protection group) Synchronization** – Displays the protection group synchronization status for all the clusters in the protection group. The synchronization status can be one of the following:

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 the partner clusters individually. Therefore, you must remove the configuration on one cluster and copy the configuration of the partner cluster.

Unknown	Information is not accessible because the partners are disconnected.

- **(Protection group) Cluster** – Displays the local cluster status of the protection group for a single cluster only. The cluster status can be one of the following:

OK	No component of the protection group is in Degraded, Error, or Unknown status. The protection group can be either online or offline.
Degraded	Data replication is either not running or is in a partial error state.
Error	At least one component of the protection group, such as configuration, data replication, or resource groups, is in an error state.
Unknown	Status of the protection group on the cluster is not accessible, or the status for at least one component of the protection group, such as configuration, data replication, or resource groups, is unknown.
None	The data replication or resource groups component is not configured in the protection group.

- **(Protection group cluster) Role** – Displays what role the cluster has for the protection group. The activation status can be one of the following:

Primary	The cluster is the Primary for this protection group.
Secondary	The cluster is the Secondary for this protection group.
Unknown	The cluster role state for the protection group is inaccessible.

- **(Protection group cluster) Activation** – Displays the activation status of a protection group for a single cluster only. The activation status can be one of the following:

Activated	The protection group is online on the cluster.
Deactivated	The protection group is offline on the cluster.
Unknown	The activation state of the protection group on the cluster is inaccessible.

- **(Protection group cluster) Configuration** – Displays the configuration status of a protection group for a single cluster only. The configuration status of a protection group can be one of the following:

OK	The protection group configuration has been validated without errors on the cluster.
Error	The validation of the protection group configuration resulted in an error. You must validate the protection group again.

Unknown	Information is not accessible because the partner clusters are disconnected.

- **(Protection group cluster) Data replication** – Displays the status of data replication that has been configured for the protection group on a single cluster. The data replication status of a protection group can be one of the following:

OK	Data replication is configured, online, and running without errors.
Degraded	Data replication is either off or in a partial error state.
Error	Data replication is not operating because of an error.
None	Data replication has not been configured.
Unknown	Information is not accessible because the partners are disconnected.
N/A	(Oracle ZFS Storage Appliance data replication only) The data replication status of the protection group could not be mapped. Data replication itself is in a valid state but is in an Error state for the protection group.

- **(Protection group cluster) Resource groups** – Displays the status of resource groups that have been configured for the protection group on a single cluster. The resource group status of a protection group can be one of the following:

OK	All resource groups are online on the primary cluster, and all resource groups are offline or unmanaged on the secondary cluster.
Error	Not all resource groups are online on the primary cluster, or not all resource groups are offline or unmanaged on the secondary cluster.
None	No resource group is protected by this protection group.
Unknown	Information is not accessible because the partners are disconnected.

Site Runtime Status

This section describes the types of status that are reported for a site, including multigroup status.

- **(Site cluster) Configuration** –Displays configuration status for each cluster in all sites that are associated with the issuing cluster. The configuration status can be one of the following:

OK	The site configuration is correct.
Error	The site configuration has an error.

Unknown   The site configuration is not checked.

- **(Site cluster) Synchronization** – Displays synchronization status for each site cluster. The synchronization status can be one of the following:

    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 cluster) Heartbeat** – Displays heartbeat status for a single site cluster only. The heartbeat status can be one of the following:

    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 cluster) Plug-In** – Displays heartbeat plug-in status for a single site cluster only. The heartbeat plug-in status can be one of the following:

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.

- **(Multigroup) Configuration** – Displays configuration status for all multigroups that are associated with the issuing cluster. The multigroup status can be one of the following:

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** – Displays synchronization with cluster status for all multigroups that are associated with the issuing cluster. The synchronization states of a multigroup can be one of the following:

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.

The following exit values are returned:

0	The command completed successfully.
nonzero	An error has occurred.

EXAMPLE 5   Enabling the Geographic Edition Framework

The following `geoadm` command enables the Geographic Edition control module on the local cluster.

```
# geoadm start
```

EXAMPLE 6   Disabling the Geographic Edition Framework

The following `geoadm` command removes the Geographic Edition framework resource groups that were created when you enabled the Geographic Edition framework.

```
# geoadm stop
```

EXAMPLE 7   Checking the Status of the Geographic Edition Framework

The following `geoadm` command displays whether the Geographic Edition framework is enabled on the local cluster and, if enable, the nodes on which the framework is active.

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

Command execution successful
#
```

EXAMPLE 8   Monitoring the Runtime Status of the Geographic Edition Framework

The following `geoadm` command displays the runtime status of the local Geographic Edition enabled cluster.

```
# 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  : None
          Resource groups   : None

     Cluster cluster-newyork : OK
          Role              : Secondary
          Activation state  : Deactivated
          Configuration     : OK
          Data replication  : None
          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  : None
          Resource groups   : None

     Cluster cluster-newyork : OK
          Role              : Secondary
          Activation state  : Deactivated
          Configuration     : OK
          Data replication  : None
          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
```

See attributes(5) for descriptions of the following attributes.

ATTRIBUTE TYPE	ATTRIBUTE VALUE
Availability	ha-cluster/geo/geo-framework
Interface Stability	Evolving

rbac(5), geohb(1M) on page 33, geomg(1M) on page 39,
geopg(1M) on page 55, geops(1M) on page 79, geosite(1M) on page 87

Name

geohb — configure and manage the heartbeat mechanism

```
/usr/cluster/bin/geohb [subcommand] -?

/usr/cluster/bin/geohb -V

/usr/cluster/bin/geohb add-plugin plugin heartbeat -p property [-p…]

/usr/cluster/bin/geohb create -r cluster-list [-p property [-p…]] heartbeat

/usr/cluster/bin/geohb delete heartbeat

/usr/cluster/bin/geohb list

/usr/cluster/bin/geohb modify-plugin -p property [-p…] plugin heartbeat

/usr/cluster/bin/geohb remove-plugin plugin heartbeat

/usr/cluster/bin/geohb set-prop -p property [-p…] heartbeat …

/usr/cluster/bin/geohb show [heartbeat[,…]]

/usr/cluster/bin/geohb status [heartbeat[,…]]
```

The geohb command enables you to configure and manage the heartbeat mechanism.

A heartbeat is a monitor between two clusters: a requester cluster and a responder cluster. Creating a partnership establishes two heartbeats, one in each direction. For example, a partnership between a primary cluster, cluster-paris, and a secondary cluster, cluster-newyork, contains two heartbeats. One heartbeat has cluster-paris as the requester and cluster-newyork as the responder. The other heartbeat has cluster-newyork as the requester and cluster-newyork as the responder.

The Oracle Solaris Cluster Geographic Edition software provides a default heartbeat mechanism that is based on the TCP/UDP plug-in as the primary plug-in and the ping plug-in as backup.

Run the geohb command on a cluster that has been enabled for partnership.

To run the geohb command to configure and manage the heartbeat mechanism, you must be assigned the proper role-based access control (RBAC) rights profile.

If you have root access, you have permissions to perform any operation. If you do not have root access, the following RBAC rights apply:

- **Basic Solaris User.** You can read information about Oracle Solaris Cluster Geographic Edition entities by running commands such as geohb list.

- **Geo Management.** You can perform all the read operations that someone with Basic Solaris User access can perform. You can also perform administrative and configuration tasks such as `geohb create`, `geohb delete`, and `geoadm set-prop`.

For more information, see the `rbac`(5) man page and "Planning Security" in "Oracle Solaris Cluster Geographic Edition Installation Guide ".

The general form of this command is as follows:

`geohb` [*subcommand*] [*options*] [*operands*]

You can omit *subcommand* only if *options* specifies the option -?, -o, -V, or -v.

Each option of this command has a long form and a short form. Both forms of each option are given with the description of the option in the OPTIONS section of this man page.

The following subcommands are supported:

`add-plugin`

Add a plug-in to an existing heartbeat. If you specify a custom plug-in, you must also specify the path to your custom plug-in command by using the `Query_cmd` property.

`create`

Create a heartbeat. You can monitor heartbeat status by running the `geoadm`(1M) on page 21 command. You must configure the remote cluster to make the heartbeat operational.

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

`delete`

Delete a heartbeat.

`list`

List the heartbeats that are configured on the cluster.

`modify-plugin`

Modify heartbeat plug-in properties.

`remove-plugin`

Remove a plug-in from a heartbeat.

`set-prop`

Modify heartbeat properties.

`show`

> Display existing configuration information for one or more heartbeats. Use a comma-separated list to specify multiple heartbeats.

`status`

> Display status information for one or more heartbeats. Use a comma-separated list to specify multiple heartbeats.

The following options are supported:

`-?`
`--help`

> Displays help information.
>
> You can specify this option without a subcommand or with a subcommand.
>
> - If you specify this option without a subcommand, the list of subcommands for this command is displayed.
> - If you specify this option with a subcommand, the usage options for the subcommand are displayed.
>
> The question mark might be interpreted as a special character by some shells. Use quotes (`-"?"`) or an escape character to avoid pattern matching.

`-p` *property*
`--property` *property*

> Specifies the properties of a heartbeat or heartbeat plug-in.
>
> A heartbeat property is assigned a value by using a *name*=*statement* pair. Multiple properties might be set at one time by using multiple statements.
>
> The values for these properties are assigned at creation and tunable at runtime.
>
> See the EXTENDED DESCRIPTION section for currently defined properties.

`-r` *cluster-list*
`--remote-cluster` *cluster-list*

> Specifies the name of a remote cluster with which the local cluster should establish heartbeat monitoring.

`-V`
`--version`

> Displays the version of the command.
>
> Do not specify this option with subcommands, operands, or other options. The subcommands, operands, or other options are ignored. The `-V` option only displays the version of the command. No other operations are performed.

The following operands are supported:

heartbeat

> Specifies an identifier for the heartbeat on the local cluster. If you are trying to create a new heartbeat, and the specified identifier already exists, the `geohb create` command fails.

plugin

> Specifies the name of the heartbeat plug-in.

The following sections list the heartbeat and heartbeat plug-in properties.

Heartbeat Properties

You can specify the following heartbeat property:

`Query_interval`

> Specifies the frequency between heartbeat status requests in seconds. 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.
>
> Optional property.
>
> Type: Integer.
>
> Tuning recommendations: The value of this property is assigned at creation and tunable at runtime.
>
> Minimum value: 20 seconds
>
> Maximum value: 300 seconds
>
> Default value: 120 seconds

Heartbeat Plug-in Properties

Heartbeat plug-in properties determine how a heartbeat functions.

`Plugin_properties`

> Specifies a property string that is specific to the plug-in.
>
> Optional property.
>
> Type: String.
>
> Tuning recommendations: The value of this property is assigned at creation and tunable at runtime.
>
> Default value: 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* string indicates that the plug-in uses IPsec.

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`

Specifies the path to the command for a heartbeat status request.

Required property if the plug-in does not specify a predefined plug-in.

Type: String.

Tuning recommendations: The value of this property is assigned at creation and tunable at runtime.

Default value: None.

`Requester_agent`

Specifies the absolute path to requester agent.

Optional property.

Type: String.

Tuning recommendations: The value of this property for the default plug-in should not be tuned except for testing purposes.

Default value: None.

`Responder_agent`

Specifies the absolute path to the responder agent.

Optional property.

Type: String.

Tuning recommendations: The value of this property for the default plug-in should not be tuned except for testing purposes.

Default value: None.

`Type`

Specifies the type of plug-in. Set to either `Primary` or `Backup`.

Required property.

Type: Enum.

Tuning recommendations: The value of this property is assigned at creation and tunable at runtime.

Default value: None, except for heartbeats with default heartbeat name `ping_plugin`. In this case, the default value is `Backup`.

The following exit values are returned:

`0`   The command completed successfully, indicating that the remote cluster is alive.

nonzero   An error has occurred, meaning that the remote cluster did
                                 not respond to the heartbeat check.

EXAMPLE 9   Creating a Heartbeat

The following geohb command creates a heartbeat that is named `paris-to-newyork`, which
communicates between the local cluster and the cluster `cluster-newyork`.

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

EXAMPLE 10   Creating a Heartbeat Plug-in

The following geohb command creates a heartbeat plug-in that is named `command1` for the
heartbeat `paris-to-newyork`.

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

EXAMPLE 11   Modifying a Heartbeat

The following geohb command modifies the properties for the default heartbeat between
`cluster-paris` and `cluster-newyork`.

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

EXAMPLE 12   Deleting a Plug-in From a Heartbeat

The following geohb command deletes the plug-in that is named `command1`, from the heartbeat
that is named `paris-to-newyork`.

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

See attributes(5) for descriptions of the following attributes.

ATTRIBUTE TYPE	ATTRIBUTE VALUE
Availability	ha-cluster/geo/geo-framework
Interface Stability	Evolving

rbac(5), geops(1M) on page 79

Name

geomg — create or manage multigroups

```
/usr/cluster/bin/geomg [subcommand] -?

/usr/cluster/bin/geomg -V

/usr/cluster/bin/geomg add-protection-group protection-group-list multigroup

/usr/cluster/bin/geomg create -s site [[-p name=value] [-p …]] multigroup

/usr/cluster/bin/geomg delete [-f] multigroup

/usr/cluster/bin/geomg list [-s site […]]

/usr/cluster/bin/geomg remove-protection-group protection-group-list multigroup

/usr/cluster/bin/geomg set-prop -p name=value [-p…] multigroup

/usr/cluster/bin/geomg show [-s site[,…]] [multigroup […]]

/usr/cluster/bin/geomg start -e {local | global} [-f] [-n] multigroup

/usr/cluster/bin/geomg status multigroup[,…]

/usr/cluster/bin/geomg stop -e {local | global} [-f] [-D] multigroup

/usr/cluster/bin/geomg switchover [-u] [-i] [-f] -s site multigroup

/usr/cluster/bin/geomg takeover [-f] -s site multigroup

/usr/cluster/bin/geomg update cluster multigroup

/usr/cluster/bin/geomg validate multigroup
```

The geomg command enables you to configure and maintain multigroups. To administer an individual protection group, instead use the geopg command. See the geopg(1M) on page 55 man page for more information.

Before you create a multigroup, ensure that the clusters whose protection groups will be members of the multigroup are already configured within the same site. See the geosite(1M) on page 87 man page for more information about sites.

Several subcommands must be issued from a particular cluster in the site:

- To run the geomg commands to create, modify, or delete a multigroup, you must run the command from a cluster that is a controller for the specified site.
- To run the geomg commands to stop or start the protection groups in a multigroup, you must run the command from a cluster that is a controller for a site that is defined for the multigroup.

- To run the `geomg` commands to switch over or take over the protection groups in a multigroup, you must run the command from a cluster that is a controller for the target site, which is defined for the multigroup.

Normally, multigroup configuration changes that are made on the issuing cluster are automatically synchronized with the other clusters in the same site as the issuing cluster. If a cluster is temporarily unavailable due to an `Error` or `Unknown` status, the cluster is autosynchronized when it returns to the `OK` status. If you need to manually synchronize changes, use the `update` subcommand.

If you have `root` access, you have permission to perform any operation. If you do not have root access, the following RBAC rights apply:

- **Basic Solaris User.** You can read information about Oracle Solaris Cluster Geographic Edition (Geographic Edition) entities by running commands such as `geomg list`.
- **Geo Management.** You can perform all the read operations that someone with Basic Solaris User access can perform. You can use commands like `geomg create`, `geomg delete`, and `geomg start` to perform administrative and configuration operations.

For more information, see the `rbac`(5) man page and "Planning Security" in "Oracle Solaris Cluster Geographic Edition Installation Guide ".

The general form of this command is as follows:

`geomg [`*subcommand*`] [`*options*`] [`*operands*`]`

You can omit *subcommand* only if *options* specifies the option `-?` or `-V`.

Each option of this command has a long form and a short form. Both forms of each option are given with the description of the option in the OPTIONS section of this man page.

The following subcommands are supported:

`add-protection-group`

Add one or more protection groups to the specified multigroup. This subcommand must be issued from a site controller.

The system performs this action on the local cluster, then propagates the action to other clusters in the site.

Specify protection groups by using the form *cluster*:*protection-group*. Use a comma-separated list to specify multiple protection groups.

To specify a dependency between protection groups, called a dependency chain, use a slash (/) to separate the name of one or more dependent protection groups from the name of the protection group to depend on.

For example, if the `pg1` protection group depends on the `pg2` protection group, and both protection groups are configured on the `berlin` cluster, you specify them as follows:

```
berlin:pg1/berlin:pg2
```

When a switchover is performed, `pg1` is stopped on its current primary partner cluster before `pg2` is stopped, and `pg2` is started on its new primary partner cluster before `pg1` is started.

To specify the dependency of multiple protection groups to another protection group, enclose the dependent protection group names with parentheses and separate the names with commas (,). The parentheses are followed by a slash and the name of the depended-on protection group.

For example, if the `pg1` and `pg2` protection groups on the `berlin` cluster depend on the `pg3` protection group on the `tokyo` cluster, you specify them as follows:

```
(berlin:pg1,berlin:pg2)/tokyo:pg3
```

You cannot use parentheses to enclose multiple protection group names after the slash in a dependency chain.

You can use more than one slash to set multiple dependency levels. For example, if the `pg1` protection group on the `berlin` cluster depends on the `pg2` protection group on the same cluster, and that protection group depends on the `pg3` protection group on the `tokyo` cluster, you specify the three protection groups and their relationships as follows:

```
berlin:pg1/berlin:pg2/tokyo:pg3
```

When you add a protection group that is in a dependency chain, such as `berlin:pg1/berlin:pg2`, you cannot later modify or remove only one of the protection groups in that interdependency. You must instead use the `remove-protection-group` subcommand to remove the entire protection group interdependency from the multigroup. Then you use the `add-protection-group` subcommand to add back any protection groups that you had to remove but want to keep in the multigroup.

See the OPERANDS section for more information about using the slash (/) and parentheses operands to specify dependencies between protection groups in a multigroup.

`create`

Create a multigroup at the specified site. This subcommand must be issued from a site controller.

The system performs this action on the local cluster, then propagates the action to other clusters in the site.

You specify the name to assign to the new multigroup. Multigroup names must be unique throughout the site. If multiple sites share a common cluster, those sites cannot contain multigroups of the same name. If you specify the name of an existing multigroup, the operation fails with an error.

The multigroup software automatically calculates and sets an appropriate value for the multigroup `Timeout` property. This value is based on the `Timeout` property settings for the protection groups in the multigroup. The multigroup `Timeout` property value might be adjusted automatically by the multigroup software when protection groups are added

or removed, or when the multigroup is validated. The `Timeout` property can also be manually set to any value that is higher than the value that is automatically calculated by the multigroup software.

delete

Delete the specified multigroup from all clusters in the site where the multigroup is configured. This subcommand must be issued from a site controller.

The system performs this action on the local cluster, then propagates the action to other clusters in the site where the multigroup is configured.

When used with the `-f` option, the multigroup is deleted even if the multigroup still contains any protection groups. Protection group status and state are not affected by multigroup deletion.

list

Display defined multigroups on one or more sites where the current cluster is a site controller or site member.

The issuing cluster must be a member of each specified site. This subcommand can be issued from a site controller or a site member.

If you omit the `-s` option, the `list` subcommand displays information for all multigroups that are configured on all sites to which the local cluster belongs.

remove-protection-group

Remove one or more specified protection groups from the specified multigroup. This subcommand must be issued from a site controller.

The system performs this action on the local cluster, then propagates the action to other clusters in the site.

Specify protection groups by using the form *cluster*:*protection-group*. Use a comma-separated list to specify multiple protection groups.

If the protection group to remove is configured in a dependency chain, such as (*cluster*:*pg1*,*cluster*:*pg2*)/*cluster*:*pg3*, specify the entire dependency chain. Then use the `add-protection-group` subcommand to add back those protection groups that you want to continue as part of the multigroup, including any dependency relationship, such as *cluster*:*pg1*/*cluster*:*pg3*.

set-prop

Modify the properties of the specified multigroup. This subcommand must be issued from a site controller. The system performs this action on the local cluster, then propagates the action to other clusters in the site.

Changes are automatically propagated to all clusters in the site where the multigroup exists. If a change is not automatically propagated, use the `validate` subcommand to manually synchronize the changes. If necessary, also use the `update` subcommand.

See the EXTENDED DESCRIPTION section for descriptions of multigroup properties.

show

Display configuration information for one or more specified multigroups in each specified
site. This subcommand can be issued from a site controller or a site member.

start

Start all protection groups in the specified multigroup with the specified scope of global or
local partner clusters. This subcommand must be issued from a site controller.

The -e option specifies the scope for starting the protection groups. For both of the
following scopes, each protection group is started from the cluster that is specified in the
protection-group list of the multigroup:

- With the -e global option, starts each protection group on both partner clusters where
  the protection group is configured
- With the -e local option, starts each protection group only on the local partner cluster

See the WARNINGS section of this man page for information about site synchronization
errors that might interfere with the operation of this subcommand.

If the -f option is used, any protection groups that are already started are skipped. If
a protection group in a dependency chain is already started, the dependencies in the
dependency chain are ignored.

If the -n option is used, data replication is not started.

status

Show the status of one or more specified multigroups to which the issuing cluster belongs.
This subcommand can be issued from a site controller or a site member.

Use a comma-separated list to specify multiple multigroups.

stop

Stop all protection groups of the specified multigroup with the specified scope of global or
local partner clusters. This subcommand must be issued from a site controller.

The -e option specifies the scope for stopping the protection groups. For both of the
following scopes, each protection group is stopped on the cluster that is specified in the
protection-group list of the multigroup:

- With the -e global option, stops each protection group on both partner clusters where
  the protection group is configured
- With the -e local option, stops each protection group only on the local partner cluster

If the -f option is used, any protection groups that are already stopped are skipped. If
a protection group in a dependency chain is already stopped, the dependencies in the
dependency chain are ignored.

If the -D option is used, only data replication is stopped. The application resource groups
are not stopped.

See the WARNINGS section of this man page for information about site synchronization errors that might interfere with the operation of this subcommand.

switchover

Switches over the primary role for all protection groups in the specified multigroup to the specified site. This subcommand must be issued from a site controller cluster where the multigroup is defined.

The protection groups in the multigroup must already be started before you issue the switchover subcommand.

See the WARNINGS section of this man page for information about site synchronization errors that might interfere with the operation of this subcommand.

takeover

Performs a takeover of each protection group from its current primary partner cluster to its standby partner cluster. This operation is not the same as a switchover.

Note - When possible, run the geomg switchover command instead of the geomg takeover command to coordinate between partner clusters for the protection groups, to avoid loss of replicated data.

Run the geomg takeover command only in situations when the primary partner cluster is not available or during unplanned downtime. The situation must justify a potential loss of data and the time required to repair and to reactivate the multigroup on the standby partner cluster in the site.

This subcommand must be issued from a site controller cluster where the multigroup is defined.

Each protection group is brought online on its own standby partner cluster in the site without first synchronizing shutdown on its own primary partner cluster and without considering the state of its primary cluster.

For each protection group in the multigroup, if its standby partner cluster can communicate with its primary partner cluster, the Geographic Edition software switches the role of the standby so that the standby partner cluster becomes the new primary partner cluster. Each protection group in the multigroup is deactivated on its new standby partner cluster.

If used with the -f option, the operation proceeds even if the primary cluster for any protection group in the multigroup is still reachable and the protection group is active on that cluster.

After the geomg takeover command successfully completes, reactivating each protection group in the multigroup on its new standby cluster might require that you recover and synchronize data.

See the WARNINGS section of this man page for information about site synchronization errors that might interfere with the operation of this subcommand.

update

> Resynchronize the multigroup configuration information in the local cluster with multigroup configuration information from the specified remote cluster of the same site. This subcommand can be issued from a site controller or a site member.
>
> When the `update` subcommand is issued, it retrieves the configuration from the specified cluster and updates the configuration information on the local cluster with the retrieved configuration information.
>
> If the local cluster is a site controller, the new local configuration information is propagated to all remote clusters. The remote clusters either accept the new configuration copy or reject it, depending on their synchronization status with the site controller that is propagating the configuration update.
>
> See the geoadm(1M) on page 21 man page for information about the multigroup synchronization statuses.

validate

> Manually validate and synchronize all clusters where the specified multigroup is active. This subcommand can be issued from a site controller or a site member.
>
> If the `validate` subcommand is issued from a site controller cluster, the multigroup configuration is validated and the local copy of the configuration is updated. That updated configuration data is then propagated to the other clusters in the site. The other clusters in the site either reject or accept the new configuration data, depending on whether they are in an `Error` synchronization status with the site controller.
>
> If the `validate` subcommand is issued from a site member cluster, the cluster retrieves validated configuration information from a controller of the site and updates only its own local copy of the configuration data. The configuration data is not propagated to any other site cluster.
>
> The multigroup software automatically calculates and sets an appropriate value for the multigroup `Timeout` property, based on the current `Timeout` property settings for the protection groups in the multigroup.
>
> Use the `validate` subcommand if multigroup changes are not correctly propagated at the time a change is made to the multigroup.
>
> See the geoadm(1M) on page 21 man page for information about the multigroup synchronization statuses.

The following options are supported:

-?
--help

> Displays help information.
>
> You can specify this option with or without a subcommand.

- If you specify this option without a subcommand, the list of subcommands for this command is displayed.
- If you specify this option with a subcommand, the usage options for the subcommand are displayed.

The question mark might be interpreted as a special character by some shells. Use quotes (-"?") or an escape character to avoid pattern matching.

`-D`
`--only-datarep`

Deactivates only data replication. leaving the protection groups active. This option is valid with the `stop` subcommand.

If you do not use this option, the entire protection group is deactivated.

To stop the protection groups in a multigroup that has already had its data replication subsystem stopped, you must run the `geomg stop` command again, without the `-D` option.

`-e {local|global}`
`--scope {local|global}`

Specifies whether the command operates only on each protection group's partner cluster in the local site, using the `local` argument, or on both partner clusters where each protection group has been configured, using the `global` argument. This option is valid with the `start` and `stop` subcommands.

`-f`
`--force`

Forces the command to perform the operation without asking you for confirmation. This option is valid with the `delete`, `start`, `stop`, `switchover`, and `takeover` subcommands.

When used with `geomg switchover -i`, any protection groups that are not yet online on the target partner cluster are switched over, ignoring any dependencies with protection groups that are already online on the target.

When used with `geomg takeover`, the operation proceeds even if the primary cluster for a protection group in the multigroup is reachable and the protection group is active on that cluster.

When used with the `start` or `stop` subcommand, protection groups that are already started or stopped are ignored.

`-i`
`--ignore`

Ignores any protection groups that are already active as primary on their target cluster in the site. This option is valid with the `switchover` subcommand.

If the `-i` option is omitted and any protection groups to switch over are already active as primary on the target site, the switchover operation is refused.

Using the `-i` option might break dependencies between two protection groups. This can happen if, on the target partner cluster, a dependent protection group is already online as primary but the protection group it depends on is not yet online as primary on the site.

Use the `-f` option with the `-i` option to switch over those dependency protection groups that are not yet online as primary on the site but to ignore dependent protection groups that are already online as primary on the site. The dependency order is still followed for any other protection groups that do not have such dependency breakage issues.

`-n`
`--nodatarep`

Specifies to not start data replication when the protection groups in the multigroup are started. This option is valid with the `start` subcommand.

If the `-n` option is omitted, when protection groups in a multigroup are started, data replication also starts.

`-p` *property*
`--property` *property*

Sets a property of the specified multigroup. This option is valid with the `set-prop` subcommand.

A multigroup property is assigned a value by using a *name*=*value* pair statement. You can set multiple properties in a single command by using multiple `-p` statements.

See the EXTENDED DESCRIPTION section for defined properties.

`-s` *site*
`--site` *site*

Specifies the name of the site where the specified multigroup is configured. This option is valid with the `create`, `list`, `show`, `switchover`, and `takeover` subcommands.

`-u`
`--unsynchronized`

Specifies running the operation in an unsynchronized manner. This option is valid with the `switchover` subcommand.

When the `-u` option is used, the `switchover` subcommand does not wait for all protection groups to be stopped on their primary partner clusters outside of the site before starting them on their standby partner clusters within the site.

`-V`
`--version`

Displays the version of the command.

Do not specify this option with subcommands, operands, or other options, as they are ignored. The `-V` option only displays the version of the command. No other operations are performed.

The following operands are supported:

cluster:*protection-group*

> Specifies the name of the protection group you want to administer and the cluster in which it is configured.

multigroup

> Specifies the name of the multigroup that you want to administer. The multigroup name must be unique throughout all clusters of the designated site.

/

> Specifies a strong dependency in the multigroup by two or more protection groups on another protection group, which is specified after a slash.
>
> The following shows the syntax to specify a dependency chain between two protection groups in a multigroup:
>
> *cluster*:*dependent-protection-group*/*cluster*:*depended-on-protection-group*
>
> For example, in the dependency chain `clust1:pg1/clust1:pg2`, if a switchover is issued, `pg1` is stopped on its current primary partner cluster before `pg2` is stopped. Then `pg2` is started on its new primary partner cluster before `pg1` is started.
>
> You can specify more than one /*cluster*: *protection-group* string in a dependency chain to indicate a series of dependencies, such as `clust1:pg1/clust1:pg2/clust1:pg3`.
>
> You can only specify one protection group name after a slash in a dependency chain. You cannot use parentheses to specify multiple protection group names after a slash.

( )

> In a dependency chain, encloses a set of protection groups that depend on another protection group in the same multigroup. The protection group names within the parentheses are separated by commas. The parentheses are followed by a slash (/) and the name of the protection group that is depended on.
>
> The following shows syntax examples of dependency chains that involve more than two protection groups in the multigroup:
>
> (*cluster*:*protection-group1*,*cluster*:*protection-group2*)/*cluster*:*protection-group3*
> *cluster*:*protection-group1*/*cluster*:*protection-group2*/*cluster*:*protection-group3*
>
> For example, in the dependency chain (`clust1:pg1,clust1:pg2`)/`clust1:pg3`, if a switchover is issued, `pg1` and `pg2` are both stopped on the current primary partner cluster before `pg3` is stopped. Then `pg3` is started on the new primary partner cluster before `pg1` and `pg2` are started.

This section contains descriptions of properties that you can specify.

> **Note -** Property names and property values, such as `true` and `false`, are *not* case sensitive.

`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.

Category:

Optional

Default:

None

Tunable:

Assigned at creation and tunable at runtime

`Timeout (integer)`

Specifies the timeout period for the multigroup in seconds. The timeout period is the longest time Geographic Edition waits for a `geomg` command to finish. If the command does not respond within the timeout period, the Geographic Edition software reports the operation as timed out, even if the underlying command eventually completes successfully.

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*

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 a per-operation timeout.

Category:

Optional

Minimum value:

    120 seconds

Maximum value:

    1000000 seconds

Default value:

    120 seconds

Tunable:

    Assigned at creation and tunable at runtime

The following exit values are returned:

`0`   The command completed successfully.

nonzero   An error occurred.

EXAMPLE 13   Creating a Multigroup

The following `geomg` command creates the multigroup `london-paris-mg`. The command is run from a node of the site controller cluster`cluster-soho` in the `site-paris` site.

```
phys-soho-1# geomg create -s site-paris london-paris-mg
```

EXAMPLE 14   Adding a Protection Group to a Multigroup

The following `geomg` command adds the protection groups `sales-pg` and `finance-pg` on the `cluster-soho` cluster to the `london-paris-mg` multigroup. The command configures the `sales-pg` protection group with a strong dependency on the `finance-pg` protection group.

```
phys-soho-1# geomg add-protection-group cluster-soho:sales-pg/cluster-soho:finance-pg \
london-paris-mg
```

EXAMPLE 15   Starting Protection Groups in a Multigroup

The following `geomg` command starts the local member protection groups of the `london-paris-mg` multigroup.

```
phys-soho-1# geomg start -e local london-paris-mg
```

EXAMPLE 16   Performing a Switchover of a Multigroup

The following geomg command switches over all protection groups in the london-paris-mg
multigroup to the site-paris site. The command ignores any protection group that is already
active on the site-paris site. The cluster-louvre cluster is a controller of the site-paris
site.

```
phys-louvre-1# geomg switchover -i -s site-paris london-paris-mg
```

EXAMPLE 17   Performing a Takeover of a Multigroup

The following geomg command performs a takeover by the site-paris site of all protection
groups in the london-paris-mg multigroup. The cluster-louvre cluster is a controller of the
site-paris site.

```
phys-louvre-1# geomg takeover -s site-paris london-paris-mg
```

EXAMPLE 18   Removing a Protection Group That Has No Dependency With Another Protection
                   Group

The following geomg command removes from the london-paris-mg multigroup the hr-pg and
finance-pg protection groups on the cluster-soho cluster. The cluster-soho:hr-pg and
cluster-soho:finance-pg protection groups have no dependency relationships with any other
protection group in the multigroup.

```
phys-soho-1# geomg remove-protection-group cluster-soho:hr-pg,cluster-soho:finance-pg \
london-paris-mg
```

EXAMPLE 19   Removing From a Multigroup a Protection Group That Is in a Dependency Chain

The following geomg command removes the cluster-soho:sales-pg protection group from
the longon-paris-mg multigroup. This protection group has a dependency on the cluster-
soho:finance-pg protection group. Another protection group, cluster-soho:hr-pg, also has a
dependency on the cluster-soho:finance-pg protection group.

The entire interdependency (cluster-soho:hr-pg,cluster-soho:sales-pg)/cluster-
soho:finance-pg is removed from the multigroup. Then the cluster-soho:hr-pg and
cluster-soho:finance-pg protection groups are added back in the dependency cluster-
soho:hr-pg/cluster-soho:finance-pg.

```
phys-soho-1# geomg show london-paris-mg
Multigroup: london-paris-mg
   Description            :
   Site name             : site-paris
   Protection group list : (cluster-soho:hr-pg,cluster-soho:sales-pg)/cluster-soho:finance-
pg
```

```
     Timeout                : 10800

phys-soho-1# geomg remove-protection-group \
(cluster-soho:hr-pg,cluster-soho:sales-pg)/cluster-soho:finance-pg \
london-paris-mg

phys-soho-1# geomg add-protection-group \
cluster-soho:hr-pg/cluster-soho:finance-pg \
london-paris-mg

phys-soho-1# geomg show london-paris-mg
Multigroup: london-paris-mg
   Description            :
   Site name             : site-paris
   Protection group list  : cluster-soho:hr-pg/cluster-soho:finance-pg
   Timeout               : 7200
```

EXAMPLE 20   Modifying a Multigroup

The following geomg command modifies the timeout property of the london-paris-mg
multigroup.

```
phys-soho-1# geomg set-prop -p Timeout=3000 london-paris-mg
```

EXAMPLE 21   Synchronizing Multigroup Configuration Information

The following geomg command resynchronizes configuration information for the london-
paris-mg multigroup on the issuing cluster, cluster-soho, with information from the remote
cluster, cluster-louvre, where the multigroup also exists.

```
phys-soho-1# geomg update cluster-louvre london-paris-mg
```

EXAMPLE 22   Deleting a Multigroup

The following geomg command deletes the london-paris-mg multigroup.

```
phys-soho-1# geomg delete london-paris-mg
```

See attributes(5) for descriptions of the following attributes.

ATTRIBUTE TYPE	ATTRIBUTE VALUE
Availability	ha-cluster/geo/geo-framework
Interface Stability	Evolving

geoadm(1M) on page 21, geohb(1M) on page 33, geopg(1M) on page 55, geops(1M) on page 79, geosite(1M) on page 87, rbac(5)

This section describes the error conditions between the issuing site controller cluster and another cluster in the site that prevent start, stop, switchover, and takeover operations from succeeding.

- If another site controller cluster has an Error synchronization status for the multigroup with the issuing site controller, the operation completely errors out.
- If a site member cluster has an Error synchronization status for the multigroup with the issuing site controller, the operation is not performed on those protection groups in the multigroup that are present on that member cluster.

  However, in the extreme case of a takeover, this rule does not apply. The operation still tries to perform takeover on those protection groups in the multigroup that are present on that member cluster.

- If a cluster in the site has an Unknown synchronization status for the multigroup with the issuing site controller, the operation is not performed on those protection groups in that multigroup that are present on that cluster.

An attempt is still made to perform an operation on those protection groups in the multigroup that are on clusters with an Ok synchronization status.

To help avoid such errors, perform the following checks of the site before you issue a start, stop, switchover, or takeover operation on a multigroup:

- Ensure that the issuing site controller is accepted by all other site clusters for the multigroup as a site controller.
- Ensure that the synchronization status of all site clusters and all multigroups for the site is Ok. The exception is a cluster that is down or on which the Geographic Edition framework is not started, which causes the Unknown status, and you do not intend to perform the operation on the protection groups on that cluster.

Name

geopg — create or manage protection groups

SYNOPSIS

```
/usr/cluster/bin/geopg [subcommand] -?
```

```
/usr/cluster/bin/geopg -V
```

```
/usr/cluster/bin/geopg add-device-group -p property [-p…] component protection-group
```

```
/usr/cluster/bin/geopg add-replication-component -p property [-p…] component protection-group
```

```
/usr/cluster/bin/geopg add-resource-group resource-group[,resource-group…] protection-group
```

```
/usr/cluster/bin/geopg create -s partnership -o local-role [-d data-replication-type]
[-p property [-p…]] protection-group
```

```
/usr/cluster/bin/geopg delete protection-group
```

```
/usr/cluster/bin/geopg get -s partnership [protection-group]
```

```
/usr/cluster/bin/geopg list
```

```
/usr/cluster/bin/geopg modify-device-group component -p property [-p…] protection-group
```

```
/usr/cluster/bin/geopg modify-replication-component component -p property [-p…]
protection-group
```

```
/usr/cluster/bin/geopg remove-device-group component protection-group
```

```
/usr/cluster/bin/geopg remove-replication-component component protection-group
```

```
/usr/cluster/bin/geopg remove-resource-group resource-group[,resource-group…]
protection-group
```

```
/usr/cluster/bin/geopg set-prop -p property [-p…] protection-group
```

```
/usr/cluster/bin/geopg show [protection-group […]]
```

```
/usr/cluster/bin/geopg start -e {local | global} [-n] protection-group
```

```
/usr/cluster/bin/geopg status [protection-group […]]
```

```
/usr/cluster/bin/geopg stop -e {local | global} [-D] protection-group
```

```
/usr/cluster/bin/geopg switchover -m new-primary-cluster [-f] protection-group
```

```
/usr/cluster/bin/geopg takeover [-f] protection-group
```

```
/usr/cluster/bin/geopg update protection-group
```

```
/usr/cluster/bin/geopg validate protection-group
```

The `geopg` command enables you to configure and maintain protection groups. To administer a multigroup of protection groups, instead use the `geomg` command. See the [geomg(1M) on page 39](#) man page.

Before you create a protection group, ensure that the clusters that will host the protection group are already placed in a partnership.

To run the `geopg` command to create, modify, or delete a protection group or a data-replication component, you must be assigned the proper role-based access control (RBAC) rights profile.

If you have `root` access, you have permission to perform any operation. If you do not have root access, the following RBAC rights apply:

- **Basic Solaris User.** You can read information about Oracle Solaris Cluster Geographic Edition (Geographic Edition) entities by running commands such as `geopg list`.
- **Geo Management.** You can perform all the read operations that someone with Basic Solaris User access can perform. You can use commands such as `geopg create`, `geopg switchover`, `geopg start`, and `geopg stop` to perform administrative and configuration operations.

For more information, see the [rbac(5)](#) man page and ["Planning Security" in "Oracle Solaris Cluster Geographic Edition Installation Guide "](#).

The general form of this command is as follows:

```
geopg [subcommand] [options] [operands]
```

You can omit *subcommand* only if *options* specifies the option -?, -o, -V, or -v.

Each option of this command has a long form and a short form. Both forms of each option are given with the description of the option in the OPTIONS section of this man page.

How Protection Groups Start

The `geopg start` command activates the protection group on both the primary and standby clusters. This activation starts the Geographic Edition management of the resource groups in the protection group. Depending on the form of the command that you use and on the cluster where you issue the command, activating the protection group might not start the resource groups on that cluster. The resource groups start, or are brought online, only on the primary cluster.

The -e option defines the scope of the `geopg start` command. If you specify -e `local`, the `geopg start` command runs on the cluster where you issue the command. If you specify -e `global`, the `geopg start` command runs on both clusters in the partnership.

The different forms of the command have the following effects:

- If you run the `geopg start -e local` command on the primary cluster, and if you run the `geopg start -e global` command on either cluster, the Geographic Edition software activates the protection group on both clusters in the partnership. The software brings online resource groups on the primary cluster only. The resource groups are put in a managed state.

- If you run the `geopg start -e local` command on the standby cluster, the Geographic Edition software activates the protection group on the standby cluster only. Resource groups are not started on the standby cluster. The resource groups are put in the `Unmanaged` state.

The following subcommands are supported:

`add-device-group`

Enables a data replication component to be part of a protection group. The system performs this action on the local cluster, then propagates the action to other clusters in the site.

This subcommand functions equivalently to the `add-replication-component` subcommand.

`add-replication-component`

Enables a data replication component to be part of a protection group. The system performs this action on the local cluster, then propagates the action to other clusters in the site.

This subcommand functions equivalently to the `add-device-group` subcommand.

`add-resource-group`

Enables an application resource group to be part of a protection group. The system performs this action on the local cluster, then propagates the action to other clusters in the site.

If a protection group is active when you add a resource group, the resource group must be in either the `Unmanaged` or `Online` state. To start an unmanaged resource group, run the `geopg start` command.

If a protection group is inactive, the resource group must be in the `Unmanaged` state when you add the resource group. If the resource group is offline, bring the resource group to the `Unmanaged` state before adding the resource group to an inactive protection group.

`create`

Creates a protection group within an existing partnership. The system performs this action on only the local cluster.

`delete`

Deletes a protection group. The system performs this action on only the local cluster.

If the protection group is also configured in a multigroup, deletion of the protection group succeeds with a warning message to remove the protection group from the multigroup that

contains it. Use the `geomg remove-protection-group` command to remove a protection group from a multigroup. For more information, see the man page.

get

> Creates the local configuration for a protection group that already exists on the remote partner cluster by retrieving the configuration from the partner cluster. If you do not specify a protection group name, all the protection groups that have been configured for the partnership on the remote cluster will be created on the local cluster. The system performs this action on only the local cluster.

list

> Displays a list of the protection groups in the issuing cluster.

modify-device-group

> Modifies the properties of a data replication component. The system performs this action on the local cluster, then propagates the action to other clusters in the site.
>
> This subcommand functions equivalently to the `modify-replication-component` subcommand.

modify-replication-component

> Modifies the properties of a data replication component. The system performs this action on the local cluster, then propagates the action to other clusters in the site.
>
> This subcommand functions equivalently to the `modify-device-group` subcommand.

remove-device-group

> Removes data replication components from a protection group. The system performs this action on the local cluster, then propagates the action to other clusters in the site.
>
> Removing a data replication component does not stop data replication or change the data replication status for that data replication component.
>
> This subcommand functions equivalently to the `remove-replication-component` subcommand.

remove-replication-component

> Removes data replication components from a protection group. The system performs this action on the local cluster, then propagates the action to other clusters in the site.
>
> Removing a data replication component does not stop data replication or change the data replication status for that data replication component.
>
> This subcommand functions equivalently to the `remove-device-group` subcommand.

remove-resource-group

> Removes resource groups from a protection group. The system performs this action on the local cluster, then propagates the action to other clusters in the site.

Removing an application resource group does not change the application resource group to the `Unmanaged` state.

set-prop

Modifies the properties of a protection group. The system performs this action on the local cluster, then propagates the action to other clusters in the site.

show

Displays the following information about the protection group:

- Defined protection groups
- Resource groups that are wrapped into protection groups
- Operation status information

Use a space-separated list to specify multiple protection groups.

If you do not specify the name of a protection group, the `show` subcommand displays information for all the protection groups configured on the local cluster. The system performs this action on only the local cluster.

start

Starts a protection group. When you start a protection group for the first time, the role of the protection group on a cluster is the role that was assigned to the protection group when you created the protection group. After that, the protection group starts with the latest role it was given.

The `start` subcommand uses Oracle Solaris Cluster commands to bring resources and resource groups under the management of the protection group.

You can start a protection group on the following levels, depending on whether you specify `local` or `global` to the `-e` option:

- With the `-e` `global` option, starts the protection group on all clusters where the protection group has been configured
- With the `-e` `local` option, starts the protection group on either the primary cluster or standby cluster, whichever cluster the command is issued from

For more details about activating a protection group and the effects on the resource groups in that protection group, see "How Protection Groups Start" in the DESCRIPTION section of this man page.

Starting a protection group enables the following events to occur:

- The protection group configuration is validated.
- Clusters can determine whether an operation has been completed.
- Data replication can be started, depending on the data replication that you use.
    - When using Oracle Solaris Availability Suite (Availability Suite) software, you can start data replication from the primary cluster only.

- When using EMC Symmetrix Remote Data Facility (SRDF) software, you can start data replication from either the primary or standby cluster.

- When using the Oracle Data Guard software, you can start data replication from the primary cluster by using the local or the global scope.

For all supported data replication software, you must not specify the -n option when using this form of the command.

- Protected applications can be started.

- If the partner can be reached, the partner cluster can be notified of the protection group activation.

Note - Protection groups that are referred to as "online" are active protection groups.

---

status

Displays the status of the protection group.

Use a space-separated list to specify multiple protection groups.

If you do not specify the name of a protection group, the status subcommand displays the status for all the protection groups configured on the local cluster. The system performs this action on only the local cluster.

stop

Stops a protection group. You can stop a protection group on the following levels, depending on whether you specify local or global to the -e option:

- With the -e global option, stops the protection group on all clusters where the protection group has been configured

- With the -e local option, stops the protection group on either the primary or standby cluster, whichever cluster the command is issued from

Stopping a protection group enables the following events to occur:

- The protection group becomes inactive.

- Data replication and application resource groups stop. You can prevent application resource groups from stopping by taking the resource groups out of the protection group *before* you stop the protection group.

  - When using Availability Suite software, you can stop data replication from the primary cluster only.

  - When using EMC Symmetrix Remote Data Facility (SRDF) software, you can stop data replication from either the primary or standby cluster.

  - When using Oracle Data Guard, you can stop data replication from either the primary or standby cluster, by using either the local or the global scope.

- If the partner cluster is reachable, the partner cluster is notified that the protection group has been stopped.

switchover

> Switches the assigned role of a cluster in the protection group. You must start the protection group before you can perform a switchover. The system performs this action on the local cluster, then propagates the action to other clusters in the site.

takeover

> Forces a cluster to become the `primary` cluster without considering the partner cluster state. The system performs this action on the local cluster, then propagates the action to other clusters in the site.
>
> If you issue a takeover from the standby cluster and the cluster is able to communicate with the partner cluster, the Geographic Edition software switches the role of the partner cluster so that the partner cluster becomes the new standby. This operation is not the same as a switchover. The protection group is deactivated on the new standby.
>
> After the `geopg takeover` command successfully completes, reactivating the protection group on the standby cluster might require that you recover and synchronize data.

Note - When possible, run the `geopg switchover` command instead of the `geopg takeover` command to coordinate between partner clusters to avoid loss of replicated data.

Run the `geopg takeover` command only in situations when the primary cluster is not available or during unplanned downtime. The situation must justify a potential loss of data and the time required to repair and to reactivate the protection group on the standby cluster.

---

update

> Resynchronizes the configuration information of the local protection group with the partner's configuration information. The `geopg update` command overwrites the configuration information of the protection group on the local cluster with the configuration information from the partner cluster. This command does not overwrite replicated data. The system performs this action on only the local cluster.

validate

> Validates a protection group on the local cluster by performing a sanity check of the dynamic values. The system performs this action on only the local cluster.

The following options are supported:

-?
--help

> Displays help information.
>
> You can specify this option with or without a subcommand.
>
> - If you specify this option without a subcommand, the list of subcommands for this command is displayed.

- If you specify this option with a subcommand, the usage options for the subcommand are displayed.

The question mark might be interpreted as a special character by some shells. Use quotes (-"?") or an escape character to avoid pattern matching.

-D
--only-datarep

Specifies that only the data replication should be deactivated, leaving the protection group active. If you do not use this option, the entire protection group is deactivated. To stop a protection group that has already had its data replication subsystem stopped, you must run the geopg stop command again and omit this option.

-d *data-replication-type*
--datarep-type *data-replication-type*

Specifies the data replication mechanism for data replication between the clusters of the protection group.

The *data-replication-type* must be one of the following strings:

avs	Specifies that the data replication mechanism is Availability Suite Remote Mirror, with patches.
odg	Specifies that the data replication mechanism is Oracle Data Guard.
sbp	Specifies that the data replication mechanism is a script-based plug-in.
srdf	Specifies that the data replication mechanism is the dynamic EMC Symmetrix Remote Data Facility (SRDF).

-e {local|global}
--scope {local|global}

Specifies whether the command operates only on the local cluster, local, or on both clusters, global, where the protection group has been configured.

-f
--force

Forces the command to perform the operation without asking you for confirmation.

-m *new-primary-cluster*
--primary *new-primary-cluster*

Specifies the name of the cluster that is to be the new primary cluster for the protection group.

`-n`
`--nodatarep`

> This option is only valid when starting a protection group that is not active. If the protection group is not active, it is started without data replication. If data replication for this protection group is already running independently, it is stopped before the protection group is started.

> If the protection group is already active with data replication enabled, data replication is not stopped. An error message is displayed that states that the `geopg stop -D` command should be used to stop data replication for an active protection group.

> If this option is omitted, data replication starts at the same time as the protection group.

`-o` *local-role*
`--role` *local-role*

> Specifies the role of the local cluster as either `primary` or `secondary` when used with the `create` subcommand. You can change the role of a cluster by running the `geopg switchover` command.

`-p` *property*
`--property` *property*

> Sets the properties of a protection group.

> A protection group property is assigned a value by using a *name*=*statement* pair statement. You can set multiple properties at one time by using multiple statements.

> See the EXTENDED DESCRIPTION section for defined properties.

`-s` *partnership*
`--partnership` *partnership*

> Specifies the name of the existing partnership where the protection group is configured.

`-V`
`--version`

> Displays the version of the command.

> Do not specify this option with subcommands, operands, or other options, as they are ignored. The `-V` option only displays the version of the command. No other operations are performed.

The following operands are supported:

component

> Specifies the name of any data replication component that you want to add. Data replication components include Availability Suite device groups, MySQL data replication components, Oracle Data Guard configurations, Oracle ZFS Storage Appliance replicated projects, and SRDF device groups.

protection-group

> Specifies the name of the protection group. The `create` subcommand fails if the protection group that you specify with this option already exists. The `modify` subcommand fails if the specified protection group is not valid.

resource-group

> Specifies a comma-separated list of resource groups to add to or delete from the protection group. The specified resource groups must already be defined.
>
> If a protection group is active when you add a resource group, the resource group must be in either the `Unmanaged` or `Online` state. To start an unmanaged resource group, run the `geopg start` command.
>
> If the protection group is inactive when you add a resource group, the resource group must be in the `Unmanaged` state before you add the resource group.

This section contains descriptions of general protection group properties and data replication component properties. You can specify general protection group properties with the `geopg set-prop` command You can specify data replication extension properties with the `geopg add-device-group`, `add-replication-component`, `geopg modify-device-group`, and `modify-replication-component` commands.

Note - Property names and property values, such as `true` and `false`, are *not* case sensitive.

General Protection Group Properties

`Description` (string)

> Describes the protection group. The system sets this property on the local cluster, then propagates the value to the partner cluster.

> **Category:**
>
> > Optional

> **Default:**
>
> > None

> **Tunable:**
>
> > Assigned at creation and tunable at runtime

`External_Dependency_Allowed` (boolean)

> Specifies whether to allow any 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. The system sets this property on the local cluster, then propagates the value to the partner cluster.

Category:

Optional

Default:

False

Tunable:

Any time

`RoleChange_ActionArgs` (string)

Specifies the arguments that follow the system-defined arguments that are paired with the role change callback script during a protection group switchover or takeover. The system sets this property on the local cluster, then propagates the value to the partner cluster.

Category:

Optional

Default:

None

Tunable:

Assigned at creation and tunable at runtime

`RoleChange_ActionCmd` (string)

Specifies the absolute path to the executable file to run when the primary cluster of the protection group changes. This file should exist on all nodes of both partner clusters that host the protection group, and have execute permissions for the user that launches the script. This script is run before the application resource group is brought online on the new primary cluster. The system sets this property on the local cluster, then propagates the value to the partner cluster.

Category:

Optional

Default:

None

Tunable:

Assigned at creation and tunable at runtime

`Timeout` (integer)

Specifies, in seconds, the longest time that the Geographic Edition software waits for a response after running a `geopg` command. Examples of responses include `start`, `stop`, `switchover`, and `takeover`. If the command does not return within the `timeout` period,

the Geographic Edition software reports the operation as timed out, even if the underlying command that was run eventually completes successfully.

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:

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

If you set the `Timeout` property to 3000 seconds, the `geopg 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:

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

If the `Timeout` property is set to 3000 seconds, the `geopg 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 protection group `Timeout` value is an estimated value that is applied to some suboperations. 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 is not timed in the `Timeout` period.

If you change the `Timeout` value of a protection group that is part of a multigroup, you must run the `geomg validate` command on that multigroup This command recalculates the multigroup `Timeout` value to include the new `Timeout` value of the modified protection group.

You can specify a maximum of `1000000` (one million) seconds for the `Timeout` property.

Category:

Optional

Default:

3600

Tunable:

Assigned at creation and tunable at runtime

Data Replication Component Properties

This section contains descriptions of the properties that you can specify for each type of data replication product.

EMC Symmetrix Remote Data Facility Protection Group and Data Replication Component Properties

Protection group property: `Cluster_dgs` (stringarray)

> Lists the disk device groups where the data is written. The disk device groups you specify must be of the same type. You can specify more than one disk device group by separating each disk device group with a comma. The system sets this property on the local cluster, then propagates the value to the partner cluster.
>
> To specify Solaris Volume Manager disk sets when a disk set name is not the same on both clusters, use the following syntax which includes the cluster names:
>
> `Cluster_dgs=`*local-cluster*`[`*disk-set*`]:`*remote-cluster*`[`*disk-set*`]`
>
> Include the brackets ([ ]) around all disk set names for each cluster. Separate multiple disk set names with a comma. For example:
>
> `Cluster_dgs=paris[dg1,dg2]:newyork[dg10,dg11]`
>
> **Category:**
>
> > Optional
>
> **Default:**
>
> > Empty
>
> **Tunable:**
>
> > Only when the protection group is inactive on both partner clusters

Data replication component property: `DG_or_CG` (string)

> Specifies the type of SRDF component: device group (`DG`) or consistency group (`CG`). You must set this property to `DG`. The system sets this property on the local cluster, then propagates the value to the partner cluster.
>
> **Category:**
>
> > Optional
>
> **Default:**
>
> > DG
>
> **Tunable:**
>
> > Never

Protection group property: `Nodelist` (`stringarray`)

>   Lists the host names of the machines that can be primary for the replication mechanism. This list is comma delimited. The system sets this property value on only the local cluster.

>   **Category:**

>   >   Optional

>   **Default:**

>   >   Empty

>   **Tunable:**

>   >   Any time

Data replication component property: `R1_SID` (`integer`)

>   Specifies the Symmetrix Remote Data Facility (SRDF) identification of the primary devices. The value also set on the partner cluster as the `R2_SID` value.

>   ---
>   **Note -** If you do not specify a value for this property when you add a device group, the Geographic Edition framework searches for the value and sets the property automatically.
>   ---

>   **Category:**

>   >   Required

>   **Default:**

>   >   None, until you add a Symmetrix Remote Data Facility (SRDF) device group

>   **Tunable:**

>   >   Any time

Data replication component property: `R2_SID` (`integer`)

>   Specifies the Symmetrix Remote Data Facility (SRDF) identification of the secondary devices. The value is also set on the partner cluster as the `R2_SID` value.

>   ---
>   **Note -** If you do not specify a value for this property when you add a device group, the Geographic Edition framework searches for the value and sets the property automatically.
>   ---

>   **Category:**

>   >   Required

>   **Default:**

>   >   None, until you add a Symmetrix Remote Data Facility (SRDF) device group

>   **Tunable:**

>   >   Any time

MySQL Data Replication Component Properties

Replication component property: `switchover_args`

    Specifies the command-line arguments for the switchover script.

Replication component property: `takeover_args`

    Specifies the command-line arguments for the takeover script.

Replication component property: `start_replication_args`

    Specifies the command-line arguments for the `start_replication` script

Replication component property: `remove_config_args`

    Specifies the command-line arguments for the `remove_configuration` script.

Replication component property: `create_config_args`

    Specifies the command-line arguments for the `create_configuration` script.

Replication component property: `stop_replication_args`

    Specifies the command-line arguments for the `stop_replication` script.

Oracle Data Guard Data Replication Component Properties

Data replication component property: `local_database_name` (string)

    Name of the local Oracle database in the Oracle Data Guard Broker configuration that is being replicated to the remote cluster. This name is the Oracle `db_unique_name` initialization parameter for the Oracle RAC database on the local cluster. The name is also set on the partner cluster as the `remote_database_name` value.

    Category:

        Required

    Default:

        None

    Tunable:

        At creation

Data replication component property: `local_db_service_name` (string)

    Oracle net service name that is used to connect to the local Oracle database. The name is also set on the partner cluster as the `remote_db_service_name` value.

    Category:

        Required

Default:

None

Tunable:

Any time

Data replication component property: `local_rac_proxy_svr_rg_name` (string)

Name of the local Oracle RAC server proxy resource group that manages the local database in the Oracle Data Guard Broker configuration. The name is also set on the partner cluster as the `remote_rac_proxy_svr_rg_name` value.

Category:

Required

Default:

None

Tunable:

At creation

Data replication component property: `remote_database_name` (string)

Name of the remote database in the Oracle Data Guard Broker configuration that is being replicated from the local cluster. This name is the Oracle `db_unique_name` initialization parameter for the Oracle RAC database on the remote cluster. The name is also set on the local cluster as the `local_database_name` value.

Category:

Required

Default:

None

Tunable:

At creation

Data replication component property: `remote_db_service_name` (string)

Oracle net service name that is used to connect to the remote Oracle database. The name is also set on the local cluster as the `local-db_service_name` value.

Category:

Required

Default:

None

Tunable:

    Any time

Data replication component property: `remote_rac_proxy_svr_rg_name` (string)

Name of the remote Oracle RAC server proxy resource group on the partner cluster that manages the remote database in the Oracle Data Guard Broker configuration. The name is also set on the local cluster as the `local_rac_proxy_svr_rg_name` value.

Category:

    Required

Default:

    None

Tunable:

    At creation

Data replication component property: `replication_mode` (string)

The Oracle Data Guard replication mode between the primary database and the standby database. The system sets this property on the local cluster, then propagates the value to the partner cluster.

Valid values to which you set this property include `maxAvailability`, `maxPerformance`, and `maxProtection`.

Category:

    Required

Default:

    None

Tunable:

    Any time

Data replication component property: `standby_type` (string)

Type of Oracle standby database that is used in the Oracle Data Guard Broker configuration. The system sets this property on the local cluster, then propagates the value to the partner cluster.

Valid values to which you set this property include `logical` and `physical`.

Category:

    Required

Default:

    None

Tunable:

> At creation

Data replication component property: `sysdba_password` (string)

> Password for the Oracle `SYSDBA` privileged database user. The system sets this property on the local cluster, then propagates the value to the partner cluster.
>
> Do not specify a password on the command line. If you specify only `-p sysdba_password=`, the geopg command prompts you to type an actual password, which is not displayed as you type it.

> **Category:**
>
> > Required

> **Default:**
>
> > None

> **Tunable:**
>
> > Any time

Data replication component property: `sysdba_username` (string)

> Name of an Oracle `SYSDBA` privileged database user who can perform the Oracle Data Guard Broker switchover and takeover operations on both the primary and standby clusters. The Oracle Solaris Cluster Geographic Edition software uses this property to monitor and manage the Oracle Data Guard Broker configurations. The system sets this property on the local cluster, then propagates the value to the partner cluster.

> **Category:**
>
> > Required

> **Default:**
>
> > None

> **Tunable:**
>
> > Any time

Oracle Solaris Availability Suite Data Replication Component Properties

Device group property: `Enable_volume_set` (boolean)

> Determines whether the volume sets that are defined in the `/var/cluster/geo/avs/`*AVS-device-group*`-volset.ini` file, and the fallback snapshots that are defined in the `/var/cluster/geo/avs/`*AVS-device-group*`-snapshot.ini` file are to be enabled by the Oracle Solaris Cluster Geographic Edition framework. You can set this property to `True` or to `False`. The system sets this property on the local cluster, then propagates the value to the partner cluster.

If the enable_volume_set property is set to True, volume sets that are defined in the /var/cluster/geo/avs/*AVS-device-group*-volset.ini file are enabled when the device group is added to the protection group or when the protection group that contains the device group is replicated from the partner cluster. Any volume sets of that device group that are not defined in the file are disabled. The /var/cluster/geo/avs/*AVS-device-group*-volset.ini file must exist on every node of the primary and standby cluster.In addition, if the enable_volume_set property is set to True, the fallback snapshots defined in the/var/cluster/geo/avs/ *AVS-device-group*-snapshot.ini file are enabled. A fallback snapshot is a compact dependent shadow volume that is created immediately prior to the resynchronization of a secondary replicated volume in the Oracle Solaris Cluster Geographic Edition framework. If resynchronization fails, the secondary replicated volume can be reconstructed by using the fallback snapshot. The /var/cluster/geo/avs/*AVS-device-group*-snapshot.ini file must exist on any node of a cluster on which a fallback snapshot will be automatically enabled. Fallback snapshot configuration is optional.

If the Enable_volume_set property is set to False, you must manually enable volume sets and fallback snapshots by using the Availability Suite commands.

The parameters in the volume set file that are handled by the Geographic Edition framework are described in the following table.

Note - The Geographic Edition framework does not handle other parameters of the volume set file, such as, size of memory queue, and number of asynchronous threads. You must adjust these parameters manually by using Availability Suite commands.

Parameter	Description
phost	Primary host. The logical host of the server on which the primary volume resides.
pdev	Primary device. Primary volume partition. Specify full path names only.
pbitmap	Primary bitmap. Volume partition in which the bitmap of the primary partition is stored. Specify full path names only.
shost	Secondary host. The logical host of the server on which the secondary volume resides.
sdev	Secondary device. Secondary volume partition. Specify full path names only.
sbitmap	Secondary bitmap. Volume partition in which the bitmap of the secondary partition is stored. Specify full path names only.
ip	Network transfer protocol. IP address.
sync | async	Operating mode. ■ In sync mode the I/O operation is confirmed as complete only when the volume on the standby cluster has been updated. ■ In async mode the primary host I/O operation is confirmed as complete before updating the volumes on the standby clusters.
g iogroupname	An I/O group name. The set must be configured in the same I/O group on both the primary and standby clusters. This parameter is optional and need only be configured if you have an I/O group.

`q qdev`	Disk queue volume. Full path name for the volume to be used as a disk-based I/O queue for an asynchronous disk set.
`C devicegroupname`	Device group name. The device group name or resource tag of the local data and bitmap volumes in cases where this information is not implied by the name of the volume. For example, `/dev/md/avsset/rdsk/vol` indicates a device group named `avsset`. It can be Solaris Volume Manager volumes or raw device volumes.

The parameters required to enable a fallback snapshot for a replicated volume are shown in the following example line from a *AVS-device-group*-`snapshot.ini` configuration file. This line defines one compact dependent shadow volume that is to be created immediately prior to a resynchronization of the listed secondary volume. Each line of the *AVS-device-group*-`snapshot.ini` file configures a fallback snapshot for one secondary volume in the device group in the format shown in the following example:

```
/dev/md/avsset/rdsk/d100 /dev/md/avsset/rdsk/d102 /dev/md/avsset/rdsk/d103
```

Each line in the *AVS-device-group*-`snapshot.ini` file contains the following types of entries:

- `/dev/md/avsset/rdsk/d100` – Secondary replicated volume
- `/dev/md/avsset/rdsk/d102` – Fallback snapshot volume
- `/dev/md/avsset/rdsk/d102` – Fallback snapshot bitmap

Category:

Optional

Default:

`False`

Tunable:

Until you add the device group to a protection group

Device group property: `Local_logical_host` (string)

Defines the local logical host name that is used for the replication of the disk device group. Do not use an underscore character (_) in the logical host name. The name is also set on the partner cluster as the `Remote_logical_host` value.

Category:

Required

Default:

None

Tunable:

Until you add the device group to a protection group

Protection group property: `Nodelist` (`stringarray`)

> Lists the host names of the machines that can be primary for the replication mechanism. You can specify more than one host name by separating each host name with a comma. The system sets this property value on only the local cluster.

> **Category:**
>> Optional

> **Default:**
>> All nodes in the cluster

> **Tunable:**
>> Only when the protection group is inactive

Device group property: `Remote_logical_host` (`string`)

> Specifies the remote logical host name that is used for the replication of the disk device group. The name is also set on the partner cluster as the `Local_logical_host` value.

> **Category:**
>> Required

> **Default:**
>> None

> **Tunable:**
>> Until you add the device group to a protection group

Oracle ZFS Storage Appliance Replication Component Properties

Data replication component property: `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.

Data replication component property: `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.

The following exit values are returned:

`0`   The command completed successfully.

nonzero   An error occurred.

EXAMPLE 23   Adding an Oracle Data Guard Broker Configuration to a Protection Group

The following geopg command adds the Oracle Data Guard Broker configuration mysales.com to the protection group sales-pg.

To ensure security, do *not* supply a password when you specify the sysdba_password property. If you do not provide a password, the geopg command prompts for one.

```
phys-paris-1# geopg add-replication-component \
-p local_database_name=sales \
-p remote_database_name=salesdr \
-p local_db_service_name=sales-svc \
-p remote_db_service_name=salesdr-svc \
-p standby_type=physical \
-p replication_mode=MaxPerformance \
-p sysdba_username=sys \
-p sysdba_password= \
-p local_rac_proxy_svr_rg_name=sales-rac-proxy-svr-rg \
-p remote_rac_proxy_svr_rg_name=salesdr-rac-proxy-svr-rg \
mysales.com sales-pg

Oracle Data Guard configuration "mysales.com" successfully added to the
protection group "sales-pg"
```

EXAMPLE 24   Creating a Protection Group That Uses Data Replication

The following geopg command creates a Availability Suite protection group avspg on the primary cluster cluster-paris.

```
# geopg create -s paris-newyork-ps -d avs -o primary \
-p Nodelist=phys-paris-1,phys-paris-2 avspg
```

EXAMPLE 25   Creating a Protection Group That Does Not Use Data Replication

The following geopg command creates a protection group example-pg that is not configured to use data replication.

```
# geopg create -s paris-newyork-ps -o primary example-pg
```

EXAMPLE 26   Deleting a Protection Group While Keeping Application Resource Groups Online

The following geopg command keeps two application resource groups, apprg1 and apprg2, online while deleting their protection group, avspg. Remove the application resource groups from the protection group, then delete the protection group.

```
# geopg remove-resource-group apprg1,apprg2 avspg
# geopg stop -e global avspg
```

```
# geopg delete avspg
```

EXAMPLE 27   Modifying a Protection Group

The following `geopg` command modifies the timeout property of the protection group `avspg`.

```
# geopg set-prop -p Timeout=3000 avspg
```

EXAMPLE 28   Adding a Data Replication Device Group to an Availability Suite Protection Group

The following `geopg` command creates an Availability Suite data replication device group in the `avspg` protection group.

```
# geopg add-replication-component -p Local_logical_host=lh-paris-1 \
-p Remote_logical_host=lh-newyork-1 -p Enable_volume_set=True avsset avspg
```

If the command returns successfully, volume sets that are defined in the `/var/cluster/geo/ avs/avsset-volset.ini` file are enabled. The file might look like this for a Solaris Volume Manager device group.

```
lh-paris-1 /dev/md/avsset/rdsk/d100 /dev/md/avsset/rdsk/d101
lh-newyork-1 /dev/md/avsset/rdsk/d100 /dev/md/avsset/rdsk/d101
ip async q /dev/md/avsset/rdsk/d102 C avsset
```

The description of the parameters is as follows:

- `lhparis-1` – Primary host
- `/dev/md/avsset/rdsk/d100` – Primary data
- `/dev/md/avsset/rdsk/d101` – Primary bitmap
- `lh-newyork-1` – Secondary host
- `/dev/md/avsset/rdsk/d100` – Secondary data
- `/dev/md/avsset/rdsk/d101` – Secondary bitmap
- `ip` – Protocol
- `async` – Mode
- `q /dev/md/avsset/rdsk/d102` – disk queue
- `C` – C tag
- `avsset` – Device group

See `attributes`(5) for descriptions of the following attributes.

ATTRIBUTE TYPE	ATTRIBUTE VALUE
Availability	ha-cluster/geo/geo-framework

ATTRIBUTE TYPE	ATTRIBUTE VALUE
Interface Stability	Evolving

geoadm(1M) on page 21, geohb(1M) on page 33, geomg(1M) on page 39, geops(1M) on page 79, rbac(5)

Name

geops — create, configure, and manage partnerships between clusters

/usr/cluster/bin/geops [*subcommand*] -?

/usr/cluster/bin/geops -V

/usr/cluster/bin/geops add-trust -c *remote-partner-cluster*

/usr/cluster/bin/geops create -c *remote-partner-cluster* [-h *heartbeat*]
[-p *property* [-p…]] *partnership*

/usr/cluster/bin/geops join-partnership *remote-cluster* [-h *heartbeat*] *partnership*

/usr/cluster/bin/geops leave-partnership *partnership*

/usr/cluster/bin/geops list

/usr/cluster/bin/geops remove-trust -c *remote-partner-cluster*

/usr/cluster/bin/geops set-prop -p *property* [-p…] *partnership*

/usr/cluster/bin/geops show [*partnership* […]]

/usr/cluster/bin/geops status [*partnership* […]]

/usr/cluster/bin/geops update *partnership*

/usr/cluster/bin/geops verify-trust [-c *remote-partner-cluster*]

The geops command enables you to create, configure, and manage the partnerships that are defined between clusters. A partnership is a pair of clusters that define a cluster infrastructure on which an application might be protected against disaster.

Two clusters must be configured to trust each other before you create a partnership between two clusters.

A partnership requires a running heartbeat between clusters. Partner clusters monitor each other with heartbeats.

The Oracle Solaris Cluster Geographic Edition software notifies all remote partners of changes in the partnership configuration.

Run the geops command on a cluster that has been enabled for partnership.

To run the geops command to create, configure, and manage the partnerships, you must be assigned the proper role-based access control (RBAC) rights profile.

If you have `root` access, you have permissions to perform any operation. If you do not have root access, the following RBAC rights apply:

- **Basic Solaris User.** You can read information about Oracle Solaris Cluster Geographic Edition entities by running commands such as `geop2 list`.
- **Geo Management.** You can perform all the read operations that someone with Basic Solaris User access can perform. You can also perform administrative and configuration tasks such as `geops create`, `geops join-partnership`, and `geops update`.

Note - You must have `root` access to run the following subcommands:

- `geops add-trust`
- `geops remove-trust`
- `geops verify-trust`

For more information, see the `rbac`(5) man page and "Planning Security" in "Oracle Solaris Cluster Geographic Edition Installation Guide ".

The general form of this command is as follows:

`geops` [*subcommand*] [*options*] [*operands*]

You can omit *subcommand* only if *options* specifies the option -?, -o, -V, or -v.

Each option of this command has a long form and a short form. Both forms of each option are given with the description of the option in the OPTIONS section of this man page.

The following subcommands are supported:

`add-trust`

Retrieve public keys of a remote cluster and add the keys to the `truststore` file on every node of the local cluster. The `truststore` file is at `/etc/cacao/instances/default/security/jsse/truststore`. The system performs this action on only the local cluster.

The new key is not added to the `truststore` file if an entry with the same alias already exists in the `truststore` file on a node. If the existing entry contains a key that is different from the new key, you must delete the entry from the `truststore` file before you add the new key.

`create`

Create a partnership between the local cluster and a remote cluster. The two clusters must be configured to trust each other. The system performs this action on only the local cluster.

After the local cluster issues the `create` subcommand, the remote cluster must issue the `join-partnership` subcommand to complete creation of the partnership.

join-partnership

> Enable the local cluster to join an existing partnership. The system performs this action on the local cluster, then propagates the action to other clusters in the site.
>
> Before issuing the `join-partnership` subcommand, the partner cluster must issue the `create` subcommand to initiate creation of a new partnership.

leave-partnership

> Remove the local cluster from a partnership. The system performs this action on the local cluster and notifies the partner cluster that the local cluster has left the partnership. This subcommand deletes the partnership when the last participating cluster in the partnership leaves.

list

> Displays a list of partnerships in the local cluster.

remove-trust

> Remove all keys for the remote cluster from the `truststore` file on the local node. You must remove all keys for the remote cluster from the local host before you can add a new key. The system performs this action on only the local cluster.
>
> To remove a key from a cluster, dissolve the partnership between the two clusters (by running `geops leave` on each cluster) and run the `remove-trust` subcommand from each node of the cluster.

set-prop

> Modify the properties of a partnership. This subcommand updates the local cluster configuration and notifies remote partner clusters of the configuration change.

show

> Display configuration information for one or more partnerships in the local cluster.
>
> Use a space-separated list to specify multiple partnerships.
>
> The system performs this action on only the local cluster.

status

> Display status information for one or more partnerships in the local cluster.
>
> Use a space-separated list to specify multiple partnerships.
>
> The system performs this action on only the local cluster.

update

> Synchronize information with the partner cluster. The `geops update` command overwrites the configuration on the local cluster with the configuration from the remote partner cluster.
>
> This subcommand cannot update a partnership while the cluster is disconnected from the partner cluster.

If the remote partner cluster is down, modifications to the local cluster might not be propagated to the remote partner cluster.

`verify-trust`

Retrieve node information on the remote cluster and establish a secure Java Management Extensions (JMX) connection from the local host to every node of the remote cluster. The `verify-trust` subcommand verifies that the certificates have been exchanged properly between the remote and the local clusters.

If you do not specify a remote cluster, the `verify-trust` subcommand verifies the connections between the local host and other nodes of the local cluster.

The following options are supported:

`-?`
`--help`

Displays help information.

You can specify this option without a subcommand or with a subcommand.

- If you specify this option without a subcommand, the list of subcommands for this command is displayed.
- If you specify this option with a subcommand, the usage options for the subcommand are displayed.

The question mark might be interpreted as a special character by some shells. Use quotes (`-"?"`) or an escape character to avoid pattern matching.

`-c` *remote-cluster*
`--cluster` *remote-cluster*

Specifies the logical hostname of the cluster with which to form a partnership. The logical hostname is used by the Oracle Solaris Cluster Geographic Edition software and maps to the name of the remote partner cluster. For example, a remote partner cluster name might resemble the following:

`cluster-paris`

When you use this option with the `add-trust`, `remote-trust`, or `verify-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 must have the following pattern:

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

Keys and only keys that belong to the remote cluster should have their alias match this pattern.

> **Note -** The `remove-trust` subcommand removes only keys you have added with the `add-trust` subcommand, and keys with an alias that matches the following pattern:
>
> *remote-partner-cluster*`.certificate[0-9]*`

`-h` *heartbeat*
`--heartbeat-name` *heartbeat*

> Specifies an identifier for the heartbeat on a partner cluster that the local cluster can use to monitor partner availability. You must create the heartbeat by running the `geohb` command before you specify the heartbeat in the `geops` command.
>
> If this option is omitted, the Oracle Solaris Cluster Geographic Edition software uses the default heartbeat mechanism between member clusters.

`-p` *property*
`--property` *property*

> Specifies the value of partnership properties. You can specify multiple properties at one time by using multiple statements.
>
> See the EXTENDED DESCRIPTION section for a description of the properties.

`-V`
`--version`

> Displays the version of the command.
>
> Do not specify this option with subcommands, operands, or other options. The subcommands, operands, or other options are ignored. The `-V` option only displays the version of the command. No other operations are performed.

The following operand is supported:

partnership   Specifies the name of the partnership

The following section lists the partnership properties. These properties are set at creation and tunable at runtime.

Partnership Properties

`Description`

> Describes the partnership.The system sets this property on the local cluster, then propagates the value to the partner cluster.
>
> Optional property.
>
> Type: String.
>
> Default value: Empty string.

`Notification_ActionCmd`

Specifies the path to the script or command that is triggered when a heartbeat-loss notification is issued. The system sets this property on the local cluster, then propagates the value to the partner cluster.

Optional property.

Type: String.

Default value: Empty string.

The Oracle Solaris Cluster Geographic Edition software enables you to specify a command to run when a heartbeat-loss notification is issued. You can specify the path to the command by using the `Notification_ActionCmd` property. The command runs with root permissions, so the file must have root ownership and execution permissions. If both `Notification_ActionCmd` and `Notification_EmailAddrs` properties have been configured, an email is sent after the command that is specified in the `Notification_ActionCmd` property is run.

> ⚠️ **Caution -** A heartbeat loss indicates only a lost connection to a remote cluster and not the cause for the lost connection. A remote cluster that has lost a heartbeat connection might still be functioning even though the heartbeat monitoring has lost its connection.
>
> Therefore, you must exercise caution in the script you specify for the `Notification_actioncmd` parameter. For example, if you specify a forced takeover or a switchover in the `Notification_actioncmd` script, a takeover or a switchover might be run even though one might not be necessary.

`Notification_EmailAddrs`

Specifies the email address to send messages to when a heartbeat-loss notification is issued. You can specify multiple email addresses by separating each email address with a comma. The system sets this property on the local cluster, then propagates the value to the partner cluster.

Optional property.

Type: String array.

Default value: Empty string.

The `Notification_EmailAddrs` and the `Notification_ActionCmd` properties enable notification in heartbeat-loss events. Heartbeat-loss events are detected locally on each cluster of the partnership, and the notification is triggered locally on the cluster where the event is detected. The email addresses and the notification action path should be valid on each cluster in the partnership.

The following exit values are returned:

`0`   The command completed successfully.

nonzero   An error has occurred.

EXAMPLE 29   Adding Trust Between Two Clusters

The following geops commands add trust between the clusters cluster-paris and cluster-newyork.

1. Add cluster cluster-newyork to the trust on the cluster cluster-paris.

    ```
    cluster-paris# /usr/cluster/bin/geops add-trust -c cluster-newyork
    ```

    Keys are stored on every node of cluster cluster-paris with alias cluster-newyork.certificate*number*.

2. Add cluster cluster-paris to the trust on the cluster cluster-newyork.

    ```
    cluster-newyork# /usr/cluster/bin/geops add-trust -c cluster-paris
    ```

3. Verify trust between the two clusters.

    ```
    cluster-paris# /usr/cluster/bin/geops verify-trust -c cluster-newyork
    ```

    ```
    cluster-newyork# /usr/cluster/bin/geops verify-trust -c cluster-paris
    ```

EXAMPLE 30   Creating a Partnership

The following geops command creates the partnership paris-newyork-ps, on the cluster cluster-paris.

```
cluster-paris# geops create -c cluster-newyork -p Description=Transatlantic \
-p Notification_emailaddrs=sysadmin@companyX.com paris-newyork-ps
```

EXAMPLE 31   Joining a Partnership

The following geops command joins cluster-paris and cluster-newyork in the paris-newyork-ps partnership.

```
cluster-paris# geops join-partnership cluster-newyork paris-newyork-ps
```

EXAMPLE 32   Modifying a Partnership

The following geops command modifies the notification email address for cluster-paris.

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

EXAMPLE 33   Leaving a Partnership

The following geops command removes the cluster cluster-paris from the partnership paris-newyork-ps.

```
cluster-paris# geops leave-partnership paris-newyork-ps
```

EXAMPLE 34   Resynchronizing a Partnership

The following geops command resynchronizes the partnership paris-newyork-ps.

The geops update command overwrites the configuration on the local cluster with the configuration from the remote partner cluster.

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

See attributes(5) for descriptions of the following attributes.

ATTRIBUTE TYPE	ATTRIBUTE VALUE
Availability	ha-cluster/geo/geo-framework
Interface Stability	Evolving

rbac(5)

Name

geosite — create, configure, and manage cluster sites

```
/usr/cluster/bin/geosite [subcommand] -?

/usr/cluster/bin/geosite -V

/usr/cluster/bin/geosite accept cluster site

/usr/cluster/bin/geosite add-member {-c cluster | -m cluster}
     [-c cluster[,…]] [-m cluster[,…]] site

/usr/cluster/bin/geosite create {-c cluster | -m cluster} [-c cluster[,…]]
     [-m cluster[,…]] [-p name=value] [-p …] site

/usr/cluster/bin/geosite join cluster site

/usr/cluster/bin/geosite leave [-f] site

/usr/cluster/bin/geosite list [site […]]

/usr/cluster/bin/geosite remove-member cluster[,…] site

/usr/cluster/bin/geosite set-prop -p name=value [-p …] site

/usr/cluster/bin/geosite set-role {-c cluster | -m cluster} site

/usr/cluster/bin/geosite show [-v] site

/usr/cluster/bin/geosite status site […]

/usr/cluster/bin/geosite update cluster site

/usr/cluster/bin/geosite validate site
```

The `geosite` command enables you to create, configure, and manage a group of clusters, called a *site*. Clusters in a site can consist of physical or virtual machines, called global clusters, zone clusters, or a combination. You can perform start, stop, switchover, or takeover operations on a specified group of protection groups, called a *multigroup*, in all clusters of a specified site at one time. For more information about multigroups, see the `geomg`(1M) on page 39 man page.

Under normal circumstances, a change of site membership or a change to a site property is automatically propagated to all clusters in the site. You can use the `validate` subcommand if a change is not correctly propagated after such an operation.

To create a site, you must be superuser or be assigned the proper role-based access control (RBAC) rights profile. To manage a site, you must also issue the command from a cluster that has the controller role for the specified site.

If you have `root` access, you have permissions to perform any operation. If you do not have root access, the following RBAC rights apply:

- **Basic Solaris User.** You can read information about Oracle Solaris Cluster Geographic Edition entities by running commands such as `geosite list`.
- **Geo Management.** You can perform all the read operations that someone with Basic Solaris User access can perform. You can also perform administrative and configuration tasks such as `geosite create`, `geosite set-role`, `geosite set-prop`, and `geosite update`.

For more information, see the `rbac`(5) man page and "Planning Security" in "Oracle Solaris Cluster Geographic Edition Installation Guide".

The general form of this command is as follows:

`geosite` [*subcommand*] [*options*] [*operands*]

You can omit *subcommand* only if *options* specifies the option `-?` or `-V`.

Each option of this command has a long form and a short form. Both forms of each option are given with the description of the option in the OPTIONS section of this man page.

The following subcommands are supported:

`accept`

Accept the specified cluster as a new controller of the specified site when no existing controller cluster is available in the site.

This subcommand is used if the `set-role` subcommand was issued to propose a site member as a site controller at a time when all existing site controller clusters were unreachable.In this situation, each site member cluster must issue this subcommand to accept the proposal of the cluster to be made a site controller.

The candidate cluster becomes a site controller for each site member cluster that issues the `accept` subcommand and specifies the candidate cluster. If a site member cluster does not accept the candidate cluster as a site controller, any multigroup command that is issued by the candidate cluster is not performed on the nonaccepting cluster.

`add-member`

Invite a specified cluster to become a member or controller of the specified site. You must issue this subcommand from an existing site controller.

The invited cluster cannot be in the same partnership as the issuing cluster.

The invited cluster must issue the `join` subcommand to accept the invitation and be added to the site. See the description of the `join` subcommand for more information.

`create`

Create a site of one or more clusters.

This command must be run from one cluster of a partnership. The issuing cluster is automatically configured as the site's first controller. To avoid a possible single point of failure, specify at least one additional cluster to be configured as a site controller.

After the site is created, the clusters that were specified to add to the site must each run the `join` command, to accept the invitation to join the site. All nodes of the cluster that issues the invitation must be online for the new member or controller to be successfully added to the site.

`join`

Accept an invitation to join the specified site.

A site controller must first issue the `create` or `add-member` subcommand to invite the new cluster.

The `join` subcommand must be issued from the invited cluster and specify the name of a site controller cluster of the site to join. At the time that the `join` subcommand is issued from the invited cluster, the following requirements must be met:

- All nodes of the site cluster that issues the invitation must be online.
- All nodes of the cluster that is attempting to join the site must be online.
- The common agent container must be enabled on all nodes of both the new site cluster and the site cluster that issued the invitation.

If any of these requirements are not met, the `join` operation does not succeed.

`leave`

Detach the issuing cluster from the specified site. When the last cluster in a site is removed, the site is automatically deleted as well.

This subcommand can be run by a site controller or a site member.

The following actions are required when the issuing cluster is the last cluster in the site. If these actions are not performed, the operation fails with an error.

- If the last cluster to leave is a site controller, the `leave` subcommand must include the `-f` option.
- If the last cluster to leave is a site member, first use the `set-role` subcommand to change the cluster to a site controller. Then issue the `leave` subcommand with the `-f` option.

If a cluster becomes permanently unavailable so that the `geosite leave` command cannot be run from that cluster, use the `remove-member` subcommand from a site controller to detach the unresponsive cluster.

`list`

Display a list of the sites to which the issuing cluster belongs. Use a space-separated list to specify multiple sites.

This subcommand can be run from a site controller or a site member.

`remove-member`

Remove one or more specified clusters from the specified site.

This subcommand must be run from a site controller.

This subcommand is used when a cluster is unavailable and cannot remove itself by issuing the `leave` subcommand.

To specify multiple sites, use a comma-separated list.

set-prop

Modifies a property of a site. See EXTENDED DESCRIPTION for the list of site properties.

set-role

Set the site role of a cluster to either controller or member.

This subcommand must be run from a site controller.

If no site controller is reachable, a site member can issue this subcommand to submit itself or another member of the site as a candidate site controller. In this situation, all other site clusters must issue the `accept` subcommand before the candidate cluster is confirmed as a site controller.

show

Display site configuration information for one or more specified sites to which the issuing cluster belongs.

This subcommand can be run from a site controller or a site member.

If used with the `-v` option, the protection-group list of each specified multigroup is also displayed.

status

Display status information for one or more specified sites to which the issuing cluster belongs. Use a space-separated list to specify multiple sites.

This subcommand can be run from a site controller or a site member.

update

Resynchronize the site configuration information in the local cluster with information from the specified remote cluster of the same site.

This subcommand can be issued by a site controller or a site member.

validate

Manually verify and synchronize the configuration of the specified site.

This subcommand can be run from a site controller or a site member. The differences in outcome for each site role are as follows:

- If the subcommand is issued by a site controller, the subcommand contacts all clusters in the site to verify the consistency of the configuration information.
- If the subcommand is issued by a site member, the subcommand contacts a site controller and requests an updated copy of the site configuration.

The following options are supported:

`-?`
`--help`

Displays help information.

- If you specify this option without a subcommand, the list of subcommands for this command is displayed.
- If you specify this option with a subcommand, the usage options for the subcommand are displayed.

The question mark might be interpreted as a special character by some shells. Use quotes (`-"?"`) or an escape character to avoid pattern matching.

`-c` *cluster*
`--controller-cluster` *cluster*

Specifies the name of one or more clusters to configure as a site controller. This option is valid with the `add-member`, `create`, and `set-role` subcommands.

Use a comma-separated list to specify multiple clusters.

`-f`
`--force`

Forces the command to perform the operation. This option is valid with the `leave` subcommand.

`-m` *cluster*
`--member-cluster` *cluster*

Specifies the name of one or more clusters to configure as a site member. This option is valid with the `add-member`, `create`, and `set-role` subcommands.

Use a comma-separated list to specify multiple clusters.

`-p` *name=value*
`--property` *name=value*

Sets site property values. This option is valid with the `create` and `set-prop` subcommands.

Multiple instances of the `-p` option are allowed. See Site Properties for the site properties that you can specify with this option.

`-V`
`--version`

Displays the version of the command.

Do not specify this option with subcommands, operands, or other options, as they are ignored. The `-V` option only displays the version of the command. No other operations are performed.

```
-v
--verbose
```

Displays verbose output of the command. This option is valid with the `show` subcommand.

The following operands are supported:

cluster

Specifies the name of a cluster.

site

Specifies the name of a site.

The following section lists the site properties. These properties are set at creation and tunable at runtime.

Site Properties

`Description`

Describes the site. The system sets this property on the local cluster, then propagates the value to the other clusters in the site.

Category:

Optional

Default:

None

Tunable:

Assigned at creation and tunable at runtime

`joinTimeout`

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.

Category:

Optional

Type:

Integer

Minimum:

    20 seconds

Maximum:

    3600 seconds

Default:

    30 seconds

Tunable:

    Assigned at creation and tunable at runtime

`Query_interval`

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.

Category:

    Optional

Type:

    Integer

Minimum:

    20 seconds

Maximum:

    300 seconds

Default:

    120 seconds

Tunable:

    Assigned at creation and tunable at runtime

`Timeout`

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.

Category:

    Optional

Type:

Integer

Minimum:

30 seconds

Maximum:

3600 seconds

Default:

30 seconds

Tunable:

Assigned at creation and tunable at runtime

The following exit values are returned:

0   The command completed successfully.

nonzero   An error has occurred.

EXAMPLE 35   Adding Site Members

The following `geosite` command adds the `cluster-chelsea` cluster as a member of the `site-london` site. The command is issued from a node of the `cluster-soho` cluster, which is a site controller of the `site-london` site.

```
cluster-soho# geosite add-member -m cluster-chelsea site-london
```

EXAMPLE 36   Creating and Joining a Site

The following `geosite` command creates the `site-london` site. The `cluster-soho` cluster is added as a controller and the `cluster-kew` cluster is added as a member. The invited clusters accept the invitation from the `cluster-chelsea` cluster to join the `site-london` site.

```
phys-cluster-chelsea1# geosite create -c cluster-soho -m cluster-kew site-london

phys-cluster-soho1# geosite join cluster-chelsea site-london
phys-cluster-kew1# geosite join cluster-chelsea site-london
```

EXAMPLE 37   Deleting a Site

The following `geosite` commands remove the site clusters `cluster-chelsea` and `cluster-kew` from the `site-london` site, and then removes the final cluster, `cluster-soho`, from the site

using the `-f` option. The site is automatically deleted when it contains no more clusters. The commands are run from a node of the site controller `cluster-soho`.

```
phys-soho1# geosite remove-member cluster-chelsea,cluster-kew site-london
phys-soho1# geosite leave -f site-london
```

EXAMPLE 38   Joining a Site

The following `geosite` command accepts the invitation from controller `cluster-soho` for `cluster-kew` to join the `site-london` site.

```
phys-kew1# geosite join cluster-soho site-london
```

EXAMPLE 39   Synchronizing Site Configuration Information

The following `geosite` command resynchronizes the local cluster, `cluster-soho` with site configuration information from the remote cluster `cluster-kew`.

```
phys-soho1# geosite update cluster-kew site-london
```

EXAMPLE 40   Leaving a Site

The following `geosite` command removes the cluster `cluster-kew` from the `site-london` site.

```
phys-kew1# geosite leave site-london
```

EXAMPLE 41   Setting the Site Role of a Cluster

The following `geosite` command sets the role of `cluster-soho` to be a member of the `site-london` site and `cluster-kew` to be a controller of the site. The command is issued from a node of the site controller `cluster-soho`.

```
phys-soho1# geosite set-role -m cluster-soho -c cluster-kew  site-london
```

See `attributes`(5) for descriptions of the following attributes.

ATTRIBUTE TYPE	ATTRIBUTE VALUE
Availability	ha-cluster/geo/geo-framework
Interface Stability	Evolving

`geoadm`(1M) on page 21, `geohb`(1M) on page 33, `geomg`(1M) on page 39, `geopg`(1M) on page 55, `geops`(1M) on page 79, `rbac`(5)

Index