Oracle® Solaris Cluster Data Replication Guide for Oracle Data Guard

Part No: E71432
May 2019

ORACLE

Oracle Solaris Cluster Data Replication Guide for Oracle Data Guard

Part No: E71432

Access to Oracle Support

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

Référence: E71432

Accès aux services de support Oracle

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

Contents

Tables

Examples

Using This Documentation

- **Overview** – Describes how to administer Oracle Data Guard data replication with the Oracle Solaris Cluster disaster recovery framework.
- **Audience** – Experienced system administrators with extensive knowledge of Oracle software and hardware.
- **Required knowledge** – Knowledge of the Oracle Solaris operating system, of Oracle Solaris Cluster software, and expertise with the volume manager software that is used with Oracle Solaris Cluster software.

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

Product Documentation Library

Documentation and resources for this product and related products are available at http://www.oracle.com/pls/topic/lookup?ctx=E69294_01.

Feedback

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

♦ ♦ ♦ CHAPTER 1

Replicating Data With Oracle Data Guard Software

This chapter describes how to configure data replication with Oracle Data Guard software in an Oracle Solaris Cluster disaster recovery framework environment (formerly called Geographic Edition).

This chapter covers the following topics:

The disaster recovery framework supports the use of Oracle Data Guard for data replication when used with HA for Oracle Database, HA for Oracle External Proxy, or Oracle RAC software.

The disaster recovery framework also supports the use of Oracle Data Guard instances running on a system that is not under Oracle Solaris Cluster control, called a *remote node*. Replication can be performed between two databases that are both running on remote nodes, or performed between a database on a remote node and another database on a cluster node that is configured with the the disaster recovery framework.

Note - To manage an Oracle Data Guard database on a remote node, you must configure the HA for Oracle External Proxy resource for the remote database instance on the same cluster where you create the protection group.

Before you can replicate data with Oracle Data Guard, you must be familiar with the Oracle Data Guard documentation. For information about installing and configuring the Oracle Data Guard software and its latest patches, see the Oracle Data Guard documentation. If you plan to configure an Oracle Data Guard database on a remote node, also see *Oracle Solaris Cluster Data Service for Oracle External Proxy Guide*.

During data replication, data from a primary cluster is copied to a backup, or standby cluster. The standby cluster can be located at a site that is geographically separated from the primary

cluster. The distance between the primary and standby clusters depends on the distance that your data replication product supports.

Caution - When using HA for Oracle Database for an Oracle Data Guard database instance, take special care to coordinate the switchover with HA for Oracle Database. HA for Oracle Database monitors the database and will take action when the database's state changes unexpectedly. Likewise, when the disaster recovery framework is used to remotely manage an Oracle Data Guard database instance, the remote node must not have HA for Oracle Database configured for this Oracle Data Guard instance.

When both HA for Oracle Database and the disaster recovery framework are used for the same Oracle Data Guard database instance on the same Oracle Solaris Cluster node, if you must switch over the Oracle Data Guard state directly, first unmonitor the HA for Oracle Database resource. After the switchover is done, ensure that the `Dataguard_role` and `Standby_mode` resource properties exactly match the current database settings locally, before you re-enable monitoring of the HA for Oracle Database resource.

Failure to take these precautions might result in HA for Oracle Database issuing database restarts and failing to stay online.

Note - If you are using Oracle Data Guard and HA for Oracle Database, set `Active_data_guard` to `TRUE` to ensure that upon a start initiated by the `SUNW.oracle_server` agent, the database is started with read only mode when the cluster is standby.

The example procedures in this chapter show how to configure Oracle Data Guard to replicate data between a primary and a standby database.

Replicating Data in an Oracle Data Guard Protection Group (Task Maps)

The following tables summarizes the tasks for configuring and administering Oracle Data Guard data replication in a protection group. These procedures apply whether you configure Oracle Data Guard replication on a cluster node under Oracle Solaris Cluster control or on a remote node that is not under direct Oracle Solaris Cluster control.

TABLE 1 Configuration Tasks for Oracle Data Guard Data Replication

Task	Description
Perform an initial configuration of the Oracle Data Guard software.	See "Initially Configuring Oracle Data Guard Replication" on page 18.

Task	Description
Create a protection group that is configured for Oracle Data Guard data replication.	See "How to Create and Configure an Oracle Data Guard Protection Group" on page 39.
Add a configuration that is controlled by Oracle Data Guard.	See "How to Add an Oracle Data Guard Broker Configuration to an Oracle Data Guard Protection Group" on page 42.
Add an application resource group to the protection group.	See "How to Add an Application Resource Group to an Oracle Data Guard Protection Group" on page 46.
Replicate the protection group configuration to a standby cluster.	See "How to Replicate the Oracle Data Guard Protection Group Configuration to a Partner Cluster" on page 68.
Activate the protection group.	See "How to Activate a Protection Group" in *Administering the Disaster Recovery Framework for Oracle Solaris Cluster 4.4*.
Check the runtime status of replication.	See "Checking the Runtime Status of Oracle Data Guard Data Replication" on page 71.

TABLE 2 Administration Tasks for Oracle Data Guard Data Replication

Task	Description
Detect failure.	See "Detecting Cluster Failure" in *Administering the Disaster Recovery Framework for Oracle Solaris Cluster 4.4*.
Migrate services by using a switchover.	See "Migrating Replication Services by Switching Over Protection Groups" in *Administering the Disaster Recovery Framework for Oracle Solaris Cluster 4.4*.
Migrate services by using a takeover.	See "Forcing a Takeover of a Protection Group" in *Administering the Disaster Recovery Framework for Oracle Solaris Cluster 4.4*.
Recover data after forcing a takeover.	See "Recovering Oracle Data Guard Data After a Takeover" on page 77.

Overview of Oracle Data Guard Data Replication

This section provides an overview of the integration of Oracle Data Guard with the disaster recovery framework and highlights the differences between support for Oracle Data Guard and other supported data replication products.

Oracle Data Guard Shadow Resource Groups

When an Oracle Data Guard broker configuration that is controlling the Oracle Data Guard software is added to a protection group, the disaster recovery framework creates a special shadow Oracle database-server resource group and resource. The shadow Oracle database-server resource group "shadows" the real Oracle database-server resource group that you created to manage and monitor the Oracle databases that are under the control of Oracle Solaris Cluster software.

The name of a shadow resource group conforms to the following format:

ODG-configuration-name`-rac-proxy-svr-shadow-rg`

This format applies regardless of which data service is configured for the Oracle Database software – Support for Oracle RAC, HA for Oracle External Proxy, or HA for Oracle Database.

For example, an Oracle RAC database in an Oracle Data Guard broker configuration named `sales` has a shadow Oracle database-server resource group named `sales-rac-proxy-svr-shadow-rg`. If, however, the configuration name contains one or more periods (`.`), the periods are converted to underscore characters (_) to construct the resource group name. Consequently, the configuration name `mysales.com` has a shadow resource group named `mysales_com-rac-proxy-svr-shadow-rg`.

Similarly, an HA for Oracle database in an Oracle Data Guard broker configuration named `inventory` that is controlled by the Oracle Data Guard software has a shadow Oracle database-server resource group named `inventory-rac-proxy-svr-shadow-rg`.

The requirements for consistent name construction are two-fold. First, this allows a shadow resource group to be added to a protection group even when one cluster uses Oracle RAC and the other uses HA for Oracle Database. The second reason is that is that this format is required for backward compatibility.

Each shadow resource group contains a single resource: a `SUNW.gds` resource whose probe script reflects the status of the Oracle database-server resource group. The name of this resource conforms to the following format:

- **HA for Oracle Database or failover HA for Oracle External Proxy** – *ODG-configuration-name*`-oracle-svr-shadow-rs`
- **Oracle RAC or scalable HA for Oracle External Proxy** – *ODG-configuration-name*`-rac-proxy-svr-shadow-rs`

For more information about Oracle database-server resource groups, see *Oracle Solaris Cluster Data Service for Oracle Real Application Clusters Guide*, *Oracle Solaris Cluster Data Service*

for Oracle External Proxy Guide, and *Oracle Solaris Cluster Data Service for Oracle Database Guide*.

A shadow Oracle database-server resource group is required because, unlike other disaster recovery framework replication products, the Oracle Data Guard software is an integral part of the Oracle Database software. Oracle Data Guard requires Oracle Database to be running and the databases started to replicate its data.

Consequently, putting the real Oracle database-server resource group under disaster recovery framework control would result in the Oracle database being shut down on the standby cluster. In contrast, the shadow Oracle database-server resource group can be placed under the control of the disaster recovery framework. You can do so without disrupting the data replication process while still allowing the configuration to conform to the usual disaster recovery framework structure for managing application resource groups. In addition, putting the shadow resource groups under disaster recovery framework control enables you to declare on the shadow resource-group affinities and other relationships with other resource groups. These other resource groups can then also be controlled by the the disaster recovery framework.

The state of the shadow Oracle database-server resource group indicates whether the database that is monitored and controlled by the Oracle database-server resource group is the primary or the standby cluster. In other words, this state indicates whether the database is online on the primary cluster and unmanaged on the standby cluster:

- If the shadow Oracle database-server resource group is online, its cluster is the primary cluster for that Oracle database, which is represented by the resource in the actual Oracle database-server resource group.
- If the shadow Oracle database-server resource group is offline and unmanaged, its cluster is the standby for that Oracle database.

Furthermore, the status of the shadow Oracle database-server resource reflects both the status of the Oracle database-server resource and whether the database is the primary or the standby.

Oracle Data Guard Replication Resource Groups

When an Oracle Data Guard broker configuration that is controlling the Oracle Data Guard software is added to a protection group, the disaster recovery framework creates a special replication resource for the specific Oracle Data Guard broker configuration in the replication resource group. By monitoring these replication resource groups, the disaster recovery framework is able to monitor the overall status of replication. One replication resource group with one replication resource for each Oracle Data Guard broker configuration is created for each protection group.

The name of the replication resource group conforms to the following format:

ODG-protection-group`-odg-rep-rg`

The replication resource in the replication resource group monitors the replication status of the Oracle Data Guard broker configuration on the local cluster, which is reported by the Oracle Data Guard broker software. The replication resource also checks the accessibility of the configuration through the remote service name, to ensure that the username and password information, or the Oracle wallet if used, is correctly configured.

The name of the replication resource conforms to the following format:

ODG-broker-configuration-name`-odg-rep-rs`

Note - In Oracle Data Guard, a data replication resource is enabled when the protection group is activated in the cluster. Consequently, in Oracle Data Guard, in a cluster in which the protection group is deactivated, the data replication status appears as `unknown`.

Initially Configuring Oracle Data Guard Replication

This section describes the initial steps that you need to perform to configure Oracle Data Guard replication in the disaster recovery framework. This includes the configuration of Oracle Data Guard replication on a system that is not under Oracle Solaris Cluster control, referred to as a *remote node*.

This section provides the following information:

Oracle Data Guard Broker Configurations

To define Oracle Data Guard broker configurations, you need to determine the following information:

- **The name of the Oracle Data Guard broker configuration**, such as `mysales.com`, being replicated between the `cluster-paris` and `cluster-newyork` clusters.
- **The unique database names that are taking part in the replication**, such as `sales` on `the cluster-paris` cluster, and `salesdr` on `the cluster-newyork` cluster.
- **The Oracle service names for these databases**, such as `sales-svc` on the `cluster-paris` cluster and `salesdr-svc` on the `cluster-newyork` cluster. These names are held in one of the following locations:
  - The `tnsnames.ora` files in the `${ORACLE_HOME}/network/admin` directory of the nodes that host the replicated Oracle database
  - The Oracle naming service directory
- **The database standby type for the Oracle Data Guard broker configuration**, which you set to either `logical`, `physical`, or `snapshot`.
- **The replication mode for the Oracle Data Guard broker configuration**, which you set to `MaxPerformance`, `MaxAvailability`, or `MaxProtection`.
- **The names of the Oracle database-server resource groups that manage the Oracle databases on each cluster**. You configure these names by using the data service configuration wizard through the `clsetup` command. Alternatively, follow the instructions in "Registering and Configuring HA for Oracle Database" in *Oracle Solaris Cluster Data Service for Oracle Database Guide* or Appendix D, "Command-Line Alternatives," in *Oracle Solaris Cluster Data Service for Oracle Real Application Clusters Guide*.
- If running the primary or secondary replicated Oracle database on a remote node, **the name of the HA for Oracle External Proxy resource and resource group on that remote node**. For configuration instructions, see "Registering and Configuring HA for Oracle External Proxy" in *Oracle Solaris Cluster Data Service for Oracle External Proxy Guide*.

The disaster recovery framework manages start, stop, and role changes between primary and standby databases during switchover and takeover operations. Your Oracle Data Guard configuration may include Far Sync instances. Usage of Far Sync instances is transparent to the disaster recovery framework since the framework manages the primary and standby states and role changes between primary and standby and corresponding commands do not change in the existence of Far Sync instances.

For more information about the Oracle Data Guard broker configuration, refer to the Oracle Data Guard broker documentation.

How to Set Up Your Primary Database

Perform this task from cluster nodes where you want Oracle Data Guard to run. These can be nodes of an Oracle Solaris Cluster configuration or a system that is not under Oracle Solaris Cluster control.

In the following steps, the primary cluster is called `cluster-paris` (nodes `phys-paris-1` and `phys-paris-2`), and the standby cluster is called `cluster-newyork` (`phys-newyork-1` and `phys-newyork-2`). The suffix `-crs` is appended to the Oracle Clusterware virtual IP host names.

Note - Oracle Clusterware is a component of Oracle Grid Infrastructure.

The primary database on `cluster-paris` is called `sales` and has instances `sales1` and `sales2`. The standby database on `cluster-newyork` is called `salesdr` and has instances `salesdr1` and `salesdr2`. The suffix `-svc` is appended to each net naming service name for each of the databases and individual instances, for example, `sales-svc` or `sales1 -svc`.

Note - Although the following example procedure uses an Oracle 10*g* RAC database, the principles are the same for an Oracle 11*g* RAC database. This release of disaster recovery framework software supports a minimum of Oracle version 11.4.

Before You Begin Ensure that you have edited your Oracle user `.profile` or `.cshrc` file to set the correct Oracle `SID`, `ORACLE_HOME`, and `PATH` environment variables for the local Oracle RAC database instance. Unless otherwise stated, you only need to run the commands from a node in the primary cluster that hosts a protected database instance.

1. **Verify that you can resolve the Oracle virtual IP addresses that are used by Oracle Clusterware on *all* primary and standby nodes.**

```
phys-paris-1# getent hosts phys-paris-1-crs
10.11.112.41    phys-paris-1-crs
…
```

2. **Create a database on the primary cluster.**

   Use either the Oracle Database Configuration Assistant (`dbca`) or the SQL*Plus utility.

3. **Verify that an Oracle password file exists for the primary database.**

```
oracle (phys-paris-1)$ cd ${ORACLE_HOME}/dbs
oracle (phys-paris-1)$ ls -l orapwsales1
lrwxrwxrwx   1 oracle   oinstall      25 November  2 02:06 orapwsales1
-> /oradata/SALES/orapwsales
```

Oracle Data Guard needs a consistent Oracle password file on all participating nodes in the primary and standby clusters.

If a password file does not exist, create one as follows:

```
oracle (phys-paris-1)$ orapwd file=${ORACLE_HOME}/dbs/orapwsales1 \
password=sysdba_password
```

You can then move this file to a location on shared storage and create a symbolic link to that file from each node. Change the file name to reflect the local `SID` on each node. Later, you will copy this file to the standby cluster (`cluster-newyork`).

4. Ensure that the database is in `logging` mode by using the `sqlplus` command.

```
oracle (phys-paris-1)$ sqlplus '/ as sysdba'
SQL> alter database force logging;
Database altered.
```

5. Configure the Oracle Data Guard Broker configuration file locations.

Run the `sqlplus` command as follows, substituting the two file names with ones that suit your configuration. Ensure that these files are located on shared storage that is visible to all `cluster-paris` nodes.

```
oracle (phys-paris-1)$ sqlplus '/ as sysdba'
SQL> alter system set dg_broker_config_file1='/oradata/SALES/dr1sales.dat'
2 scope=both sid='*';
System altered.
SQL> alter system set dg_broker_config_file2='/oradata/SALES/dr2sales.dat'
2 scope=both sid='*';
System altered.
```

6. Shut down all database instances.

7. On the primary database, mount a single database instance and enable the Oracle database flashback capability.

```
oracle (phys-paris-1)$ sqlplus '/ as sysdba'
SQL> startup mount;
ORACLE instance started.

Total System Global Area  532676608 bytes
Fixed Size                  2031416 bytes
Variable Size             276824264 bytes
Database Buffers          247463936 bytes
Redo Buffers                6356992 bytes
Database mounted.
```

```
System altered.
SQL> alter database archivelog;
Database altered.
SQL> alter database flashback on;
Database altered.
SQL> alter database open;
Database altered.
```

8. **Restart the other database instances.**

9. **Create database standby redo logs.**

   Depending on your configuration, you might need to add a number of standby redo logs. The name, number, and size of these logs depend on a number of factors, including whether you use the Optimal Flexible Architecture (OFA), how many online redo log files you have, and the size of those log files.

   The following example shows how to configure a single 50-Mbyte standby redo log file, where the OFA naming scheme is being used. A default, two-node Oracle database database normally requires that you add six log files.

   ```
   oracle (phys-paris-1)$ sqlplus '/ as sysdba'
   SQL> alter database add standby logfile size 50m;
   Database altered.
   ```

10. **Configure the Oracle log archiving destinations.**

    Depending on your configuration, you might need to alter or add one or more of the Oracle log archive destination parameters. These parameters have a number of tunable properties. Consult the Oracle documentation for details.

    The following example shows two log archive destinations being set, one for the local cluster and one for the standby cluster, where OFA naming is used.

    ```
    oracle (phys-paris-1)$ sqlplus '/ as sysdba'
    SQL> alter system set log_archive_dest_1='location=use_db_recovery_file_dest
    2 arch mandatory valid_for=(all_logfiles,all_roles)
    3 db_unique_name=sales' scope=both sid='*';
    System altered.

    SQL> alter system set log_archive_dest_2='service=salesdr-svc
    2 lgwr sync affirm valid_for=(online_logfiles,primary_role)
    3 db_unique_name=salesdr' scope=both sid='*';
    System altered.

    SQL> alter system set log_archive_dest_10='location=use_db_recovery_file_dest'
    2 scope=both sid='*';
    System altered.
    ```

```
SQL> alter system set standby_file_management='AUTO' scope=both sid='*';
System altered.
```

11. Configure the Fetch Archive Log (FAL) parameters.

For the database to know where to get missing archive redo logs on the server and where to send them on the client, you need to set the FAL system properties. These properties use the net service names of the source and destination databases. You run the following `sqlplus` command to set the parameters to the correct values for your configuration.

```
oracle (phys-paris-1)$ sqlplus '/ as sysdba'
SQL> alter system set fal_server='salesdr-svc' scope=both sid='*';
System altered.

SQL> alter system set fal_client='sales-svc' scope=both sid='*';
System altered.
```

▼ How to Configure the Primary Database Listener and Naming Service

Perform this task on each primary cluster node where you want Oracle Data Guard to run. These can be nodes of an Oracle Solaris Cluster configuration or a system that is not under Oracle Solaris Cluster control.

1. Create a static listener for Oracle Data Guard.

Note - Perform this step on all `cluster-paris` nodes.

Oracle Data Guard requires that you configure a static listener. The following example uses `${ORACLE_HOME}=/oracle/oracle/product/10.2.0/db_1` and shows where to add the entry for the static listener in the `${ORACLE_HOME}/network/admin/listener.ora` file. The `SID_LIST_LISTENER_PHYS-PARIS-1` and `(SID_NAME = sales1)` lines vary from node to node, while the `(GLOBAL_DBNAME=sales_DGMGRL)` differs on `cluster-newyork`. Later, you will add these entries on the `cluster-newyork` nodes.

```
oracle (phys-paris-1)$ cat ${ORACLE_HOME}/network/admin/listener.ora
SID_LIST_LISTENER_PHYS-PARIS-1 =
(SID_LIST =
(SID_DESC =
(SID_NAME = PLSExtProc)
(ORACLE_HOME = /oracle/oracle/product/10.2.0/db_1)
(PROGRAM = extproc)
)
```

```
(SID_DESC =
(SID_NAME = sales1)
(GLOBAL_DBNAME=sales_DGMGRL)
(ORACLE_HOME = /oracle/oracle/product/10.2.0/db_1)
)
)
oracle (phys-paris-1)$
```

2. Restart the listener.

To enable the static entries, restart the Oracle listener processes on each of the nodes on `cluster-paris`.

```
oracle (phys-paris-1)$ lsnrctl stop LISTENER_PHYS_PHYS-PARIS-1
LSNRCTL for Solaris: Version 10.2.0.4.0 - Production on 29-OCT-2008 02:04:56

Copyright (c) 1991, 2006, Oracle.  All rights reserved.

Connecting to (ADDRESS=(PROTOCOL=tcp)(HOST=)(PORT=1521))
The command completed successfully
oracle$ lsnrctl start LISTENER_PHYS_PHYS-PARIS-1
LSNRCTL for Solaris: Version 10.2.0.4.0 - Production on 29-OCT-2008 02:05:04
…Services Summary...
Service "PLSExtProc" has 1 instance(s).
Instance "PLSExtProc", status UNKNOWN, has 1 handler(s) for this service...
Service "sales_DGMGRL" has 1 instance(s).
Instance "sales1", status UNKNOWN, has 1 handler(s) for this service...
The command completed successfully
```

Wait while databases register with listener

```
oracle (phys-paris-1)$ lsnrctl status LISTENER_PHYS_PHYS-PARIS-1
LSNRCTL for Solaris: Version 10.2.0.4.0 - Production on 29-OCT-2008 02:04:56

Copyright (c) 1991, 2006, Oracle.  All rights reserved.
…
Services Summary...
Service "PLSExtProc" has 1 instance(s).
Instance "PLSExtProc", status UNKNOWN, has 1 handler(s) for this service...
Service "sales" has 2 instance(s).
Instance "sales1", status READY, has 2 handler(s) for this service...
Instance "sales2", status READY, has 1 handler(s) for this service...
Service "salesXDB" has 2 instance(s).
Instance "sales1", status READY, has 1 handler(s) for this service...
Instance "sales2", status READY, has 1 handler(s) for this service...
Service "sales_DGB" has 2 instance(s).
Instance "sales1", status READY, has 2 handler(s) for this service...
Instance "sales2", status READY, has 1 handler(s) for this service...
Service "sales_DGMGRL" has 1 instance(s).
Instance "sales1", status UNKNOWN, has 1 handler(s) for this service...
```

```
Service "sales_XPT" has 2 instance(s).
Instance "sales1", status READY, has 2 handler(s) for this service...
Instance "sales2", status READY, has 1 handler(s) for this service...
The command completed successfully
```

3. Verify the network service naming entries for all database instances.

Ensure that the naming service method that you are using, either `tnsnames.ora` or the directory service, has entries defined for all the Oracle database instances in both clusters.

The following example shows the type of entries that you include for the `cluster-paris` cluster only. Entries for the `cluster-newyork` cluster are added in "How to Configure the Standby Database Listener and Naming Service" on page 29. Also, add entries for the standby (`salesdr`) database instances that you create later when you modify the `pfile` parameter file. In the example, the `sales` database dynamically registers a service name of `sales` with the listeners (see the database `service_names` initialization parameter).

```
oracle (phys-paris-1)$ cat ${ORACLE_HOME}/network/admin/tnsnames.ora
SALES1-SVC =
(DESCRIPTION =
(ADDRESS_LIST =
(ADDRESS = (PROTOCOL = TCP)(HOST = phys-paris-1-crs)(PORT = 1521)
(SEND_BUF_SIZE = 65535)(RECV_BUF_SIZE = 65535))
)
(CONNECT_DATA =
(SERVER = DEDICATED)
(SERVICE_NAME = sales)
(INSTANCE_NAME = sales1)
)
)

SALES2-SVC =
(DESCRIPTION =
(ADDRESS_LIST =
(ADDRESS = (PROTOCOL = TCP)(HOST = phys-paris-2-crs)(PORT = 1521)
(SEND_BUF_SIZE = 65535)(RECV_BUF_SIZE = 65535))
)
(CONNECT_DATA =
(SERVER = DEDICATED)
(SERVICE_NAME = sales)
(INSTANCE_NAME = sales2)
)
)

SALES-SVC =
(DESCRIPTION =
(ADDRESS_LIST =
(ADDRESS = (PROTOCOL = TCP)(HOST = phys-paris-1-crs)(PORT = 1521)
```

```
(SEND_BUF_SIZE = 65535)(RECV_BUF_SIZE = 65535))
(ADDRESS = (PROTOCOL = TCP)(HOST = phys-paris-2-crs)(PORT = 1521)
(SEND_BUF_SIZE = 65535)(RECV_BUF_SIZE = 65535))
(LOAD_BALANCE = yes)
)
(CONNECT_DATA =
(SERVER = DEDICATED)
(SERVICE_NAME = sales)
)
)

LISTENERS_SALES =
(ADDRESS_LIST =
(ADDRESS = (PROTOCOL = TCP)(HOST = phys-paris-1-crs)(PORT = 1521))
(ADDRESS = (PROTOCOL = TCP)(HOST = phys-paris-2-crs)(PORT = 1521))
)
```

How to Prepare Your Standby Database

Perform this task from cluster nodes where you want Oracle Data Guard to run. These can be nodes of an Oracle Solaris Cluster configuration or a remote server that is not under Oracle Solaris Cluster control.

1. **Create a backup of the primary database.**

   The following example shows how to use the Oracle Recovery Manager (RMAN) utility to create a copy of the primary database that you can restore on the standby `cluster-newyork` cluster. The example also shows how to avoid performing a separate step to create a control file for the standby database. For more information about the options for completing this step, see your Oracle Database documentation.

   ```
   oracle (phys-paris-1)$ rman
   RMAN> connect target sys/DBA_password@sales-svc;
   RMAN> connect auxiliary /;
   RMAN> backup device type disk tag 'mybkup' database include current
   2> controlfile for standby;
   RMAN> backup device type disk tag 'mybkup' archivelog all not backed up;
   ```

2. **Copy the backup files to the standby system.**

   Create the appropriate directory hierarchies on the `cluster-newyork` cluster and copy the database backup to this cluster. The actual locations that you specify for the files that are shown in the example depend on the specific choices that you made when you configured the database.

   ```
   oracle (phys-newyork-1)$ mkdir -p $ORACLE_BASE/admin/salesdr
   oracle (phys-newyork-1)$ cd $ORACLE_BASE/admin/salesdr
   ```

```
oracle (phys-newyork-1)$ mkdir adump bdump cdump dpdump hdump pfile udump
```

Make the directory for the database backup

```
oracle (phys-newyork-1)$ mkdir -p /oradata/flash_recovery_area/SALES/backupset/date
```

Copy over the files

```
oracle (phys-newyork-1)$ cd /oradata/flash_recovery_area/SALES/backupset/date
oracle (phys-newyork-1)$ scp oracle@phys-paris-1:`pwd`/\* .
```

Make the base directory for new database files

```
oracle (phys-newyork-1)$ mkdir -p /oradata/SALESDR
```

3. Create a `pfile` parameter file.

Create a suitable server initialization file for the standby (`salesdr`) database. The easiest way to create this file is to copy the parameters for the primary database and modify them. The following example shows how to create a `pfile` parameter file:

```
oracle (phys-paris-1)$ sqlplus '/ as sysdba'
SQL> CREATE PFILE='/tmp/initpfile_for_salesdr.ora' FROM SPFILE;
File created.
SQL> quit
```

4. Modify the `pfile` parameter file.

Change all entries that are particular to the primary cluster to entries that are suitable for the standby cluster, as shown in the following example. Modify entries that are prefixed by an Oracle `SID`, that is, `sales1` or `sales2`, to use standby database instance `SID` names, that is, `salesdr1` and `salesdr2`. Depending on your configuration, you might need to make additional changes.

Note - Do not change the `db_name` parameter, as it must remain `sales` on both clusters.

You created these directories previously

```
*.audit_file_dest='/oracle/oracle/product/10.2.0/db_1/admin/salesdr/adump'
*.background_dump_dest='/oracle/oracle/product/10.2.0/db_1/admin/salesdr/bdump'
*.user_dump_dest='/oracle/oracle/product/10.2.0/db_1/admin/salesdr/udump'
*.core_dump_dest='/oracle/oracle/product/10.2.0/db_1/admin/salesdr/cdump'
```

Remove the following entry

```
*.control_files='...list primary control files...'
```

Add this entry

```
*.db_unique_name='salesdr'

*.dg_broker_config_file1='/oradata/SALESDR/dr1salesdr.dat'
*.dg_broker_config_file2='/oradata/SALESDR/dr2salesdr.dat'

*.dispatchers='(PROTOCOL=TCP) (SERVICE=salesdrXDB)'
```

Switch the client and server entries around, as shown in the following entries

```
*.fal_client='salesdr-svc'
*.fal_server='sales-svc'

*.remote_listener='LISTENERS_SALESDR'
```

Switch the log archive destinations

```
*.log_archive_dest_1='location=use_db_recovery_file_dest arch
mandatory valid_for=(all_logfiles,all_roles) db_unique_name=salesdr'
*.log_archive_dest_2='service=sales-svc lgwr sync affirm
valid_for=(online_logfiles,primary_role) db_unique_name=sales'
```

5. Copy the `pfile` parameter file to the standby system.

6. Start the standby database and convert the `pfile` parameter file to an `spfile` server parameter file.

a. As the `oracle` user, log in to one of the `cluster-newyork` nodes and convert the `pfile` parameter file to an `spfile` server parameter file.

```
oracle (phys-newyork-1)$ ORACLE_SID=salesdr1 export ORACLE_SID
oracle (phys-newyork-1)$ sqlplus '/ as sysdba'
SQL> startup nomount pfile='/tmp/initpfile_for_salesdr.ora';
SQL> create spfile='/oradata/SALESDR/spfilesalesdr.ora'
2> from pfile='/tmp/initpfile_for_salesdr.ora';
SQL> shutdown
```

b. Create an `${ORACLE_HOME}/dbs/initsalesdr1.ora` file on all `cluster-newyork` nodes and, in that file, insert the following entry:

```
oracle (phys-newyork-1) cat ${ORACLE_HOME}/dbs/initsalesdr1.ora
SPFILE='/oradata/SALESDR/spfilesalesdr.ora'
```

c. Restart the database, on one node only, to prepare for restoring the backed-up primary database.

```
oracle (phys-newyork-1) sqlplus '/ as sysdba'
    You are now starting from the spfile
SQL> startup nomount
ORACLE instance started.

Total System Global Area  532676608 bytes
Fixed Size                  2031416 bytes
Variable Size             289407176 bytes
Database Buffers          234881024 bytes
Redo Buffers                6356992 bytes
```

7. **Copy the Oracle password file for the primary database for use by the standby database.**

   a. **Copy the Oracle password file that you created on the `cluster-paris` cluster.**

      Place the file on shared storage on the `cluster-newyork` cluster.

   b. **Create links to this file from each of the `cluster-newyork` nodes.**

      Again change the name of the symbolic link to reflect the Oracle `SID` on the local standby node.

How to Configure the Standby Database Listener and Naming Service

Perform this task on each node of the standby cluster where you want Oracle Data Guard to run. These can be nodes of an Oracle Solaris Cluster configuration or a system that is not under Oracle Solaris Cluster control.

1. **Create a static listener for Oracle Data Guard.**

Note - Perform this step on all `cluster-newyork` nodes.

Oracle Data Guard requires that you configure a static listener.

The following example uses `${ORACLE_HOME}=/oracle/oracle/product/10.2.0/db_1` and shows where to add the entry for the static listener in the `${ORACLE_HOME}/network/admin/listener.ora` file. The `SID_LIST_LISTENER_PHYS-NEWYORK-1` and `(SID_NAME = salesdr1)` lines vary from node to node, while the `(GLOBAL_DBNAME=salesdr_DGMGRL)` differs on `cluster-paris`.

```
oracle (phys-newyork-1)$ cat ${ORACLE_HOME}/network/admin/listener.ora
SID_LIST_LISTENER_PHYS-NEWYORK-1 =
(SID_LIST =
(SID_DESC =
(SID_NAME = PLSExtProc)
(ORACLE_HOME = /oracle/oracle/product/10.2.0/db_1)
(PROGRAM = extproc)
)
(SID_DESC =
(SID_NAME = salesdr1)
(GLOBAL_DBNAME=salesdr_DGMGRL)
(ORACLE_HOME = /oracle/oracle/product/10.2.0/db_1)
```

```
)
)
oracle (phys-newyork-1)$
```

2. Restart the listener.

To enable the static entries, restart the Oracle listener processes on each of the nodes on `cluster-newyork`.

```
oracle (phys-newyork-1)$ lsnrctl stop LISTENER_PHYS_PHYS-NEWYORK-1
LSNRCTL for Solaris: Version 10.2.0.4.0 - Production on 29-OCT-2008 02:04:56

Copyright (c) 1991, 2006, Oracle.  All rights reserved.

Connecting to (ADDRESS=(PROTOCOL=tcp)(HOST=)(PORT=1521))
The command completed successfully
oracle$ lsnrctl start LISTENER_PHYS_PHYS-NEWYORK-1
LSNRCTL for Solaris: Version 10.2.0.4.0 - Production on 29-OCT-2008 02:05:04

Copyright (c) 1991, 2006, Oracle.  All rights reserved.

Starting /oracle/oracle/product/10.2.0/db_1/bin/tnslsnr: please wait...

TNSLSNR for Solaris: Version 10.2.0.4.0 - Production

Services Summary...
Service "PLSExtProc" has 1 instance(s).
Instance "PLSExtProc", status UNKNOWN, has 1 handler(s) for this service...
Service "salesdr_DGMGRL" has 1 instance(s).
Instance "salesdr1", status UNKNOWN, has 1 handler(s) for this service...
The command completed successfully
```

Wait while databases register with listener

```
oracle (phys-newyork-1)$ lsnrctl status LISTENER_PHYS_PHYS-NEWYORK-1
LSNRCTL for Solaris: Version 10.2.0.4.0 - Production on 29-OCT-2008 02:04:56

Copyright (c) 1991, 2006, Oracle.  All rights reserved.…
Services Summary...
Service "PLSExtProc" has 1 instance(s).
Instance "PLSExtProc", status UNKNOWN, has 1 handler(s) for this service...
Service "salesdr" has 2 instance(s).
Instance "salesdr1", status READY, has 2 handler(s) for this service...
Instance "salesdr2", status READY, has 1 handler(s) for this service...
Service "salesdrXDB" has 2 instance(s).
Instance "salesdr1", status READY, has 1 handler(s) for this service...
Instance "salesdr2", status READY, has 1 handler(s) for this service...
Service "salesdr_DGB" has 2 instance(s).
```

```
Instance "salesdr1", status READY, has 2 handler(s) for this service...
Instance "salesdr2", status READY, has 1 handler(s) for this service...
Service "salesdr_DGMGRL" has 1 instance(s).
Instance "salesdr1", status UNKNOWN, has 1 handler(s) for this service...
Service "salesdr_XPT" has 2 instance(s).
Instance "salesdr1", status READY, has 2 handler(s) for this service...
Instance "salesdr2", status READY, has 1 handler(s) for this service...
The command completed successfully
```

3. Verify the net service naming entries for all database instances.

Ensure that the naming service method that you are using, either `tnsnames.ora` or the directory service, has entries defined for all the Oracle database instances in both clusters.

The following example shows the type of entries that you include for the `cluster-newyork` cluster only. Entries for the `cluster-paris` cluster are added in "How to Configure the Primary Database Listener and Naming Service" on page 23. In the example, the `salesdr` database dynamically registers a service name of `salesdr` with the listeners (see the database `service_names` initialization parameter).

```
oracle (phys-newyork-1)$ cat ${ORACLE_HOME}/network/admin/tnsnames.ora
SALESDR1-SVC =
(DESCRIPTION =
(ADDRESS_LIST =
(ADDRESS = (PROTOCOL = TCP)(HOST = phys-newyork-1-crs)(PORT = 1521)
(SEND_BUF_SIZE = 65535)(RECV_BUF_SIZE = 65535))
)
(CONNECT_DATA =
(SERVER = DEDICATED)
(SERVICE_NAME = salesdr)
(INSTANCE_NAME = salesdr1)
)
)

SALESDR2-SVC =
(DESCRIPTION =
(ADDRESS_LIST =
(ADDRESS = (PROTOCOL = TCP)(HOST = phys-newyork-2-crs)(PORT = 1521)
(SEND_BUF_SIZE = 65535)(RECV_BUF_SIZE = 65535))
)
(CONNECT_DATA =
(SERVER = DEDICATED)
(SERVICE_NAME = salesdr)
(INSTANCE_NAME = salesdr2)
)
)

SALESDR-SVC =
```

```
(DESCRIPTION =
(ADDRESS_LIST =
(ADDRESS = (PROTOCOL = TCP)(HOST = phys-newyork-1-crs)(PORT = 1521)
(SEND_BUF_SIZE = 65535)(RECV_BUF_SIZE = 65535))
(ADDRESS = (PROTOCOL = TCP)(HOST = phys-newyork-2-crs)(PORT = 1521)
(SEND_BUF_SIZE = 65535)(RECV_BUF_SIZE = 65535))
(LOAD_BALANCE = yes)
)
(CONNECT_DATA =
(SERVER = DEDICATED)
(SERVICE_NAME = salesdr)
)
)

LISTENERS_SALESDR =
(ADDRESS_LIST =
(ADDRESS = (PROTOCOL = TCP)(HOST = phys-newyork-1-crs)(PORT = 1521))
(ADDRESS = (PROTOCOL = TCP)(HOST = phys-newyork-2-crs)(PORT = 1521))
)
```

4. **Verify that the standby listener `listener.ora` and `tnsnames.ora` files have the correct entries, and restart the listener process.**

   Ensure that these files include the static Oracle Data Guard listener entry and the naming service entries for the primary and standby cluster database service. If you are not using the Oracle directory naming service lookup, you need to include the entries in `tnsnames.ora`.

```
oracle (phys-newyork-1)$ lsnrctl stop LISTENER_PHYS-NEWYORK-1
LSNRCTL for Solaris: Version 10.2.0.4.0 - Production on 29-OCT-2008 02:04:56

Copyright (c) 1991, 2006, Oracle.  All rights reserved.

Connecting to (ADDRESS=(PROTOCOL=tcp)(HOST=)(PORT=1521))
The command completed successfully
oracle$ lsnrctl start LISTENER_PHYS-NEWYORK-1
LSNRCTL for Solaris: Version 10.2.0.4.0 - Production on 29-OCT-2008 02:05:04

Copyright (c) 1991, 2006, Oracle.  All rights reserved.

Starting /oracle/oracle/product/10.2.0/db_1/bin/tnslsnr: please wait...

TNSLSNR for Solaris: Version 10.2.0.4.0 - Production
…
Services Summary...
Service "PLSExtProc" has 1 instance(s).
Instance "PLSExtProc", status UNKNOWN, has 1 handler(s) for this service...
Service "salesdr_DGMGRL" has 1 instance(s).
Instance "salesdr1", status UNKNOWN, has 1 handler(s) for this service...
```

```
The command completed successfully
```

How to Start and Recover Your Standby Database

1. **Restore the database backup.**

   Continuing to work on the `cluster-newyork` cluster, you can now restore the data from the backup of the primary database to the standby database.

   The following example shows how to use the Oracle Recovery Manager (RMAN) utility.

```
oracle (phys-newyork-1) rman
RMAN> connect target sys/oracle@sales-svc;
RMAN> connect auxiliary /;
RMAN> duplicate target database for standby nofilenamecheck;
…
```

2. **Add standby redo logs to the standby database.**

   The exact requirements that you must meet depend on your configuration. The steps you follow are identical to those that you followed for the primary cluster.

3. **Enable flashback on the standby database.**

```
oracle (phys-newyork-1)$ sqlplus '/ as sysdba'
SQL> alter database flashback on;
Datbase altered.
SQL> shutdown immediate;
SQL> startup mount;
ORACLE instance started.
…
```

4. **Recover the standby database.**

```
oracle (phys-newyork-1) sqlplus '/ as sysdba'
SQL> alter database recover managed standby database using current logfile disconnect;
```

How to Verify That Your Configuration Is Working Correctly

1. **Verify that the log file transmission is working.**

   When the `SQL>` prompt is displayed, log in to one of the database instances on the `cluster-paris` cluster and perform a couple log switches.

```
oracle (phys-paris-1)$ sqlplus '/ as sysdba'
SQL> alter system switch logfile;
SQL> alter system switch logfile;
```

2. **Check the `${ORACLE_HOME}/admin/sales/bdump/alert_sales1.log` for any problems that might have prevented the logs from being archived.**

   If there are errors, correct them. This process might take time. You can check that the network connectivity is correct by using the following command:

```
oracle (phys-paris-1)$ tnsping salesdr-svc
oracle (phys-newyork-1)$ tnsping sales-svc
```

How to Complete Configuring and Integrating Your Standby Oracle Database

1. **If you are using Oracle Clusterware, register the new database.**

   Note - Oracle Clusterware is a component of Oracle Grid Infrastructure.

   Place the standby database under Oracle Clusterware control and configure it to open when Oracle Clusterware starts.

   - **For a single-instance database, run the following command:**

```
oracle (phys-newyork-1)$ srvctl add database -d salesdr \
-r PHYSICAL_STANDBY -o $ORACLE_HOME -s open;
```

   - **For Oracle RAC, run the following commands:**

```
oracle (phys-newyork-1)$ srvctl add database -d salesdr \
-r PHYSICAL_STANDBY -o $ORACLE_HOME -s open;
oracle (phys-newyork-1)$ srvctl add instance -d salesdr \
-i salesdr1 -n $phys-newyork-1;
oracle (phys-newyork-1)$ srvctl add instance -d salesdr \
-i salesdr2 -n $phys-newyork-2;
```

2. **Enable Oracle Data Guard on both the primary and standby databases.**

   Perform the following commands on only one node in each cluster (`cluster-paris` and `cluster-newyork`).

```
oracle (phys-newyork-1)$ sqlplus '/ as sysdba'
```

```
SQL> alter system set dg_broker_start=true scope=both sid='*';
SQL> quit

oracle (phys-paris-1)$ sqlplus '/ as sysdba'
SQL> alter system set dg_broker_start=true scope=both sid='*';
SQL> quit
```

▼ How to Create and Enable an Oracle Data Guard Broker Configuration

To use Oracle Data Guard with the disaster recovery framework, you need to create an Oracle Data Guard broker configuration.

In the following example procedure, the Oracle Data Guard broker configuration is called `mysales.com`. The `salesdr` database is a physical copy of the `sales` database.

1. Create an Oracle Data Guard broker configuration for the primary database.

You use the `dgmgrl` command to create the Oracle Data Guard broker configuration. You need to know the name of the Oracle Data Guard broker configuration that you want to create, the name of the primary database, and the net service name through which to connect. You will need to know these properties again, when you specify the configuration to the disaster recovery framework.

```
oracle (phys-paris-1)$ dgmgrl sys/sysdba_password@sales-svc
DGMGRL> create configuration mysales.com as primary
DGMGRL> database is sales connect identifier is sales-svc;
```

If you find errors when you connect to the Oracle Data Guard broker, check the `${ORACLE_HOME}/admin/sales/bdump/alert_`*prim_sid*`.log` file. You can check that the configuration has been created by using the following command:

```
oracle (phys-paris-1)$ dgmgrl sys/sysdba_password@sales-svc
DGMGRL> show configuration;
Configuration
Name:                mysales.com
Enabled:             NO
Protection Mode:     MaxPerformance
Fast-Start Failover: DISABLED
Databases:
sales   - Primary database

Current status for "mysales.com":
DISABLED
```

2. **Add the standby database to the Oracle Data Guard broker configuration.**

   You need to know the name of the standby database, the net service name through which to connect, and the type of standby (`physical` or `logical`).

   ```
   oracle (phys-paris-1)$ dgmgrl sys/sysdba_password@sales-svc
   DGMGRL> add database salesdr as connect identifier is
   salesdr-svc maintained as physical;
   ```

3. **Configure the apply instance for the standby database.**

   If the standby database is also a multi-instance Oracle RAC database, you can specify the instance on which you would prefer the transmitted archive redo logs to be applied. Before you enable the configuration, issue the following command:

   ```
   oracle$ dgmgrl sys/sysdba_password@sales-svc
   DGMGRL> edit database salesdr set property PreferredApplyInstance='salesdr1';
   ```

4. **To verify that the Oracle Data Guard broker configuration is working correctly, enable the configuration.**

   ```
   oracle (phys-paris-1)$ dgmgrl sys/sysdba_password@sales-svc
   DGMGRL> enable configuration;
   ```

   If you have successfully performed all steps, you can check the status of the configuration by using the following command:

   ```
   oracle$ dgmgrl sys/sysdba_password@sales-svc
   DGMGRL> show configuration;
   Configuration
   Name:                mysales.com
   Enabled:             YES
   Protection Mode:     MaxPerformance
   Fast-Start Failover: DISABLED
   Databases:
   sales   - Primary database
   salesdr - Physical standby database

   Current status for "mysales.com":
   SUCCESS
   ```

5. **Verify that the Oracle Data Guard broker configuration can switch over.**

   Before you add the Oracle Data Guard broker configuration to the disaster recovery framework, you need to verify that you can perform a switchover of the database from the primary to the standby and back again. If this switchover does not work, the disaster recovery framework will not be able to perform this operation either.

   ```
   oracle (phys-paris-1)$ dgmgrl sys/sysdba_password@sales-svc
   ```

```
DGMGRL> switchover to salesdr
Performing switchover NOW, please wait...
Operation requires shutdown of instance "sales1" on database "sales"
Shutting down instance "sales1"...
ORA-01109: database not open

Database dismounted.
ORACLE instance shut down.
Operation requires shutdown of instance "salesdr1" on database "salesdr"
Shutting down instance "salesdr1"...
ORA-01109: database not open

Database dismounted.
ORACLE instance shut down.
Operation requires startup of instance "sales1" on database "sales"
Starting instance "sales1"...
ORACLE instance started.
Database mounted.
Operation requires startup of instance "salesdr1" on database "salesdr"
Starting instance "salesdr1"...
ORACLE instance started.
Database mounted.
Switchover succeeded, new primary is "salesdr"

DGMGRL> switchover to sales;
Performing switchover NOW, please wait...
Operation requires shutdown of instance "salesdr1" on database "salesdr"
Shutting down instance "salesdr1"...
ORA-01109: database not open

Database dismounted.
ORACLE instance shut down.
Operation requires shutdown of instance "sales1" on database "sales"
Shutting down instance "sales1"...
ORA-01109: database not open

Database dismounted.
ORACLE instance shut down.
Operation requires startup of instance "salesdr1" on database "salesdr"
Starting instance "salesdr1"...
ORACLE instance started.
Database mounted.
Operation requires startup of instance "sales1" on database "sales"
Starting instance "sales1"...
ORACLE instance started.
Database mounted.
Switchover succeeded, new primary is "sales"
```

Next Steps Go to "How to Configure Oracle Solaris Cluster Manageability Resources for the Primary and Standby Databases" on page 38.

▼ How to Configure Oracle Solaris Cluster Manageability Resources for the Primary and Standby Databases

This procedure integrates the primary and standby Oracle Data Guard databases with Oracle Solaris Cluster software. By configuring these resource groups and resources, you enable Oracle Solaris Cluster software to monitor and manage the primary and standby databases. These resource groups are then used by the disaster recovery framework to monitor and manage the Oracle Data Guard configuration.

Perform this procedure on nodes of the Oracle Solaris Cluster configuration.

- **Configure the Oracle Solaris Cluster manageability resources.**

  The Oracle Solaris Cluster resource groups that you create are then entered as properties when the Oracle Data Guard configuration is added to the protection group. The disaster recovery framework uses these resource groups to monitor and manage the Oracle Data Guard configuration during protection group operations such as switchover and takeover.

  You can use either the data service configuration wizard that is available through the `clsetup` utility or use Oracle Solaris Cluster maintenance commands. Follow procedures for the data service you use:

  - **HA for Oracle Database** – "How to Register and Configure HA for Oracle Database With Oracle Grid Infrastructure for a Cluster (CLI)" in *Oracle Solaris Cluster Data Service for Oracle Database Guide*
  - **Oracle RAC** – "How to Enable Oracle Solaris Cluster and Oracle Grid Infrastructure to Interoperate" in *Oracle Solaris Cluster Data Service for Oracle Real Application Clusters Guide* or "Creating Resources for Interoperation With Oracle Grid Infrastructure by Using Oracle Solaris Cluster Maintenance Commands" in *Oracle Solaris Cluster Data Service for Oracle Real Application Clusters Guide*.
  - **HA for Oracle External Proxy** – "Registering and Configuring HA for Oracle External Proxy" in *Oracle Solaris Cluster Data Service for Oracle External Proxy Guide*

♦ ♦ ♦ CHAPTER 2

Creating Oracle Data Guard Protection Groups

This chapter describes how to create an Oracle Data Guard protection group.

Creating and Validating an Oracle Data Guard Protection Group

This section covers the following topics:

▼ How to Create and Configure an Oracle Data Guard Protection Group

The following procedure builds on the example configuration that was described in Chapter 1, "Replicating Data With Oracle Data Guard Software".

Note - You can also accomplish this procedure by using the Oracle Solaris Cluster Manager browser interface. Click Partnerships, click the partnership name, and in the Protection Groups section click `Create`. For more information about Oracle Solaris Cluster Manager, see Chapter 12, "Using the Oracle Solaris Cluster Manager Browser Interface" in *Administering an Oracle Solaris Cluster 4.4 Configuration*.

In this example, the `sales` database is online on the `cluster-paris` cluster and is protected by Oracle Data Guard.

Before You Begin Ensure that the following conditions are met:

- The Oracle Data Guard broker configuration exists. The disaster recovery framework does *not* create the configuration for you.
- Your clusters are members of a partnership.
- The protection group that you are creating does not already exist.

Note - Protection group names are unique in the global disaster recovery framework namespace. You cannot use the same protection group name in two partnerships on the same system.

You can also replicate the existing configuration of a protection group from a remote cluster to the local cluster. For more information, see "Replicating the Oracle Data Guard Protection Group Configuration to a Partner Cluster" on page 68.

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

   For more information, see "Securing Disaster Recovery Framework Software" in *Installing and Configuring the Disaster Recovery Framework for Oracle Solaris Cluster 4.4*.

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

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

   The `/var/cluster/geo` directory must have the correct access control lists (ACL) applied for compatibility between the Geo Management rights profile and Oracle Data Guard.

   ---

2. **On all nodes of the local cluster, create a new protection group.**

   ```
   phys-node-n# geopg create -s partnership -d odg \
   -o local-role [-p property [-p...]] protection-group
   ```

   `-s` *partnership*

   Specifies the name of the partnership.

   `-d odg`

   Specifies that the protection group data is replicated by Oracle Data Guard software.

`-o` *local-role*

Specifies the role of this protection group on the local cluster as either primary or secondary.

`-p` *property-setting*

Specifies the properties of the protection group.

You can specify the following properties:

- `Description` – Describes the protection group.
- `Timeout` – Specifies the timeout period for the protection group, in seconds.

protection-group

Specifies the name of the protection group.

For information about the names and values that are supported by the disaster recovery framework, see Appendix B, "Legal Names and Values of Disaster Recovery Framework Entities," in *Administering the Disaster Recovery Framework for Oracle Solaris Cluster 4.4*.

For more information about the `geopg` command, refer to the `geopg(8)` man page.

Before creating the protection group, the data replication layer validates that the configuration is correct.

- If the validation is successful, the local `Configuration` status is set to `OK` and the `Synchronization` status is set to `Error`.
- If the validation is unsuccessful, the protection group is not created.

Adding an Oracle Data Guard Broker Configuration to an Oracle Data Guard Protection Group

A protection group is the container for the replication component and the application resource groups, which contain data for services that are protected from disaster. the disaster recovery framework protects the data by replicating it from the primary cluster to the standby cluster. By adding an Oracle Data Guard broker configuration to a protection group, the disaster recovery framework monitors the status of the data replication that corresponds to the database in the Oracle Data Guard broker configuration.

The disaster recovery framework also controls the role and state of the Oracle Data Guard broker configuration during protection group operations, such as start, stop, switchover, and takeover.

▼ How to Add an Oracle Data Guard Broker Configuration to an Oracle Data Guard Protection Group

Before You Begin

Before you add an Oracle Data Guard broker configuration to a protection group, ensure that the following conditions are met:

- The protection group is defined on the local cluster.
- If the partner cluster can be reached, the protection group is offline on the local cluster and the partner cluster.
- The Oracle Data Guard broker configuration exists on both the primary and standby database systems.
- The Oracle database-server resource group and Oracle database-server resources that manage the Oracle database that is replicated by Oracle Data Guard exist on both the local and the partner cluster.
- If the cluster will remotely manage an Oracle database, each cluster node is installed and configured with HA for Oracle External Proxy.
- If you are using HA for Oracle External Proxy to manage replication of a remote Oracle database, all cluster nodes where HA for Oracle External Proxy is installed are also installed with the Oracle Database software client administrative kit. This software installs `$ORACLE_HOME/bin/sqlplus` and `$ORACLE_HOME/bin/dgmgrl`.
- If you are using HA for Oracle External Proxy, ensure that one of the following is true on all cluster nodes:
  - The `TNS_ADMIN` property of the HA for Oracle External Proxy resource is set to `$ORACLE_HOME/network/admin`.
  - The `/var/cluster/geo/odg/`*odg_configuration_name*`_config` file contains the entry `ORACLE_HOME=$ORACLE_HOME`.
- If you are using HA for Oracle External Proxy, ensure that you have granted the `sysdba` privilege to the user that the agent utilizes to monitor the database. The following SQL command grants this privilege:

```
$ sqlplus '/ as sysdba'
SQL> grant sysdba to hauser;
Grant succeeded.
```

The *hauser* is the name you set in the `dbuser` property of the corresponding `SUNW.oracle_external_proxy` resource. For more information, see "Remote Database User" in *Oracle Solaris Cluster Data Service for Oracle External Proxy Guide*.

1. **Ensure that the Oracle Data Guard broker properties `BystandersFollowRoleChange` and `FAST_START FAILOVER` are set properly.**

   a. **Set `BystandersFollowRoleChange` to `NONE`.**

   ```
   DGMGRL> edit configuration set property BystandersFollowRoleChange=NONE;
   ```

   b. **Ensure that `FAST_START FAILOVER` is disable.**

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

   For more information, see "Securing Disaster Recovery Framework Software" in *Installing and Configuring the Disaster Recovery Framework for Oracle Solaris Cluster 4.4*.

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

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

   The `/var/cluster/geo` directory must have the correct access control lists (ACL) applied for compatibility between the Geo Management rights profile and Oracle Data Guard.

   ---

3. **For HA for Oracle Database, ensure that the `Standby_mode` and `Dataguard_role` extension properties of the `SUNW.oracle_server` resource match the current standby mode of the Oracle Data Guard broker database.**

   Perform this step on one node of **each** partner cluster that runs HA for Oracle Database.

   ```
   phys-newyork-n# clresource set -p Standby_mode=mode \
   -p Dataguard_role=role ora-db-rs
   phys-paris-n# clresource set -p Standby_mode=mode \
   -p Dataguard_role=role ora-db-rs
   ```

4. **Add an Oracle Data Guard broker configuration to the protection group.**

   Note - You can also accomplish this step by using the Oracle Solaris Cluster Manager browser interface. Click Partnerships, click the partnership name, highlight the protection group name, and in the Data Replication Components section click `Add`. For more information about Oracle Solaris Cluster Manager, see Chapter 12, "Using the Oracle Solaris Cluster Manager Browser Interface" in *Administering an Oracle Solaris Cluster 4.4 Configuration*.

This command adds a configuration to a protection group on the local cluster and propagates the new configuration to the partner cluster if the partner cluster contains a protection group of the same name.

```
phys-node-n# geopg add-replication-component \
-p property [-p…] ODG-configuration-name protection-group
```

`-p` *property*

Specifies the properties of either the Oracle Data Guard broker configuration, the Oracle database-server resource group, or the Oracle database user name and the associated password.

You can specify the following properties:

- `local_database_name` – Name of the local database in the Oracle Data Guard broker configuration.
- `local_db_service_name` – Oracle net service name for the local database.
- `local_oracle_svr_rg_name` – Name of the local Oracle database-server resource group that manages the local database in the Oracle Data Guard broker configuration. This resource group can be configured for Oracle RAC, HA for Oracle Database, or HA for Oracle External Proxy.
- `remote_database_name` – Name of the remote database in the Oracle Data Guard broker configuration.
- `remote_db_service_name` – Oracle net service name for the remote database.
- `remote_oracle_svr_rg_name` – Name of the Oracle database-server resource group on the partner cluster that manages the remote database in the Oracle Data Guard broker configuration. This resource group can be configured for Oracle RAC, HA for Oracle Database, or HA for Oracle External Proxy.
- `standby_type` – Standby type for the database in the Oracle Data Guard broker configuration.
- `sysdba_password` – Password for the Oracle `SYSDBA` privileged database user.
  Do not specify the actual 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.

  If you use an Oracle wallet, you do not need to specify this password.
- `sysdba_username` – Name of an Oracle `SYSDBA` privileged database user who can perform the Oracle Data Guard broker switchover and takeover operations.

  If you use an Oracle wallet, you do not need to specify this password.

For more information about the properties that you can set, see Appendix A, "Standard Disaster Recovery Framework Properties," in *Administering the Disaster Recovery Framework for Oracle Solaris Cluster 4.4*.

ODG-configuration-name

Specifies the name of the new Oracle Data Guard broker configuration.

protection-group

Specifies the name of the protection group that contains the new Oracle Data Guard broker configuration.

For information about the names and values that are supported by the disaster recovery framework, see Appendix B, "Legal Names and Values of Disaster Recovery Framework Entities," in *Administering the Disaster Recovery Framework for Oracle Solaris Cluster 4.4*.

For more information about the `geopg` command, refer to the `geopg`(8) man page.

5. **Ensure that the External_Dependency_Allowed property of the protection group is set to `True` if both of the following conditions exist.**

   - The resource group that is defined in the `local_rac_proxy_svr_rn_name` or `remote_rac_proxy_svr_rg_name` property has an affinity with, or a dependency on, the *rac-framework-rg*.
   - The corresponding cluster has other resource groups that have an affinity with, or a dependency on, the *rac-framework-rg*.

   ```
   # geopg show protection-group | grep -i external_dependency_allowed
   ```

   If necessary, change the property value to `True`.

   ```
   # geopg set-prop -p External_Dependency_Allowed=True protection-group
   ```

Example 1 Adding an Oracle Data Guard broker Configuration to an Oracle Data Guard Protection Group

This example shows how to add an Oracle Data Guard broker configuration to the `sales-pg` protection group.

To run the following command successfully, you must already be able to connect to both a local and a remote database service.

```
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 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
```

Adding an Application Resource Group to an Oracle Data Guard Protection Group

To make an application highly available, you must ensure that the application is managed as a resource in an application resource group. Unlike other data replication modules, the Oracle database-server resource group is not added to the protection group. Instead, a shadow Oracle database-server resource group is added to represent this resource group.

You can add and remove the shadow Oracle database-server resource group to and from the protection group at any time without affecting the Oracle Data Guard data replication. This fact does not prevent you from adding other, non-Oracle database-server resource groups to the protection group if necessary. However, these applications cannot use any data that requires replication to the standby cluster as only Oracle Data Guard is supported in this type of protection group.

You need to replicate, on the standby cluster, all entities that you configure for the primary cluster's application resource group. Examples of entities that you need to replicate are application data resources, configuration files, and resource groups. Resource group names must also match on both clusters. In addition, the data that the application resource uses needs to be replicated on the standby cluster.

▼ How to Add an Application Resource Group to an Oracle Data Guard Protection Group

Note - You can also accomplish this procedure by using the Oracle Solaris Cluster Manager browser interface. Click Partnerships, click the partnership name, click the protection group name, and in the Resource Groups section click `Add Resource Groups`. For more information about Oracle Solaris Cluster Manager, see Chapter 12, “Using the Oracle Solaris Cluster Manager Browser Interface” in *Administering an Oracle Solaris Cluster 4.4 Configuration*.

Before You Begin You can add an existing resource group, other than an Oracle database-server resource group containing an Oracle database-server resource, to the list of application resource groups for a protection group. If you do try to add an Oracle database-server resource group, the `geopg` command returns an error.

Before you add an application resource group (of any other type) to a protection group, ensure that the following conditions are met:

- The protection group is defined.
- The application resource group does not need any data replicating. You are not prevented from adding such resource groups, but the Oracle Data Guard module does not coordinate the switchover of other types of data replication.
- The resource group to add already exists on both clusters and is in an appropriate state.

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

   For more information, see "Securing Disaster Recovery Framework Software" in *Installing and Configuring the Disaster Recovery Framework for Oracle Solaris Cluster 4.4*.

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

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

   The `/var/cluster/geo` directory must have the correct access control lists (ACL) applied for compatibility between the Geo Management rights profile and Oracle Data Guard.

2. **Ensure that the `Auto_start_on_new_cluster` property of the resource group is set to `False`.**

   ```
   # clresourcegroup show -p Auto_start_on_new_cluster resource-group
   ```

   If necessary, change the property value to `False`.

   ```
   # clresourcegroup set -p Auto_start_on_new_cluster=False resource-group
   ```

3. **If the application resource group must have dependencies on resource groups and resources that are not managed by this protection group, ensure that the `External_Dependency_Allowed` property of the protection group is set to `True`.**

   Issue the following command to view the current property setting:

   ```
   # geopg list protection-group | grep -i external_dependency_allowed
   ```

   If necessary, change the property value to `True`.

   ```
   # geopg set-prop -p External_Dependency_Allowed=TRUE protection-group
   ```

4. **Start the protection group or change the state of the application resource group to a state that is required for the addition to be allowed.**

   The following are disaster recovery framework requirements for application resource groups:

   - On the secondary cluster, the application resource group be in the `Unmanaged` state.
   - If the protection group is stopped on the primary cluster, the application resource group must be `Unmanaged` on the primary cluster.
   - If the protection group is active on the primary cluster, the application resource group must be in the `Unmanaged` or `Online` state on the primary cluster.

5. **Add an application resource group to the protection group.**

   `phys-`*`node-n`*`#` **`geopg add-resource-group`** *application-resource-group protection-group*

   application-resource-group

   Specifies the name of the application resource group. You can specify more than one resource group in a comma-separated list.

   protection-group

   Specifies the name of the protection group.

   This command adds an application resource group to a protection group on the local cluster. If the partner cluster contains a protection group of the same name, the command then propagates the new configuration information to the partner cluster.

   For information about the names and values that are supported by the disaster recovery framework, see Appendix B, "Legal Names and Values of Disaster Recovery Framework Entities," in *Administering the Disaster Recovery Framework for Oracle Solaris Cluster 4.4*.

   After the application resource group is added to the protection group, the application resource group is managed as an entity of the protection group. The application resource group is affected by protection group operations such as start, stop, switchover, and takeover.

Example 2 Adding an Application Resource Group to an Oracle Data Guard Protection Group

This example shows how to add two application resource groups, `apprg1` and `apprg2`, to `sales-pg`.

```
phys-paris-1# geopg add-resource-group apprg1,apprg2 sales-pg
```

Validating an Oracle Data Guard Protection Group

This section provides the following information:

How the Data Replication Layer Validates the Application Resource Groups

During protection group validation, the Oracle Data Guard data replication layer validates the application resource groups in the protection group. The Oracle Data Guard data replication layer verifies the following conditions:

- **The resource group under the control of the protection group that is being validated does not contain a resource group that contains an Oracle database-server resource**.
  - If you add a failover resource group, it must not contain a `SUNW.oracle_server` resource.
  - If you add a scalable resource group, it must not contain a `SUNW.scalable_rac_server_proxy` resource.
  - Neither a failover nor a scalable resource group can contain an `ORCL.oracle_external_proxy` resource.

  You cannot add these resource groups to an Oracle Data Guard protection group because the Oracle database that is managed by the Oracle database-server resource is shut down on the standby cluster when the protection group is started globally, thus disabling the Oracle Data Guard data replication.
- **The `Auto_start_on_new_cluster` property in an application resource group in the protection group is set to `False`.**

  When you bring a protection group online on the primary cluster, the data replication layer brings the application resources groups that are participating in that protection group online only on the same primary cluster. Setting the `Auto_start_on_new_cluster` property to `False` prevents the Oracle Solaris Cluster resource group manager from automatically starting the application resource groups. In this case, the startup of resource groups is reserved for the disaster recovery framework.

  When the protection group is activated, application resource groups in the protection group need to be online only on the primary cluster.

The `Configuration` status is set to `OK` after successful validation. If validation is not successful, the `Configuration` status is set to `Error`.

How the Data Replication Subsystem Verifies the Oracle Data Guard Broker Configuration

When you add an Oracle Data Guard broker configuration to a protection group, the data replication layer verifies that the Oracle Data Guard broker configuration exists.

When you run the `geopg add-replication-component` command, a shadow Oracle database-server resource group and a replication resource group for the Oracle Data Guard broker configuration are created. In addition, the configuration is successfully validated on the local cluster. However, the configuration might not be valid on the remote cluster. You can use the `geopg validate` *protection-group* command on the remote cluster to troubleshoot an invalid configuration.

Note - To avoid possible configuration errors, do not create these resource groups separately, before running the `geopg add-replication-component` command.

The shadow Oracle database-server resource group contains an Oracle Solaris Cluster resource. This resource is based on the Generic Data Service `SUNW.gds` resource type. The shadow Oracle database-server resource shadows the real Oracle database-server resource that manages and monitors the Oracle database in the Oracle Data Guard broker configuration.

For more information about the shadow Oracle database-server resource group, see “Oracle Data Guard Shadow Resource Groups” on page 16.

The replication resource group contains an Oracle Solaris Cluster resource that is based on the `SUNW.gds` resource type. The replication resource monitors the state of the database replication as reported by Oracle Data Guard broker.

For more information about replication resources, see “Oracle Data Guard Replication Resource Groups” on page 17.

For the validation to be successful, ensure that the following conditions are met:

- The resource group that is named in the `local_oracle_svr_rg_name` property contains a resource of the appropriate resource type:
  - For a scalable resource group, the property specifies a resource group that contains a resource of the `SUNW.scalable_rac_server_proxy` or `SUNW.oracle_external_proxy` resource type.
  - For a failover resource group, the property specifies a resource group that contains a resource of the `SUNW.oracle_server` or `SUNW.oracle_external_proxy` resource type.

When the resource group contains a resource of the `SUNW.scalable_rac_server_proxy` or `SUNW.oracle_server` resource type, that resource is used to determine the values for `${ORACLE_HOME}` and the local Oracle database `SID` values.

- The Oracle Data Guard `dgmgrl` command shows a `SUCCESS` status for the Oracle Data Guard broker configuration. The presence of Oracle `ORA-` messages in the output from the `dgmgrl` command might indicate that the `sysdba_username` password is incorrect or that the cluster has been disabled. Oracle errors are returned as part of the messages that are generated by the validate command.
- The `sysdba_username` password is valid for the standby cluster to ensure that switchovers are possible. Or, the Oracle wallet connection mechanism, `dgmgrl /@`*service_name*, can successfully connect to the broker.
- The Oracle Data Guard broker configuration details match those held by the disaster recovery framework. The details to check include which cluster is primary, the configuration name, the database mode (for both the primary and standby cluster), the replication mode, the standby type, that `FAST_START FAILOVER` is disabled, and that `BystandersFollowRoleChange` is set to `NONE`.

Caution - Do not use Oracle Solaris Cluster commands to change, remove, or bring offline these resources or resource groups. Use only disaster recovery framework commands to administer shadow Oracle database-server resource groups, replication resource groups, and resources that are internal entities that are managed by the disaster recovery framework. Altering the configuration or state of these entities directly with Oracle Solaris Cluster commands could result in an unrecoverable failure.

▼ How to Validate an Oracle Data Guard Protection Group

Note - You can also accomplish this procedure by using the Oracle Solaris Cluster Manager browser interface. Click Partnerships, click the partnership name, highlight the protection group name, and click `Validate`. For more information about Oracle Solaris Cluster Manager, see Chapter 12, “Using the Oracle Solaris Cluster Manager Browser Interface” in *Administering an Oracle Solaris Cluster 4.4 Configuration*.

When the `Configuration` status of a protection group is displayed as `Error` in the output of the `geoadm status` command, you can validate the configuration by using the `geopg validate` command. This command checks the current state of the protection group and its entities.

If the protection group and its entities are valid, the `Configuration` status of the protection groups is set to `OK`. If the `geopg validate` command finds an error in the configuration files, the command displays a message about the error and the configuration remains in the error state. In

such a case, you can fix the error in the configuration and run the `geopg validate` command again.

This command validates the configuration of the protection group on the local cluster only. To validate the protection group configuration on the partner cluster, run the command again on the partner cluster.

Before You Begin Before validating the configuration of a protection group, ensure that the protection group you want to validate exists locally and that the common agent container is online on all nodes of both clusters in the partnership.

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

   For more information, see "Securing Disaster Recovery Framework Software" in *Installing and Configuring the Disaster Recovery Framework for Oracle Solaris Cluster 4.4*.

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

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

   The `/var/cluster/geo` directory must have the correct access control lists (ACL) applied for compatibility between the Geo Management rights profile and Oracle Data Guard.

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

   This command validates the configuration of a single protection group on the local cluster only.

   ```
   phys-node-n# geopg validate protection-group
   ```

Example 3 Validating the Configuration of a Protection Group

This example shows how to validate a protection group.

```
phys-node-n# geopg validate sales-pg
```

♦ ♦ ♦ CHAPTER 3

Configuring Oracle Data Guard Protection Groups

This chapter describes how to administer data replication with Oracle Data Guard software.

This chapter covers the following topics:

Working With Oracle Data Guard Protection Groups

Unlike other supported data replication mechanisms, Oracle Data Guard is an integral part of Oracle database software. Consequently, you do not place Oracle database-server resource groups under disaster recovery framework control as you do when you are using one of these host or storage-based data replication mechanisms.

You can add Oracle Data Guard broker configurations for databases that are being replicated by Oracle Data Guard to the disaster recovery framework without stopping the databases.

You must set the Oracle Data Guard broker property `BystandersFollowRoleChange` to `NONE` as soon as the broker configuration is created and preferably before the broker is added to the disaster recovery framework configuration.

Overview of Administering Protection Groups

To add an existing Oracle Data Guard broker configuration that contains an Oracle Data Guard replicated database to a new protection group, you will complete the following general procedures.

1. On a node in either cluster, create the protection group.

   This procedure is covered in "How to Create and Configure an Oracle Data Guard Protection Group" on page 39.
2. On the same node, add the Oracle Data Guard broker configuration to the protection group.

   This procedure is covered in "How to Add an Oracle Data Guard Broker Configuration to an Oracle Data Guard Protection Group" on page 42.
3. On a node in the *other* cluster, retrieve the protection group configuration.

   This procedure is covered in "How to Replicate the Oracle Data Guard Protection Group Configuration to a Partner Cluster" on page 68.
4. On the same node, add the shadow Oracle database-server resource group and application resource group to the protection group.

   This procedure is covered in "How to Add an Application Resource Group to an Oracle Data Guard Protection Group" on page 46.
5. Activate the protection group, either globally from either cluster or locally from the primary.

   This procedure is covered in "How to Activate a Protection Group" in *Administering the Disaster Recovery Framework for Oracle Solaris Cluster 4.4*.

▼ How to Administer an Oracle Data Guard Protection Group (Example)

Note - The following example shows all the steps that are involved in administering Oracle Data Guard protection groups, as described in more detail in procedures that are included later in this chapter.

1. Ensure that the Oracle Data Guard broker properties `BystandersFollowRoleChange` and `FAST_START FAILOVER` are set properly.

a. Set `BystandersFollowRoleChange` to `NONE`.

```
DGMGRL> edit configuration set property BystandersFollowRoleChange=NONE;
```

b. Ensure that `FAST_START FAILOVER` is disable.

2. Create the protection group on the `cluster-paris` cluster.

```
phys-paris-1# geopg create -d odg -o primary -s paris-newyork-ps sales-pg
Protection group "sales-pg" has been successfully created
```

The `cluster-paris` cluster is the primary cluster. You do not need to set any additional Oracle Data Guard protection group properties.

3. Add the Oracle Data Guard broker configuration, `mysales.com`, to the protection group.

This command creates the `mysales_com-rac-proxy-svr-shadow-rg` shadow resource group.

Caution - To ensure security, do *not* supply a password when you specify the `sysdba_password` property. 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. You can pipe the password to the command if you want to drive the `geopg` command from another shell script.

For ease of management and greater security, configure an Oracle wallet to manage public key security credentials on Oracle clients and servers. If you use an Oracle wallet, specify only `-p sysdba_username=` and the disaster recovery framework Data Guard module will use the `/@`*local_service_name* or `/@`*remote_service_name* connect string syntax. For more information, see "Using Oracle Wallet Manager" in *Oracle Database Advanced Security Administrator's Guide* (`http://docs.oracle.com/cd/E11882_01/network.112/e40393/asowalet.htm#ASOAG160`).

Also, to run the following command successfully, you must already be able to connect to both a local and a remote database service.

```
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 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"
```

4. Confirm that the shadow Oracle database and replication resource groups and resources that you added to the protection group in the preceding step were added.

```
phys-paris-1# clresourcegroup status
=== Cluster Resource Groups ===

Group Name                                Node Name        Suspended   Status
----------                                ---------        ---------   ------
rac-framework-rg                         phys-paris-1     No         Online
                                         phys-paris-2     No         Online

scal-oradata-dg-rg                       phys-paris-1     No         Online
                                         phys-paris-2     No         Online

qfs-oradata-mds-rg                       phys-paris-1     No         Online
                                         phys-paris-2     No         Offline

scal-oradata-mp-rg                       phys-paris-1     No         Online
                                         phys-paris-2     No         Online

rac_server_proxy-rg                      phys-paris-1     No         Online
                                         phys-paris-2     No         Online

geo-clusterstate                         phys-paris-1     No         Online
                                         phys-paris-2     No         Online

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

sales-pg-odg-rep-rg                  phys-paris-1     No         Online
                                         phys-paris-2     No         Offline

mysales_com-rac-proxy-svr-shadow-rg   phys-paris-1    No         Unmanaged
                                         phys-paris-2    No         Unmanaged

phys-paris-1# clresource status
Resource Name                            Node Name        State
 Status Message
-------------                            ---------        -----
 --------------
rac-framework-rs                         phys-paris-1     Online
 Online
                                         phys-paris-2     Online
 Online

rac-udlm-rs                              phys-paris-1     Online
 Online
```

```
                                        phys-paris-2    Online
 Online

rac-svm-rs                              phys-paris-1    Online
 Online
                                        phys-paris-2    Online
 Online

crs_framework-rs                        phys-paris-1    Online
 Online
                                        phys-paris-2    Online
 Online

scal-oradata-dg-rs                      phys-paris-1    Online
 Online - Diskgroup online
                                        phys-paris-2    Online
 Online - Diskgroup online

qfs-oradata-mds-rs                      phys-paris-1    Online
 Online - Service is online.
                                        phys-paris-2    Offline
 Offline

scal-oradata-mp-rs                      phys-paris-1    Online
 Online
                                        phys-paris-2    Online
 Online

rac_server_proxy-rs                     phys-paris-1    Online
 Online - Oracle instance UP
                                        phys-paris-2    Online
 Online - Oracle instance UP

geo-servicetag                          phys-paris-1    Online but not monitored
 Online
                                        phys-paris-2    Online but not monitored
 Online

geo-clustername                         phys-paris-1    Offline
 Offline
                                        phys-paris-2    Online
 Online - LogicalHostname
online.

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

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

mysales_com-odg-rep-rs              phys-paris-1     Offline                    Offline
                                       phys-paris-2     Offline
 Offline

mysales_com-rac-proxy-svr-shadow-rs phys-paris-1     Offline                    Offline
                                       phys-paris-2     Offline
 Offline
```

5. Locally activate the protection group.

```
phys-paris-1# geopg start -e local sales-pg
Processing operation... The timeout period for this operation on
each cluster is 3600 seconds (3600000 milliseconds)...
Protection group "sales-pg" successfully started.
```

If your `mysales.com` Oracle Data Guard broker configuration is not already enabled, this process might take a few minutes or more. The actual time that the process takes depends on the configuration of your primary and standby databases as well as the distance between the clusters.

6. Verify that the data replication is successfully started.

```
phys-paris-1# geoadm status
Cluster: cluster-paris

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

Heartbeat "hb_cluster-paris~cluster-newyork" monitoring \
"paris-newyork-ps" OK
Plug-in "ping-plugin"            : Inactive
Plug-in "tcp_udp_plugin"         : OK

Protection group "sales-pg"              : Error
Partnership                        : paris-newyork-ps
Synchronization                    : Error

Cluster cluster-paris              : OK
Role                             : Primary
Activation State                 : Activated
```

```
Configuration                   : OK
Data replication                : OK
Resource groups                 : None

Cluster cluster-newyork            : Unknown
Role                            : Unknown
Activation State                : Unknown
Configuration                   : Unknown
Data Replication                : Unknown
Resource Groups                 : Unknown
```

7. On one node of the partner cluster, retrieve the protection group.

```
phys-newyork-1# geopg get -s paris-newyork-ps sales-pg
Protection group "sales-pg" has been successfully created.
```

8. Confirm that the shadow Oracle RAC and replication resource groups and resources for the protection group that you retrieved in the preceding step were retrieved.

```
phys-newyork-1# clresourcegroup status

=== Cluster Resource Groups ===

Group Name                                Node Name         Suspended   Status
----------                                ---------         ---------   ------
rac-framework-rg                          phys-newyork-1    No          Online
                                          phys-newyork-2    No          Online

scal-oradata-dg-rg                        phys-newyork-1    No          Online
                                          phys-newyork-2    No          Online

qfs-oradata-mds-rg                        phys-newyork-1    No          Online
                                          phys-newyork-2    No          Offline

scal-oradata-mp-rg                        phys-newyork-1    No          Online
                                          phys-newyork-2    No          Online

rac_server_proxy-rg                       phys-newyork-1    No          Online
                                          phys-newyork-2    No          Online

geo-clusterstate                          phys-newyork-1    No          Online
                                          phys-newyork-2    No          Online

geo-infrastructure                        phys-newyork-1    No          Offline
                                          phys-newyork-2    No          Online

sales-pg-odg-rep-rg                  phys-newyork-1    No          Online
```

```
                                        phys-newyork-2    No          Offline

mysales_com-rac-proxy-svr-shadow-rg   phys-newyork-1    No          Unmanaged
                                        phys-newyork-2    No          Unmanaged

phys-newyork-1# clresource status

=== Cluster Resources ===

Resource Name                           Node Name         State         Status
 Message
-------------                           ---------         -----
 --------------
rac-framework-rs                        phys-newyork-1    Online        Online
                                        phys-newyork-2    Online        Online

rac-udlm-rs                             phys-newyork-1    Online        Online
                                        phys-newyork-2    Online        Online

rac-svm-rs                              phys-newyork-1    Online        Online
                                        phys-newyork-2    Online        Online

crs_framework-rs                        phys-newyork-1    Online        Online
                                        phys-newyork-2    Online        Online

scal-oradata-dg-rs                      phys-newyork-1    Online        Online -
 Diskgroup online
                                        phys-newyork-2    Online        Online -
 Diskgroup online

qfs-oradata-mds-rs                      phys-newyork-1    Online        Online -
 Service is online.
                                        phys-newyork-2    Offline       Offline

scal-oradata-mp-rs                      phys-newyork-1    Online        Online
                                        phys-newyork-2    Online        Online

rac_server_proxy-rs                     phys-newyork-1    Online        Online -
 Oracle instance UP
                                        phys-newyork-2    Online        Online -
 Oracle instance UP

geo-servicetag                          phys-newyork-1    Online but    Online not
 monitored
                                        phys-newyork-2    Online but    Online not
 monitored

geo-clustername                         phys-newyork-1    Offline       Offline
```

```
                                        phys-newyork-2    Online        Online -
 LogicalHostname online.

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

geo-failovercontrol                     phys-newyork-1    Offline       Offline
                                        phys-newyork-2    Online        Online -
 Service is online.

mysales_com-odg-rep-rs                phys-newyork-1    Offline       Offline
                                        phys-newyork-2    Offline       Offline

mysales_com-rac-proxy-svr-shadow-rs   phys-newyork-1    Offline       Offline
                                        phys-newyork-2    Offline       Offline
```

9. From any node in a partner cluster, add the shadow Oracle database-server resource group to the protection group.

```
# geopg add-resource-group mysales_com-rac-proxy-svr-shadow-rg sales-pg
Following resource groups were successfully added:
"mysales_com-rac-proxy-svr-shadow-rg"
```

Adding the shadow Oracle database-server resource group to the protection group is not critical to the operation of the replication. The resource contained within it simply reflects the status of the real Oracle database-server resource group and highlights whether the cluster is the Oracle Data Guard primary cluster.

10. From any node in a partner cluster, globally activate the protection group on both clusters.

```
# geopg start -e global sales-pg
Processing operation... The timeout period for this operation on
each cluster is 3600 seconds (3600000 milliseconds)...
Protection group "sales-pg" successfully started.
```

11. Verify that the protection group is successfully created and activated.

```
phys-newyork-1# geoadm status
Cluster: cluster-newyork

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

Heartbeat "hb_cluster-newyork~cluster-paris" monitoring "cluster-paris": OK
```

```
Heartbeat plug-in "ping_plugin"   : Inactive
Heartbeat plug-in "tcp_udp_plugin": OK

Protection group "sales-pg"  : OK
Partnership            : "paris-newyork-ps"
Synchronization        : OK

Cluster cluster-newyork   : OK
Role                 : Primary
PG activation state  : Activated
Configuration        : OK
Data replication     : OK
Resource groups      : OK

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

Administering Oracle Data Guard Application Resource Groups

This section shows you how to delete an application resource group from an Oracle Data Guard protection group.

▼ How to Delete an Application Resource Group From an Oracle Data Guard Protection Group

You can remove an application resource group from a protection group without altering the state or contents of the application resource group. You can remove shadow Oracle database-server resource groups at any time, without affecting the Oracle database-server resource groups or Oracle databases that they represent. You can remove these resource groups because the shadow Oracle database-server resource groups simply reflect the status of the real Oracle database-server resource groups and do not control the Oracle databases.

Note - You can also accomplish this task by using the Oracle Solaris Cluster Manager browser interface. Click Partnerships, click the partnership name, click the protection group name, and in the Data Replication Components section click `Remove`. For more information about Oracle Solaris Cluster Manager, see Chapter 12, "Using the Oracle Solaris Cluster Manager Browser Interface" in *Administering an Oracle Solaris Cluster 4.4 Configuration*.

Before You Begin Ensure that the following conditions are met:

- The protection group is defined on the local cluster.
- The resource group to remove is part of the application resource groups of the protection group.

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

   For more information, see "Securing Disaster Recovery Framework Software" in *Installing and Configuring the Disaster Recovery Framework for Oracle Solaris Cluster 4.4*.

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

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

   The `/var/cluster/geo` directory must have the correct access control lists (ACL) applied for compatibility between the Geo Management rights profile and Oracle Data Guard.

2. **Remove the application resource group from the protection group.**

   `phys-`*node-n*`#` **`geopg remove-resource-group`** *resource-group-list protection-group*

   resource-group-list

   Specifies the name of the application resource group.

   You can specify more than one resource group in a comma-separated list.

   protection-group

   Specifies the name of the protection group.

   This command removes an application resource group from a protection group on the local cluster. If the partner cluster contains a protection group of the same name, the application resource group is also removed from the protection group of the partner cluster.

If the resource group that is being removed shares dependencies with other resource groups in the protection group and the `External_Dependency_Allowed` protection group property is set to `FALSE`, you also need to remove all other resource groups that share dependencies with the resource group that is being removed.

If the remove operation fails on the local cluster, the configuration of the protection group is not modified. Otherwise, the `Configuration` is removed and its status is set to `OK` on the local cluster.

If the `Configuration` status is set to `OK` on the local cluster, but the remove operation is unsuccessful on the partner cluster, the `Configuration` is removed from the partner cluster and the configuration status is set to `Error` on the partner cluster.

Example 4 Deleting an Application Resource Group From a Protection Group

This example shows how to remove two application resource groups, `apprg1` and `apprg2`, from `sales-pg`.

```
phys-paris-1# geopg remove-resource-group apprg1,apprg2 sales-pg
```

Administering Oracle Data Guard Broker Configurations

The following procedures describe how to administer Oracle Data Guard broker data replication configurations in an Oracle Data Guard protection group.

- “Oracle Data Guard Broker Configuration Properties” on page 64
- “How to Modify Disaster Recovery Framework Properties of an Oracle Data Guard Broker Configuration” on page 65
- “How to Delete an Oracle Data Guard Broker Configuration From an Oracle Data Guard Protection Group” on page 66

For details about configuring an Oracle Data Guard protection group, see “How to Create and Configure an Oracle Data Guard Protection Group” on page 39.

Oracle Data Guard Broker Configuration Properties

The `Protection Mode` property is the only Oracle Data Guard broker configuration property that you can change by using the disaster recovery framework maintenance commands.

You cannot use the disaster recovery framework to modify other Oracle Data Guard broker properties in the configuration, such as the `DelayMins`, `MaxFailure`, `MaxConnections`, and `NetTimeout` properties. You must adjust these properties manually by using the Oracle Data Guard broker command, or by modifying the appropriate database parameters that are held in the `spfile` server parameter file or the `init${SID}.ora` file through SQL*Plus.

For example, if you change the `standby_type` property with the disaster recovery framework, this change does not convert the configuration between the physical and snapshot standby states. Therefore, you must always set the value of the `standby_type` property to a value that matches the standby state that the database currently has configured. Otherwise, the configuration will experience probe and validate errors.

How to Modify Disaster Recovery Framework Properties of an Oracle Data Guard Broker Configuration

Note - You can also accomplish this procedure by using the Oracle Solaris Cluster Manager browser interface. Click Partnerships, click the partnership name, click the protection group name, click the data replication component name, and click `Edit`. For more information about Oracle Solaris Cluster Manager, see Chapter 12, "Using the Oracle Solaris Cluster Manager Browser Interface" in *Administering an Oracle Solaris Cluster 4.4 Configuration*.

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

   For more information, see "Securing Disaster Recovery Framework Software" in *Installing and Configuring the Disaster Recovery Framework for Oracle Solaris Cluster 4.4*.

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

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

   The `/var/cluster/geo` directory must have the correct access control lists (ACL) applied for compatibility between the Geo Management rights profile and Oracle Data Guard.

2. **Modify the Oracle Data Guard broker configuration.**

This command modifies the properties of an Oracle Data Guard broker configuration in a protection group on the local cluster. The command then propagates the new configuration to the partner cluster if the partner cluster contains a protection group of the same name.

```
phys-node-n# geopg modify-replication-component -p property [-p…] \
ODG-configuration-name  protection-group
```

`-p` *property*

Specifies the properties of the data replication Oracle Data Guard broker configuration.

For more information about the properties that you can set, see Appendix A, "Standard Disaster Recovery Framework Properties," in *Administering the Disaster Recovery Framework for Oracle Solaris Cluster 4.4*.

ODG-configuration-name

Specifies the name of the Oracle Data Guard broker configuration.

protection-group

Specifies the name of the protection group that contains the Oracle Data Guard broker configuration.

How to Delete an Oracle Data Guard Broker Configuration From an Oracle Data Guard Protection Group

Note - You can also accomplish this procedure by using the Oracle Solaris Cluster Manager browser interface. Click Partnerships, click the partnership name, click the protection group name, highlight the data replication component name, and click `Remove`. For more information about Oracle Solaris Cluster Manager, see Chapter 12, "Using the Oracle Solaris Cluster Manager Browser Interface" in *Administering an Oracle Solaris Cluster 4.4 Configuration*.

Before You Begin

Before you remove an Oracle Data Guard broker configuration from a protection group, ensure that the following conditions are met:

- The protection group is defined on the local cluster.
- If the partner cluster can be reached, the protection group is offline on the local cluster and the partner cluster.
- The Oracle Data Guard broker configuration is managed by the protection group.

For information about deleting protection groups, refer to "Deleting Protection Groups and Data Replication Components" in *Administering the Disaster Recovery Framework for Oracle Solaris Cluster 4.4*.

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

   For more information, see "Securing Disaster Recovery Framework Software" in *Installing and Configuring the Disaster Recovery Framework for Oracle Solaris Cluster 4.4*.

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

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

   The `/var/cluster/geo` directory must have the correct access control lists (ACL) applied for compatibility between the Geo Management rights profile and Oracle Data Guard.

2. **Remove the Oracle Data Guard broker configuration.**

   This command removes an Oracle Data Guard broker configuration from a protection group on the local cluster. The command then propagates the new configuration to the partner cluster if the partner cluster contains a protection group of the same name.

   This command removes the Oracle Data Guard broker configuration from the protection group. This command also deletes the shadow Oracle database-server resource group and replication resource group for this Oracle Data Guard broker configuration.

   ```
   phys-node-n# geopg remove-replication-component ODG-configuration-name protection-group
   ```

   ODG-configuration-name

   Specifies the name of the Oracle Data Guard broker configuration.

   protection-group

   Specifies the name of the protection group.

Example 5 Deleting an Oracle Data Guard Broker Configuration From an Oracle Data Guard Protection Group

This example shows how to delete an Oracle Data Guard broker configuration from an Oracle Data Guard protection group.

```
phys-paris-1# geopg remove-replication-component mysales.com sales-pg
```

Replicating the Oracle Data Guard Protection Group Configuration to a Partner Cluster

You can replicate the configuration of a protection group to the partner cluster either before or after you configure data replication, resource groups, and resources on both clusters.

▼ How to Replicate the Oracle Data Guard Protection Group Configuration to a Partner Cluster

Perform this procedure from a node of the cluster to which you want information replicated, for example, `phys-newyork-1`.

Note - You can also accomplish this procedure by using the Oracle Solaris Cluster Manager browser interface. Click Partnerships, then click the partnership name. In the Protection Groups section, click `Get Protection Groups` and select the protection group to replicate. For more information about Oracle Solaris Cluster Manager, see Chapter 12, "Using the Oracle Solaris Cluster Manager Browser Interface" in *Administering an Oracle Solaris Cluster 4.4 Configuration*.

Before You Begin Before you replicate the configuration of an Oracle Data Guard protection group to a partner cluster, ensure that the following conditions are met:

- The protection group is defined on the remote cluster, not on the local cluster.
- The Oracle Data Guard broker configuration in the protection group on the remote cluster exists on the local cluster.
- The application resource groups in the protection group on the remote cluster exist on the local cluster.
- The `Auto_start_on_new_cluster` property of the resource groups is set to `False`. You can view this property by using the `clresourcegroup show` command.

  ```
  phys-node-n# clresourcegroup show -p Auto_start_on_new_cluster apprg
  ```

  Set the `Auto_start_on_new_cluster` property to `False` as follows:

  ```
  phys-node-n# clresourcegroup set -y Auto_start_on_new_cluster=False apprg1
  ```

  Setting the `Auto_start_on_new_cluster` property to `False` prevents the Oracle Solaris Cluster resource group manager from automatically starting the resource groups in the

protection group. The disaster recovery framework restarts and communicates with the remote cluster to ensure that it is running and that it is the standby cluster for that resource group. The disaster recovery framework does not automatically start the resource group on the primary cluster.

When the protection group is activated, application resource groups need to be online only on the primary cluster.

- You have *not* added the shadow Oracle database-server resource group for an Oracle Data Guard broker configuration to a protection group application resource group list before that resource group exists on all clusters.

Note - You must replicate the protection group configuration to a partner cluster *before* you can add a shadow Oracle database-server resource group to a protection group.

When you successfully add the Oracle Data Guard configuration to the protection group on the clusters on which the protection group exists, Oracle Data Guard creates the shadow Oracle database-server resource group on the clusters. The means by which you can successfully add a shadow Oracle database-server resource group to a protection group include the following:

- If an Oracle Data Guard protection group contains an Oracle Data Guard broker configuration, when you replicate the protection group to the partner cluster, the disaster recovery framework module for Oracle Data Guard creates any missing shadow Oracle database-server resource group on the partner cluster.
- If an Oracle Data Guard protection group does not contain an Oracle Data Guard broker configuration, once you replicate the protection group on the partner cluster and add the Oracle Data Guard broker configuration to it, the disaster recovery framework module for Oracle Data Guard adds the shadow Oracle database-server resource group on both clusters.

Once a shadow Oracle database-server resource group exists on both clusters, you can add that resource group to the protection group.

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

   For more information, see "Securing Disaster Recovery Framework Software" in *Installing and Configuring the Disaster Recovery Framework for Oracle Solaris Cluster 4.4*.

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

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

The `/var/cluster/geo` directory must have the correct access control lists (ACL) applied for compatibility between the Geo Management rights profile and Oracle Data Guard.

2. **Replicate the protection group configuration to the partner cluster.**

```
phys-newyork-1# geopg get -s partnership ODG-protection-group
```

`-s` *partnership*

Specifies the name of the partnership from which the protection group configuration information is gathered.

ODG-protection-group

Specifies the name of the protection group.

The `geopg get` command retrieves the configuration information of the protection group from the remote cluster and creates the protection group on the local cluster. If the corresponding Oracle Data Guard configuration is enabled, the `geopg get` command also disables the Oracle Data Guard configuration.

The `geopg get` command replicates disaster recovery framework related entities. To replicate Oracle Solaris Cluster resource groups, resource types, and resources, use the `cluster export -t rg,rt,rs` command to generate an XML cluster configuration file, modify the XML file for the expected configuration on the secondary cluster. Then run the `clresource create` command with the `-a` option to apply the configuration updates. For more information, see "How to Configure Oracle Solaris Cluster Software on All Nodes (XML)" in *Installing and Configuring an Oracle Solaris Cluster 4.4 Environment* and the `cluster`(8CL)and `clresource`(8CL)man pages.

Example 6 Replicating the Oracle Data Guard Protection Group Configuration to a Partner Cluster

This example shows how to replicate the configuration of `sales-pg` to `cluster-newyork`.

```
# ssh root@phys-newyork-1
phys-newyork-1# geopg get -s paris-newyork-ps sales-pg
```

The configuration of the protection group is retrieved from the remote cluster, in this example `cluster-paris`, and then validated by the data replication subsystem on the local cluster `cluster-newyork`.

- If the validation is successful, the `Configuration` status is set to `OK` and the protection group is created on the local cluster.
- If the validation fails, the protection group is not created on the local cluster. Resolve the error and replicate the protection group again.

Checking the Runtime Status of Oracle Data Guard Data Replication

You can obtain an overall view of the status of replication, as well as a more detailed runtime status of the Oracle Data Guard software from the status of the replication resource groups. The following sections describe how to check the runtime status of replication:

- "Displaying an Oracle Data Guard Runtime Status Overview" on page 71
- "Displaying a Detailed Oracle Data Guard Runtime Status" on page 72
- "Enabling Debugging for Runtime Messages" on page 75

Displaying an Oracle Data Guard Runtime Status Overview

The status of each Oracle Data Guard data replication resource indicates the status of replication on a particular Oracle Data Guard broker configuration. The status of all the resources under a protection group are aggregated in the replication status.

To view the overall status of replication, look at the protection group state, as described in the following procedure.

▼ How to Check the Overall Runtime Status of Replication

Note - You can also accomplish this procedure by using the Oracle Solaris Cluster Manager browser interface. Click Partnerships, click the partnership name, and scroll to the Protection Groups section. For additional details, click the protection group name. For more information about Oracle Solaris Cluster Manager, see Chapter 12, "Using the Oracle Solaris Cluster Manager Browser Interface" in *Administering an Oracle Solaris Cluster 4.4 Configuration*.

---

1. **Log in to a node of a cluster where the protection group is defined.**

   To complete this step, you need to be assigned the Basic Solaris User rights profile. For more information see "Securing Disaster Recovery Framework Software" in *Installing and Configuring the Disaster Recovery Framework for Oracle Solaris Cluster 4.4*.

2. **Check the runtime status of replication.**

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

   Refer to the `Protection Group` section of the output for replication information. The output of this command includes the following information:

   - Whether the local cluster is enabled for partnership participation
   - Whether the local cluster is involved in a partnership
   - Status of the heartbeat configuration
   - Status of the defined protection groups
   - Status of current transactions

3. **Check the runtime status of data replication for each Oracle Data Guard protection group.**

   ```
   phys-paris-1 clresource status -v ODG-configuration-name-odg-rep-rs
   ```

   Refer to the `Status` and `StatusMessage` fields that are presented for the Oracle Data Guard broker configuration data replications that you want to check. For more information about these fields, see Table 3, "Status and Status Messages of an Online Oracle Data Guard Replication Resource Group," on page 73.

Displaying a Detailed Oracle Data Guard Runtime Status

One replication resource group exists for each protection group. The name of the replication resource group conforms to the following format:

ODG-protection-group`-odg-rep-rg`

If you add an Oracle Data Guard broker configuration to a protection group, the disaster recovery framework creates a resource for that configuration. This resource monitors and displays the status of replication for the Oracle Data Guard broker configuration. The name of each resource conforms to the following format:

ODG-configuration-name`-odg-rep-rs`

You can monitor the state of the replication resource to give you the overall status of replication. Use the `clresource status` command as follows to obtain the `State` and `Status Message` values for the replication status of the Oracle Data Guard broker configuration:

```
phys-node-n# clresource status -v ODG-configuration-name-odg-rep-rs
```

The `State` is `Online` while the resource is online.

The following table describes the `Status` and `Status Message` values that are returned by the `clresource status` command when the `State` of the Oracle Data Guard replication resource group is `Online`.

TABLE 3 Status and Status Messages of an Online Oracle Data Guard Replication Resource Group

Status	Status Message	Possible Causes
`Faulted`	Program *program-name* returned a nonzero exit code	
`Faulted`	Protection mode "*replication-mode*" given for local database *database* does not match configured value "*replication-mode*"	The Oracle Data Guard broker configuration has been changed by using the Oracle Data Guard command-line interface (`dgmgrl`) and has not been updated in the disaster recovery framework.
`Faulted`	Database *database* does not exist in the configured Oracle Data Guard database list "*List-of-databases*"	The database has been deleted from the Oracle Data Guard broker configuration using the Oracle Data Guard command-line interface (`dgmgrl`).
`Faulted`	Oracle errors "*List-of-ORA-xxxxx-errors*" were found in the Oracle Data Guard broker (`dgmgrl`) output when connecting by using "*connect-string*"	
`Faulted`	Role "*role*" given for database *database* does not match role "*role*" configured for Oracle Data Guard	The database might have been changed from a physical standby to a snapshot standby.
`Unknown`	Unexpected error - *unexpected-error*	
`Unknown`	The Oracle Data Guard broker (`dgmgrl` *connect-string*) did not complete a response to the command "*command-string*" within "*number*" seconds and was timed out.	The Oracle Data Guard command-line interface (`dgmgrl`) did not respond to the `show configuration` command within the specified time, or the Oracle Data Guard broker was busy performing a health check during this period.
`Unknown`	Password or connect name (*connect-string*) for remote cluster is incorrect	The `sysdba_username`, `sysdba_password`, `local_db_service_name`, or `remote_db_service_name` parameter does not match the information that is maintained by the disaster recovery framework.

Status	Status Message	Possible Causes
`Unknown`	File *filename* does not exist	A temporary internal file that is used by the Oracle Data Guard module was deleted before it could be read.
`Unknown`	A switchover is in progress	Self-explanatory.
`Unknown`	A failover is in progress	Self-explanatory.
`Degraded`	Program *program-name* failed to read the Cluster Configuration Repository (CCR)	One of the programs that is used to retrieve information from the CCR failed.
`Degraded`	Failed to get password for `sysdba` user name for Oracle Data Guard configuration *ODG-configuration-name* in protection group *ODG-protection-group*	The field for the `sysdba_password` was not found in the Cluster Configuration Repository (CCR) or was longer than expected.
`Degraded`	Local cluster *cluster-name* is not primary for Oracle Data Guard configuration *ODG-configuration-name*	A switchover or failover has been performed in the Oracle Data Guard broker by using a command in the Oracle Data Guard command-line interface (`dgmgrl`), and the disaster recovery framework configuration has not been updated.
`Degraded`	Oracle Data Guard configuration name *ODG-configuration-name* found does not match *ODG-configuration-name*	
`Degraded`	Database *database-name* is in the disabled state	A database has been disabled in the Oracle Data Guard broker using a command in the Oracle Data Guard command-line interface (`dgmgrl`), and the disaster recovery framework configuration has not been updated.
`Degraded`	Oracle Data Guard configuration *ODG-configuration-name* is disabled on cluster *cluster-name*	The standby database in the Oracle Data Guard broker configuration has been disabled by using a command in the Oracle Data Guard command-line interface (`dgmgrl`), and the disaster recovery framework configuration has not been updated.
`Degraded`	Oracle Data Guard configuration *ODG-configuration-name* is disabled	The Oracle Data Guard broker configuration has been disabled by using a command in the Oracle Data Guard command-line interface (`dgmgrl`), and the disaster recovery framework configuration has not been updated.
`Degraded`	The `BystandersFollowRoleChange` property for the Oracle Data Guard broker configuration *Broker_Config_Name_Variable* must be set to 'NONE'	The Oracle Data Guard broker property is not set to 'NONE'.
`Degraded`	Fast-start failover must be disabled for the Oracle Data Guard broker configuration *Broker_Config_Name_Variable*	Fast-start failover is enabled.

Status	Status Message	Possible Causes
`Online`	Online or replicating in *replication-mode* mode	

For more information about the `clresource` command, see the `clresource`(8CL) man page.

Enabling Debugging for Runtime Messages

This section provides information to enable debug messages for the Oracle Data Guard control shell scripts as well as for the common agent container, and to verify that debugging is enabled.

▼ How to Enable Debugging for Runtime Messages

Perform this task on each cluster node where you want debugging active for Oracle Data Guard data replication.

1. **Assume the `root` role on the cluster node.**

2. **Determine whether debugging is active for Oracle Data Guard replication with the disaster recovery framework.**

```
# grep info /etc/syslog.conf
#
```

Debugging is active if `*.info` is set in the `/etc/syslog.conf` file.

3. **If debugging is inactive, add the `*.info` setting to the `/etc/syslog.conf` file.**

4. **Verify the addition of the `*.info` setting to the `/etc/syslog.conf` file.**

```
# grep info /etc/syslog.conf
*.err;kern.debug;daemon.notice;mail.crit,*.info /var/adm/messages
#
```

5. **Restart the `system-log` service.**

```
# svcadm restart system-log
```

Next Steps To disable debugging, remove the `*.info` setting from the `/etc/syslog.conf` file and restart the `system-log` service.

♦ ♦ ♦ CHAPTER 4

Migrating Services That Use Oracle Data Guard Data Replication

This chapter provides information about migrating services for maintenance or as a result of cluster failure.

This chapter covers the following topics:

Recovering Oracle Data Guard Data After a Takeover

After a successful takeover operation, the standby cluster, `cluster-newyork`, becomes the primary for the protection group, and the services are online on the standby cluster. After the recovery of the original primary cluster, the services can be brought online again on the original primary cluster by using a process called *failback*.

The disaster recovery framework supports the following two kinds of failback:

- **Failback switchover.** During a failback switchover, applications are brought online again on the original primary cluster, `cluster-paris`, after the primary cluster data has been resynchronized with the data on the standby cluster `cluster-newyork`.

  For a reminder of which clusters are `cluster-paris` and `cluster-newyork`, see "Example Disaster Recovery Framework Cluster Configuration" in *Administering the Disaster Recovery Framework for Oracle Solaris Cluster 4.4*.
- **Failback takeover.** During a failback takeover, applications are brought online again on the original primary cluster and use the current data on the primary cluster. Any updates that occurred on the standby cluster are discarded.

If you want to leave the new primary, `cluster-newyork`, as the primary cluster and the original primary cluster, `cluster-paris`, as the standby cluster after the original primary cluster starts again, you can resynchronize and revalidate the protection group configuration. You can resynchronize and revalidate the protection group without performing a switchover or takeover.

How to Perform a Failback Switchover or Failback Takeover

Follow this procedure to restart an application on the original primary cluster, `cluster-paris`.

Note - You can also accomplish some steps in this procedure by using the Oracle Solaris Cluster Manager browser interface. Click Partnerships, click the partnership name, highlight the protection group name, and click the button for the action you want to perform. For more information about Oracle Solaris Cluster Manager see Chapter 12, "Using the Oracle Solaris Cluster Manager Browser Interface" in *Administering an Oracle Solaris Cluster 4.4 Configuration*.

The failback procedures apply only to clusters in a partnership. Perform the following procedure only once for each partnership.

Before You Begin Ensure that the clusters have the following roles:

- The protection group on `cluster-newyork` is assigned the primary role.
- The protection group on `cluster-paris` has either the primary role or the secondary role, depending on whether the protection group could be reached during the takeover.

1. **If the original primary cluster, `cluster-paris`, failed, confirm that the cluster is restarted and that the disaster recovery framework is enabled on the cluster.**

   For more information about restarting a cluster, see "Booting a Cluster" in *Administering the Disaster Recovery Framework for Oracle Solaris Cluster 4.4*.

2. **For HA for Oracle Database, on the original primary cluster, verify that the `SUNW.oracle_server` resource is in a healthy state.**

   If HA for Oracle Database is not running on the original primary cluster, omit this step.

   - **If the resource is not in a faulted state, unmonitor the database resource.**

     ```
     # clresource unmonitor oracle_server-rs
     ```

- **If the resource is in a faulted state or repeatedly restarts, perform the following steps:**

  a. **Disable the HA for Oracle Database resource or resource group.**

     - **If the `dataguard_role` property is set to STANDBY, disable the HA for Oracle Database resource.**

       A STANDBY value is set if the takeover was performed when the old primary was running at the time of the takeover.

       ```
       phys-paris-1# clresource disable oracle_server-rs
       ```

     - **If the `dataguard_role` property is set to PRIMARY, disable the HA for Oracle Database resource group.**

       A PRIMARY value is set if the takeover was performed when the old primary was down during the takeover.

       ```
       phys-paris-1# clresourcegroup quiesce -k oracle_server-rg
       phys-paris-1# clresource disable oracle_server-rs
       phys-paris-1# clresourcegroup offline oracle_server-rg
       phys-paris-1# clresourcegroup online oracle_server-rg
       ```

Note - When the cluster restarts, an attempt is made to start a database that needs to be reinstated. Therefore, you must disable the resource as soon as possible. You might need to quiesce the HA for Oracle Database resource group if the RGM has already attempted to bring it online.

If the Oracle Database resource is in the `stop_failed` state, clear the `stop_failed` flag by using the following command.

```
phys-paris-1# clresource clear oracle_server-rs
```

---

  b. **Determine whether the database is shut down on the cluster nodes.**

  c. **If the database is not shut down, become the Oracle user on that node and stop the database by using one of the following methods:**

     First method:

     ```
     phys-paris-1$ srvctl stop database -d database_name
     ```

     Second method:

```
phys-paris-1$ ORACLE_SID=db_SID export ORACLE_SID
phys-paris-1$ sqlplus /nolog
SQL> connect sys/sysdba password as sysdba
SQL> shutdown immediate
SQL> exit
```

d. Start mount the database.

```
phys-paris-1$ sqlplus /nolog
SQL> connect sys/sysdba passord as sysdba
SQL> startup mount
…
SQL> exit
```

3. Reinstate the old Oracle Data Guard primary database to become the standby for the current primary database.

If you issue the `dgmgrl` command from the old primary cluster, include the new primary's database service name in the connection string.

```
phys-newyork-1$ dgmgrl
DGMGRL> connect sys/password[@new_primary_service_name]
DGMGRL> reinstate database old_primary_database_name
...
DGMGRL> exit
```

Note - If the database cannot be reinstated, you might need to re-create it or otherwise recover the database by using an appropriate method. For instructions, refer to "Using Flashback Database After a Failover" in *Oracle Data Guard Concepts and Administration*.

---

4. To perform a failback takeover instead of a failback switchover, flashback your primary database to the point at which the original takeover occurred.

5. For HA for Oracle Database, update and re-enable the HA for Oracle Database resource on the original primary cluster.

If HA for Oracle Database is not running on the original primary cluster, omit this step.

```
phys-paris-1# clresource set -p dataguard_role=STANDBY oracle_server-rs
phys-paris-1# clresource enable oracle_server-rs
    Restore monitoring if monitoring of the resource was previously disabled
phys-paris-1# clresource monitor oracle_server-rs
```

6. If the original primary cluster was down at the point of failure, update the original primary cluster to be the secondary.

a. From a node of the original primary cluster, stop the protection group.

If the original primary cluster was down at the time of takeover, the protection group should already be stopped.

```
phys-paris-1# geopg stop -e local protection-group
```

`-e local`

Specifies the scope of the command. By specifying a local scope, the command operates on the local cluster only.

protection-group

Specifies the name of the protection group.

b. Verify that the protection group is stopped.

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

c. Update the protection group.

```
phys-paris-1# geopg update protection-group
```

The roles are now correct, but both clusters are marked as deactivated.

For more information about synchronizing protection groups, see "Resynchronizing a Protection Group" in *Administering the Disaster Recovery Framework for Oracle Solaris Cluster 4.4*.

7. From one node in each cluster, locally validate the configuration for each protection group.

Note - Ensure that the protection group is not in an `Error` state. You cannot start a protection group when it is in an `Error` state.

```
phys-paris-1# geopg validate protection-group
phys-newyork-1# geopg validate protection-group
```

For more information, see "How to Validate an Oracle Data Guard Protection Group" on page 51.

8. From one node in either cluster, globally activate the protection group on both clusters.

```
phys-node-n# geopg start -e global protection-group
```

9. From one node in either cluster, switch over the protection group to the original primary.

```
phys-node-n# geopg switchover -f -m cluster-paris protection-group
```

For more information, see "Migrating Replication Services by Switching Over Protection Groups" in *Administering the Disaster Recovery Framework for Oracle Solaris Cluster 4.4*.

The `cluster-paris` cluster resumes its original role as primary cluster for the protection group.

10. **Ensure that the switchover was performed successfully.**

```
phys-node-n# geoadm status
```

Verify that the protection group is now primary on `cluster-paris` and secondary on `cluster-newyork` and that the states that are shown for the `Data replication` and the `Resource groups` properties are `OK` on both clusters.

Recovering From an Oracle Data Guard Data Replication Error

When an error occurs at the data replication level, the error is reflected in the status of the resource in the replication resource group of the relevant the Oracle Data Guard broker configuration.

For example, suppose that the Oracle Data Guard broker configuration `sales-pg`, which contains the replicated database `sales`, is changed from protection mode `MaxAvailability` to `MaxPerformance`. The state changes for `FAULTED` are reflected in the following resource status:

```
Resource Status = "FAULTED"
Resource status message = "FAULTED - Protection mode "MaxAvailability" given
for local database sales does not match configured value "MaxPerformance""
```

Note - The `Resource State` remains `Online` because the probe is still running correctly.

Because the resource status has changed, the protection group status also changes. In this case, the local `Data Replication` state, the `Protection Group` state on the local cluster, and the overall `Protection Group` state all become `Error`.

To recover from an error state, perform the following procedure.

How to Recover From a Data Replication Error

1. **Use the procedures in the Oracle Data Guard documentation to determine the causes of the `FAULTED` state.**

2. **Recover from the faulted state by following the Oracle Data Guard procedures.**

   Note - You can also accomplish this procedure by using the Oracle Solaris Cluster Manager browser interface. Click Partnerships, click the partnership name, click the protection group name, click the data replication component name, and click `Edit`. For more information about Oracle Solaris Cluster Manager, see Chapter 12, "Using the Oracle Solaris Cluster Manager Browser Interface" in *Administering an Oracle Solaris Cluster 4.4 Configuration*.

   If the recovery procedures change the state of the Oracle Data Guard broker configuration, this state is automatically detected by the resource and is reported as a new protection group state. If the replication mode does not match the disaster recovery framework settings, type:

   ```
   phys-paris-1# geopg modify-replication-component \
   ODG-configuration-name  protection-group
   ```

3. **Revalidate the protection group configuration.**

   Note - You can also accomplish this procedure by using the Oracle Solaris Cluster Manager browser interface. Click Partnerships, click the partnership name, highlight the protection group name, and click `Validate`. For more information about Oracle Solaris Cluster Manager, see Chapter 12, "Using the Oracle Solaris Cluster Manager Browser Interface" in *Administering an Oracle Solaris Cluster 4.4 Configuration*.

   ```
   phys-paris-1# geopg validate protection-group
   ```

   where *protection-group* specifies the name of the Oracle Data Guard protection group.

4. **Review the status of the protection group configuration.**

   ```
   phys-paris-1# geopg list protection-group
   ```

   where *protection-group* specifies the name of the Oracle Data Guard protection group.

Synchronizing Roles Between an Oracle Data Guard Configuration and its Disaster Recovery Framework Protection Group

Under certain circumstances after a switchover, the protection group roles might no longer match the database roles. Examples of when this might happen are if the Database Administrator issues an Oracle Data Guard switchover directly, or if checks run during a protection-group switchover fail but the switchover successfully completes. If the protection group and database roles no longer match, you must manually synchronize them.

▼ How to Synchronize the Roles of an Oracle Data Guard Configuration and its Disaster Recovery Framework Protection Group

Perform this procedure to change the roles of a disaster recovery framework protection group to match the roles of an Oracle Data Guard configuration that has been switched.

1. **Assume the `root` role on the Oracle Data Guard primary where the Oracle Data Guard protection group has a secondary role.**

2. **Compare the database and protection-group roles to confirm that they do not match.**

   `# clresource status` *odg-configuration-odg-rep-rs*

   a. **The `siamst_siamstdr-odg-rep-rs` shows:**

      `Faulted - Role "physical standby database" given for database` *local-database* `does not match role "primary database" configured Oracle Data Guard.`

   b. **Display the protection group status.**

```
# geadm status
   Cluster: cluster-paris

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

```
        Heartbeat "hb_cluster-paris~cluster-newyork" monitoring "cluster-newyork": OK
             Plug-in "ping_plugin"      : Inactive
             Plug-in "tcp_udp_plugin"   : OK

    Protection group "odg-pg"    : Error
         Partnership             : paris-newyork-ps
         Synchronization         : OK

         Cluster cluster-paris           : Error
              Role               : Secondary
              Activation state   : Activated
              Configuration      : Error
              Data replication   : Error
              Resource groups    : OK

         Cluster cluster-newyork         : Error
              Role               : Primary
              Activation state   : Deactivated
              Configuration      : Error
              Data replication   : Error
              Resource groups    : OK
```

c. Display the database configuration.

```
DGMGRL> show configuration;
Configuration
  Name:               odg-configuration
  Enabled:            YES
  Protection Mode:    MaxPerformance
  Databases:
    local-database    - Primary database
    remote-database - Physical standby database

Fast-Start Failover: DISABLED

Current status for "odg-configuration":
SUCCESS

DGMGRL>
```

3. From one node of the protection group's secondary cluster, deactivate the protection group locally.

```
phys-paris-1# geopg stop -e local odg-pg
```

4. From one node of the protection group's secondary cluster, issue a takeover.

```
phys-paris-1# geopg takeover -f odg-pg
```

```
Processing operation... The timeout period for this operation on each cluster
is 3600 seconds (3600000 milliseconds)…
"Takeover" operation succeeded for the protection group "odg-pg".
phys-paris-1#
```

5. **From one node of the protection group's primary and secondary cluster, validate the protection group.**

```
phys-paris-1# geopg validate odg-pg
phys-newyork-1# geopg validate odg-pg
```

6. **From one node of either cluster start pg globally.**

```
phys-paris-1# geopg start -e global odg-pg
Processing operation... The timeout period for this operation on each cluster
is 3600 seconds (3600000 milliseconds)…
Protection group "odg-pg" successfully started.
phys-paris-1#
```

7. **Verify the protection group status.**

```
phys-paris-1# geoadm status
   Cluster: cluster-paris

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

       Heartbeat "hb_cluster-paris~cluster-newyork" monitoring "cluster-newyork": OK
            Plug-in "ping_plugin"      : Inactive
            Plug-in "tcp_udp_plugin"   : OK

   Protection group "odg-pg"    : OK
        Partnership             : paris-newyork-ps
        Synchronization         : OK

        Cluster cluster-paris           : OK
             Role               : Primary
             Activation state   : Activated
             Configuration      : OK
             Data replication   : OK
             Resource groups    : OK

        Cluster cluster-newyork         : OK
             Role               : Secondary
             Activation state   : Activated
             Configuration      : OK
```

```
Data replication   : OK
Resource groups    : OK
```

♦ ♦ ♦ APPENDIX A

Disaster Recovery Framework Properties for Oracle Data Guard Broker Configurations

This appendix describes the properties for disaster recovery framework data replications that use Oracle Data Guard.

Oracle Data Guard Broker Replication Component Properties

This section describes the Oracle Data Guard broker replication component properties that the disaster recovery framework defines.

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 database on the local cluster.

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.

Category:

Required

Default:

None

Tunable:

Any time

Data replication component property: `local_oracle_svr_rg_name` (string)

Name of the local Oracle database server resource group that manages the local database in the Oracle Data Guard broker configuration. A shadow Oracle database-server resource group shadows the real resource group. If you want, add the shadow to the protection group application resource group list.

Note - The previous name of this property, `local_rac_proxy_svr_rg_name`, is still valid.

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 database on the remote cluster.

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.

Category:

Required

Default:

None

Tunable:

Any time

Data replication component property: `remote_oracle_svr_rg_name` (string)

Name of the remote Oracle database-server resource group on the partner cluster that manages the remote database in the Oracle Data Guard broker configuration. A shadow Oracle database-server resource group shadows the real resource group. If you want, add the shadow to the protection group application resource group list.

Note - The previous name of this property, `remote_rac_proxy_svr_rg_name`, is still valid.

Category:

Required

Default:

None

Tunable:

At creation

Data replication component property: `standby_type` (string)

Type of Oracle standby database that is used in the Oracle Data Guard broker configuration.

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

Category:

Required

Default:

None

Tunable:

Any time

Data replication component property: `sysdba_password` (`string`)

Password for the Oracle `SYSDBA` privileged database user.

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 if an Oracle wallet is not used

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. Use this property to monitor and manage the Oracle Data Guard broker configurations.

Category:

Required if an Oracle wallet is not used

Default:

None

Tunable:

Any time

Index

O

P

R

S

T

V