Oracle® Solaris Cluster Data Service for Oracle GoldenGate Guide

ORACLE®

Contents

Using This Documentation

- **Overview** – Describes how to use the Oracle GoldenGate software to create a highly available, custom Oracle Solaris Cluster data service.
- **Audience** – Technicians, system administrators, and authorized service providers
- **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

Late-breaking information and known issues for this product are included in the documentation library at http://www.oracle.com/pls/topic/lookup?ctx=E39579.

Access to Oracle Support

Oracle customers 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.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program web site at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Feedback

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

♦♦♦ C H A P T E R   1

1

Installing and Configuring HA for Oracle GoldenGate

This chapter explains how to install and configure HA for Oracle GoldenGate.

Note - You can install and configure this data service to run in either the global zone or a zone cluster. For updated information about supported configurations of this data service, see the *Oracle Solaris Cluster 4 Compatibility Guide*.

This chapter contains the following sections.

HA for Oracle GoldenGate Overview

The Oracle GoldenGate software defines an instance that consists of the following processes:

- Manager
- Extract
- Replicat
- Server (needed only if the replicat processess are configured)

The manager controls the other processes, including the start and stop functions.

The extract process pulls data out of a database and stores it either locally or remotely. If the extract process has to store the data locally, it does so by writing it to a file. If the extract process has to store the data remotely, it communicates with a remote server process and gets the server to store the data in files. The files that contain the data are called `trails`.

A replicat process picks up the information from a `trail` and stores it into a target database.

A healthy Oracle GoldenGate instance has at least the manager up and running. Although such an instance is valid, it cannot process any data. A fully working Oracle GoldenGate instance has at least an extract process or a replicat process up and running. You need at least two Oracle GoldenGate instances; one to serve as a sender and the other to serve as the receiver. The two Oracle GoldenGate instances can be configured either on the same cluster or on two different clusters. Also, it is valid to have one instance in a cluster and the other instance unclustered.

Note - The Oracle GoldenGate instance cannot exist on its own and must always be co-located with a database.

The Oracle GoldenGate instance and the underlying components have data services that protect them in an Oracle Solaris Cluster configuration, as described in the following table.

TABLE 1-1   Protection of Oracle GoldenGate Components by Oracle Solaris Cluster Data Services

Cluster Component	Data Service
Oracle GoldenGate instance	Resource type for Oracle GoldenGate. The resource type is `ORCL.GoldenGate`.
Database	The data service for the database that you are using. For example: ■ For the MySQL database, the data service is Oracle Solaris Cluster HA for MySQL. For more information about this dataservice, see "Oracle Solaris Cluster Data Service for MySQL Guide ". For the Oracle database, the data service is Oracle Solaris Cluster HA for Oracle. For more information about this data service, see "Oracle Solaris Cluster Data Service for Oracle Database Guide ".

Overview of the Installation and Configuration Process for HA for Oracle GoldenGate

The following table summarizes the tasks for installing and configuring HA for Oracle GoldenGate and provides cross-references to detailed instructions for performing these tasks. Perform the tasks in the order that they are listed in the table.

TABLE 1-2 Tasks for Installing and Configuring HA for Oracle GoldenGate

Task	For Instructions, Go To …
Plan the Oracle GoldenGate installation	Your `Oracle GoldenGate` documentation "Oracle Solaris Cluster Data Services Planning and Administration Guide " "Planning the HA for Oracle GoldenGate Installation and Configuration" on page 11
Install Oracle GoldenGate software and configure the application to run in a cluster	"Installing and Configuring Oracle GoldenGate" on page 14
Verify the Oracle GoldenGate software installation and configuration	"Verifying the Oracle GoldenGate Installation and Configuration" on page 18
Register and configure the HA for Oracle GoldenGate data service	"Registering and Configuring HA for Oracle GoldenGate" on page 20
(Optional) Use alternate project identifier (ID)	"Using Alternate Project Identification" on page 23
(Optional) Tune the HA for Oracle Golden Gate fault monitors	"Tuning the HA for Oracle GoldenGate Fault Monitors" on page 24
Verify the HA for Oracle GoldenGate installation and configuration	"Verifying the HA for Oracle GoldenGate Installation and Configuration" on page 26

Planning the HA for Oracle GoldenGate Installation and Configuration

This section contains the information that you need to plan your HA for Oracle GoldenGate installation and configuration.

Note - Before you begin, consult your `Oracle GoldenGate` documentation for configuration restrictions and requirements that are not stated in the Oracle Solaris Cluster documentation or imposed by the Oracle Solaris Cluster software.

Note - HA for Oracle GoldenGate can be configured to run in either a global zone or a zone cluster.

Configuration Restrictions

Caution - Your data service configuration might not be supported if you do not observe these restrictions.

For restrictions that apply to all data services, see the release notes for your release of Oracle Solaris Cluster. The following configuration restriction applies only to HA for Oracle GoldenGate.

■ The Oracle GoldenGate resource must be co-located with the database resource. To achieve this in a failover configuration, you can always place them in the same resource group. If you do not want the resources in the same resource group, define strong positive affinities or strong positive affinities with failover delegation between the resource groups.

■ Ensure that the Oracle GoldenGate user login process is not interactive. For example, the `su - gg-user` command must not require any additional inputs from the user.

Configuration Requirements

Caution - Configuration requirements represent actions that you must perform. If your data service configuration does not satisfy these requirements, the data service configuration might not be supported.

For requirements that apply to all data services, see "Configuration Guidelines for Oracle Solaris Cluster Data Services" in "Oracle Solaris Cluster Data Services Planning and Administration Guide ".

The following configuration requirements apply only to HA for Oracle GoldenGate.

■ The Oracle GoldenGate manager must be configured to restart abended extract and replicat processes.

■ Replications between clusters or within a cluster must be started by the HA for Oracle GoldenGate.

■ The PATH variable that is set in the Oracle GoldenGate user environment must not contain paths from the NFS servers. If the NFS servers hang for some reason, then it can cause availability issues for HA for Oracle GoldenGate.

■ The resource default failover mode implies that an Oracle GoldenGate resource can trigger a failover of the resource group. This default mode will disrupt the client access to the underlying database. If this behavior is not acceptable, configure the `Failover_mode` property to `restart_only` or `log_only`.

Note - The criteria to pick a desired behavior depends on the importance of the underlying database. If undisrupted client access to the underlying database is more important than undisrupted replication, change the `Failover_mode` of the Oracle GoldenGate resource.

Supported Data Service Configurations

HA for Oracle GoldenGate supports configurations that conform to the requirements that are specified in the previous section.

All configurations of HA for Oracle GoldenGate have the following requirements:

- The Oracle GoldenGate instance is configured as a failover resource.
- The Oracle GoldenGate instance is configured as a multiple master resource.

Since Oracle GoldenGate does not have any other client access besides the administrative one, it is not practical to have a scalable configuration.

Configuration Considerations

The following configuration considerations affect the installation and configuration of HA for Oracle GoldenGate.

- In failover configurations, the binaries of the Oracle GoldenGate software can reside on local storage. The configuration data and the `trail` files must reside on shared storage.

  See "Configuration Guidelines for Oracle Solaris Cluster Data Services" in "Oracle Solaris Cluster Data Services Planning and Administration Guide ". This section details the advantages and disadvantages of placing the Oracle GoldenGate binaries on the local file system or on the cluster file system.
- In failover configurations, you need at least one corresponding Oracle GoldenGate instance. The second instance can be configured outside of a cluster, although it is highly recommended to make it highly available as well. If you plan to make the other instance highly available also, you will need to repeat the instructions in this guide.
- In multiple master configurations, the complete Oracle GoldenGate software including the configuration data and the `trail` files can reside in the local storage.

Configuration Planning Questions

Use the questions in this section to plan the installation and configuration of HA for Oracle GoldenGate.

- Will you run the Oracle GoldenGate instance as a failover resource or as a multiple master resource?
- What is the logical hostname in a failover configuration for the corresponding Oracle GoldenGate instances that will access the data service?
- Where will the Oracle GoldenGate configuration files reside?

  See "Configuration Guidelines for Oracle Solaris Cluster Data Services" in "Oracle Solaris Cluster Data Services Planning and Administration Guide " for the advantages and

disadvantages of placing the Oracle GoldenGate binaries on the local file system or on the cluster file system.

Installing and Configuring Oracle GoldenGate

To enable HA for Oracle GoldenGate to make Oracle GoldenGate highly available, additional installation and configuration operations are required. These operations supplement the standard installation and standard configuration of the Oracle GoldenGate instance.

The procedure in this section describes the installation and configuration of an Oracle GoldenGate instance.

▼ How to Enable the Oracle GoldenGate Instance to Run in a Cluster

The following resource group is created in this procedure.

- Storage Resource group to contain the following resource:
  - Storage resource (required in the case of failover resource groups, but optional for multiple master resource groups)
- Database resource group, to contain the following resources:
  - Database resource
  - GoldenGate resource
  - Logical hostname for the resources (required in the case of failover resource groups)

Note - Depending on whether the file system that is managed by the storage resource is a global file system or a failover file system, you might need to add affinities between the database resource group and the storage resource group. You do have an option to combine the storage resource group and the database resource group into one resource group.

Before You Begin   Ensure that the /etc/netmasks file has IP-address subnet and netmask entries for all logical hostnames. If necessary, edit the /etc/netmasks file to add any missing entries.

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

2. **Create the resource group for your storage topology.**

    In failover topologies, Oracle GoldenGate requires shared storage, so you **must** make your storage highly available. For example, you can create a resource group containing the HAStoragePlus resource managing the global file systems.

> **Note -** If you plan to create a failover file system, place the storage resource in the same resource group as the Oracle GoldenGate resource.

```
# clresourcegroup create storage-rg
```

storage-rg

   Specifies the name of the resource group to be created. Choose a unique name for each resource group within the cluster.

3. **Register the `SUNW.HAStoragePlus` resource type.**

```
# clresourcetype register SUNW.HAStoragePlus
```

4. **Create your storage resource.**

```
# clresource create -g storage-rg \
-t SUNW.HAStoragePlus \
-p filesystemmountpoints=mountpoint-list \
hsp-rs
```

-g *storage-rg*

   Specifies that the resource is to be added to the Oracle GoldenGate storage resource group.

-t *SUNW.HAStoragePlus*

   Specifies that the resource is an instance of the `SUNW.HAStoragePlus` resource type.

-p filesystemmountpoints=*mountpoint-list*

   Specifies a list of valid mount points for the file system. For more information, see the SUNW.HAStoragePlus(5) man page.

hsp-rs

   Specifies the name of the resource that you are creating.

The resource is created in the enabled state.

5. **Bring online the storage resource group.**

```
# clresourcegroup online -M storage-rg
```

-M

   Specifies that the resource group is to be brought into the managed state.

storage-rg

   Specifies the name of the resource group to be brought online.

6. Create a resource group for the database and the Oracle GoldenGate resource.

The Oracle GoldenGate instance resource group can be configured as a failover or multiple master resource group to contain the resource for the database and the Oracle GoldenGate resource. If the Oracle GoldenGate instance is configured as a failover resource, the logical hostname for this resource is also contained in the Oracle GoldenGate resource group.

- To create a failover resource group, run the following command:

  `# `**`clresourcegroup create`**` db-rg`

- To create a multiple master resource group, run the following command:

  `# `**`clresourcegroup create`**` \`
  `-p Maximum_primaries=``value \`
  `-p Desired_primaries=``value \`
  `db-rg`

`-p Maximum_primaries=`*value*

Specifies the maximum number of primary nodes for this multiple master resource group.

`-p Desired_primaries=`*value*

Specifies the desired number of primary nodes for this multiple master resource group.

db-rg

Specifies the name of the resource group to be added. This name can be your choice but must be unique for the resource groups within the cluster.

7. If the resource group that you created in Step 6 is configured as a failover resource group, add a logical hostname resource to the resource group.

If the resource group that you created in Step 6 is configured as a multiple master resource group, skip this step.

`# `**`clreslogicalhostname create -g`**` db-rg db-logical-hostname`

`-g` *db-rg*

Specifies the name of the resource group.

db-logical-hostname

Specifies the logical hostname *db-logical-hostname* of the Oracle GoldenGate resource. By default, the name of the logical host resource is set to the same name as this. If you wish to specify a separate name, then place the name of the resource at the end of the command listing.

Note - If you require a fully qualified hostname, you must specify the fully qualified name with the `-h` option and you cannot use the fully qualified form in the resource name.

---

8. **Bring online the Oracle GoldenGate resource group.**

   ```
   # clresourcegroup online -M db-rg
   ```

   -M

       Specifies that the resource group is to be brought into the managed state.

   db-rg

       Specifies the name of the resource group that needs to be brought online.

9. **Install the database software.**

   Install your database software and make the database highly available using an Oracle Solaris Cluster resource.

   Tip - One option is to place the database resource in the same resource group where you plan on creating the Oracle GoldenGate resource.

   The following is a list of several databases and a link to the corresponding documentation.

   - If you plan to use Oracle as the database, configure the Oracle Solaris Cluster HA for Oracle data service. For more information about the data service, see "Oracle Solaris Cluster Data Service for Oracle Database Guide ".
   - If you plan to use MySQL as the database, configure the Oracle Solaris Cluster HA for MySQL data service. For more information about the data service, see "Oracle Solaris Cluster Data Service for MySQL Guide ".
   - You also can develop your own resource type or use a generic data service to make your database highly available. For more information, see "Oracle Solaris Cluster Data Services Developer's Guide ".

   If you have a specific agent associated with your database, then use the configuration procedure for the data service in your database.

10. **Install the Oracle GoldenGate software.**

    Follow the instructions in the Oracle GoldenGate installation manual to install the Oracle GoldenGate software. You will also need to configure the replication between the different Oracle GoldenGate instances.

    Note - For failover configurations, ensure that you replicate every user and group to all the servers on which you plan to run Oracle GoldenGate.

Verifying the Oracle GoldenGate Installation and Configuration

Before you install the HA for Oracle GoldenGate packages, verify that the Oracle GoldenGate software is correctly installed and configured to run in a cluster. These procedures do *not* verify that the Oracle GoldenGate application is highly available because you have not yet installed your data service.

Note - For more information, see the `GoldenGate documentation`.

The procedure in this section verifies the installation and configuration of the Oracle GoldenGate components.

▼ How to Verify the Installation of an Oracle GoldenGate Instance

On the node where you installed the Oracle GoldenGate instance, as the root user, do the following:

1. **Switch to the Oracle GoldenGate administrative user.**

   `# su - goldengate`

2. **Run the `ggsci` utility.**

   $ *installation directory*`/ggsci`

3. **Stop the extract and the replicat processes.**

   $ `GGSCI (`*node*`) 1>  stop er *`

4. **Stop the manager processes.**

   $ `GGSCI (`*node*`) 2>  stop mgr !`

5. **Exit the `ggsci` utility.**

   $ `GGSCI (`*node*`) 3>  exit`

6. **If the replicat processes are configured, kill all remaining server processes.**

   $ `pkill -f` *installation directory*

7. **Start the manager as well as the configured extract and the replicat processes using the `ggsci` utility.**

```
$ installation directory/ggsci
GGSCI (node) 1>  start mgr
GGSCI (node) 2>  start er *
```

8. **Check if the instance goes online.**

```
$ GGSCI (node) 3>  info all
```

Repeat this until all processes show the status as RUNNING.

9. **Stop the Oracle GoldenGate instance using the ggsci utility.**

```
$ installation directory/ggsci
GGSCI (node) 1>  stop er *
GGSCI (node) 2>  stop mgr !
GGSCI (node) 3>  exit
```

10. **If the replicat processes are configured, kill all remaining server processes.**

```
$ pkill -f installation directory
```

Preparing the Corresponding Oracle GoldenGate Instance for High Availability

Oracle GoldenGate is a replication software. In general, you need at least two instances of Oracle GoldenGate for the software to be fully functional.

The corresponding Oracle GoldenGate instance can be configured either as a failover or as a multiple master.

For failover configuration, the following two options are available:

■   The corresponding Oracle GoldenGate instance can to be configured within the same cluster or in a different cluster. If you plan on using a different cluster for configuring the corresponding Oracle GoldenGate instance, then repeat the configuration steps as described in this document.

■   If you do not plan on making the corresponding Oracle GoldenGate instance highly available, then just install and configure Oracle GoldenGate on the unclustered server. If you do plan on making the corresponding Oracle GoldenGate instance highly available on a platform other than Oracle Solaris Cluster, then follow the documentation for the other platform.

In a multiple master configuration, you will also have a multiple master configuration for your database. Install and configure the replication for the Oracle GoldenGate instances for every database that you made highly available. You might have to configure replication with Oracle GoldenGate between the databases within this cluster, or just have the databases as replication

targets for a different Oracle GoldenGate instance, depending on the Oracle GoldenGate replication topology.

Registering and Configuring HA for Oracle GoldenGate

To enable the HA for Oracle GoldenGate data service to make Oracle GoldenGate applications highly available, configure the data service as described in the procedure in this section. The procedure performs the following task.

- Register and configure the Oracle GoldenGate resource.

The resource groups and the storage resource were created in "How to Enable the Oracle GoldenGate Instance to Run in a Cluster" on page 14.

Before You Begin

Before performing these procedures, ensure that the HA for Oracle GoldenGate data service packages are installed.

The procedures in this section assume that the database resources have been previously created and that these resources are online.

Make your database highly available by following the instructions in the documentation for your database.

Perform these procedures as a superuser on a cluster node.

Setting HA for Oracle GoldenGate Extension Properties

The extension properties for HA for Oracle GoldenGate are described in Appendix A, "HA for Oracle GoldenGate Extension Properties".

You can update some extension properties dynamically. You can update other properties, however, only when you create or disable a resource. The Tunable entry indicates when you can update a property.

The instructions for registering and configuring resources explain how to set *only* extension properties that HA for Oracle GoldenGate requires you to set.

To set an extension property of a resource, include the following option in the `clresource` command that creates or modifies the resource:

`-p` *property*=*value*

-p *property*

> Identifies the extension property that you are setting.

value

> Specifies the value to which you are setting the extension property.

Registering and Configuring the HA for Oracle GoldenGate by using Oracle Solaris Cluster Command Line Interface (CLI)

The procedures in this section describe the registration and configuration of the following components of the Oracle GoldenGate by using Oracle Solaris Cluster CLI.

▼ How to Register and Configure the Oracle GoldenGate Resource

The Oracle GoldenGate resource must be co-located with the database resource. You can achieve this by setting strong positive affinities between the two resource groups, or by placing both resources in the same resource group. The Oracle GoldenGate resource must depend on the database resource, to ensure that it gets started after the database.

You have to configure the Oracle GoldenGate resource to depend on its storage resource. This dependency ensures that the Oracle GoldenGate resource does not attempt to start until its associated storage resource is online.

1. **Register the `ORCL.GoldenGate` resource type.**

   `# clresourcetype register ORCL.GoldenGate`

   register

   > Specifies that a new resource type is to be added.

   ORCL.GoldenGate

   > Specifies the name of the resource type to be added. This name is predefined for Oracle GoldenGate .

2. **Create an Oracle GoldenGate resource.**

   Note - You can choose to create a failover resource or a multiple master resource as follows.

   - **Create a failover Oracle GoldenGate resource.**

```
# clrs create -d-g db-rg \
-t ORCL.GoldenGate \
-p user=goldengate_owner \
-p Basepath=/goldengate \
-p port=7809 \
-p parameter_file=/goldengate/dirprm/mgr.prm \
-p database_resource=database-rs \
-p resource_dependencies=database-rs, logical-host \
-p resource_dependencies_offline_restart=gg-hsp-rs
 \
goldengate-rs
```

■   **Create a multiple master Oracle GoldenGate resource.**

```
# clrs create -d-g db-rg \
-t ORCL.GoldenGate \
-p user{node1}=goldengate_owner_1 \
-p user{node2}=goldengate_owner_2 \
-p Basepath{node1}=/goldengate_1 \
-p Basepath{node2}=/goldengate_2 \
-p port{node1}=7809 \
-p port{node2}=7909 \
-p parameter_file{node1}=/goldengate_1/dirprm/mgr.prm \
-p parameter_file{node2}=/goldengate_2/dirprm/mgr.prm \
-p database_resource{node1}=database1-rs \
-p database_resource{node2}=database2-rs \
-p resource_dependencies{LOCAL_NODE}=database1-rs, database2-rs
 \
goldengate-rs
```

-g *db-rg*

Specifies the resource group to which the resource is to be added. The resource group is confirmed either as a failover resource group or as a multiple master resource group.

-t ORCL.GoldenGate

Specifies that the resource is an instance of the ORCL.GoldenGate resource type.

-p user=*goldengate_owner*

Specifies the Oracle GoldenGate software owner.

-p Basepath=*/goldengate*

Specifies the path of the Oracle GoldenGate installation.

-p port=*7809*

Specifies the port on which the Oracle GoldenGate manager listens.

-p parameter_file=*/goldengate/dirprm/mgr.prm*

> Specifies the absolute path to the manager's parameter file.

-p database_resource=*database_rs*

> Specifies the name of the database resource.

-p resource_dependencies=*database-rs, logical-host*

> Specifies the list of dependencies.

-p resource_dependencies_offline_restart=*gg-hsp-rs*

> Specifies the list of offline restart dependencies.

`goldengate-rs`

> Specifies the resource name of the Oracle GoldenGate resource.

3. **Verify that the default values for the GoldenGate extension properties are acceptable.**

   Refer to "`ORCL.GoldenGate` Extension Properties" on page 43.

4. **Enable the GoldenGate resource.**

   `#  `**`clresource enable`** *goldengate-rs*

   `enable`

   > Specifies that the specified resource is to be enabled.

   goldengate-rs

   > Specifies the name of the resource to be enabled.

Using Alternate Project Identification

You can create a specific project, that is, an alternate project identification (ID), for the application. Set either `RG_project_name` or `Resource_project_name` or both, for the resource group and/or the resource, so that the application is started up under the specified project. Refer to the Resource Group Manager documentation for how to set these system properties.

See Oracle Solaris Cluster Administration and Application Development section in "Oracle Solaris Cluster Concepts Guide ".

Tuning the HA for Oracle GoldenGate Fault Monitors

Fault monitoring for the HA for Oracle GoldenGate data service is provided by the fault monitor for the Oracle GoldenGate instance.

Each fault monitor is contained in a resource whose resource type is shown in the following table.

TABLE 1-3   Resource Type for the Fault Monitors of HA for Oracle GoldenGate

Component	Resource Type
Oracle GoldenGate	`ORCL.GoldenGate`

System properties and extension properties of the resource types control the behavior of the fault monitors. The default values of these properties determine the preset behavior of the fault monitors. The preset behavior should be suitable for most Oracle Solaris Cluster installations. Therefore, you should tune the fault monitors *only* if you need to modify this preset behavior.

Tuning these fault monitors involves the following tasks:

- Setting the interval between fault monitor probes
- Setting the timeout for fault monitor probes
- Defining the criteria for persistent faults
- Specifying the failover behavior of a resource

Perform these tasks when you register and configure HA for Oracle GoldenGate, as described in "Registering and Configuring HA for Oracle GoldenGate" on page 20.

For detailed information about these tasks, see "Tuning Fault Monitors for Oracle Solaris Cluster Data Services" in "Oracle Solaris Cluster Data Services Planning and Administration Guide ".

Operation of the Fault Monitor for the Oracle GoldenGate Resource Type

To determine whether the Oracle GoldenGate process is operating correctly, the fault monitor for the Oracle GoldenGate resource type probes these resources periodically.

The probe checks if the manager process is in the process table. If this is true, then the probe opens a socket to the manager port. If not, then the probe declares the resource as faulted.

If the manager is OK, the probe checks if there are extract or replicat processes running. If not, it checks the status of these processes with the following command:

```
# su - ggowner -c "echo 'info all' | base directory/ggsci"
```

If the status of every configured extract or replicat process is abended, the probe returns with the value of the property DEGRADED_CODE.

The number of tolerated consecutive DEGRADED_CODE probe return within the retry_interval seconds is calculated by the number 100 divided by DEGRADED_CODE. If this number is greater than the retry_interval divided by thorough_probe_interval, then the probe return values of DEGRADED_CODE will be tolerated forever.

Tuning the Process Monitoring

The HA for Oracle GoldenGate agent can be tuned to run with process monitoring or without process monitoring.

You can tune the process monitoring by setting the pmf_managed property to either true or false.

The benefit of running with process monitoring is that if the whole process tree collapses, a resource action happens immediately. The benefit of running without process monitoring is that Oracle Solaris Cluster can take control of an Oracle GoldenGate instance that is already running.

Tuning the Failover Behavior

The default Failover_mode mode for an Oracle GoldenGate resource is soft. This default mode indicates that the resource can trigger a failover to another cluster node if probe errors happen too often and will translate into an outage for the underlying database as well. If this is not the desired behavior, consider setting the Failover_mode of the Oracle GoldenGate resource to log_only or resource_restart.

For more generic information of fault monitors, see "Tuning Fault Monitors for Oracle Solaris Cluster Data Services" in "Oracle Solaris Cluster Data Services Planning and Administration Guide ".

The Oracle GoldenGate agent inherits the genuine behavior of the ORCL.gds resource type. For more information about the ORCL.gds resource type, see "Oracle Solaris Cluster Generic Data Service (GDS) Guide ".

Verifying the HA for Oracle GoldenGate Installation and Configuration

You have installed, registered, and configured the data service for Oracle GoldenGate. Now verify that the data service makes the Oracle GoldenGate highly available by verifying the operation of the fault monitor for the Oracle GoldenGate instance.

See "Tuning the HA for Oracle GoldenGate Fault Monitors" on page 24 for a description of the fault monitors.

▼ How to Verify the Operation of the Fault Monitor for the GoldenGate Process

Perform this procedure on each set of two nodes where the Oracle GoldenGate process can run.

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

2. **Ensure that the resource group hosting the Oracle GoldenGate resource is offline on all nodes.**

3. **Bring online the resource group to which the Oracle GoldenGate belongs.**

   `# clresourcegroup online -n` *node db-rg*

   `-n` *node*

   > Specifies the name of the node on which the resource group is to be brought online. This node is the node that you have just logged in to.

   db-rg

   > Specifies the name of the resource group to be brought online. This group is the resource group to which the Oracle GoldenGate resource belongs.

4. **Terminate the Oracle GoldenGate process.**

   `# su -` *ggowner* `-c "echo 'stop er *' |` *basepath*`/ggsci"`
   `# su -` *ggowner* `-c "echo 'stop mgr \!' |` *basepath*`/ggsci"`

5. **Kill all remaining Oracle GoldenGate processes.**

6. **Confirm that the Oracle GoldenGate process gets restarted.**

   Run the `clresource status` command to confirm that the Oracle GoldenGate resource is online on the first node or zone.

7. **Failover the resource group hosting the Oracle GoldenGate instance to another node if the resource group is configured as a failover resource group. If the resource group is configured as a multiple master resource group, skip this step.**

    `# clresourcegroup switch -n` *node* *db-rg*

    `-n` *node*

    > Specifies the name of the node or the zone on which the resource group is to be brought online.

    db-rg

    > Specifies the name of the resource group to be brought online. This group is the resource group to which the Oracle GoldenGate resource belongs.

Generic Test Case

For all Oracle GoldenGate resources, kill the manager process. The resource gets restarted, or a failover happens depending on the configuration and the fault history.

For all Oracle GoldenGate resources with replicat, kill the extract and replicat processes until the manger gives up on restarting them. The resource status will transfer to `Degraded`.

For all Oracle GoldenGate resources with replicats configured and extracts running on the remote site, kill the server processes. The remote Oracle GoldenGate resource status transfers to `Degraded`, if the remote configuration does not contain replicats as well. The status transfer happens only if the cluster probe comes before a potential Oracle GoldenGate restart of the extract processes.

For deployment examples of HA for Oracle GoldenGate, see the following chapters:

- Chapter 2, "Deployment Example: Configuring the Oracle GoldenGate Agent in a Failover Configuration"
- Chapter 3, "Deployment Example: Configuring the Oracle GoldenGate Agent in a Multiple Master Configuration"

2

Deployment Example: Configuring the Oracle GoldenGate Agent in a Failover Configuration

This chapter provides a deployment example of how to install two Oracle GoldenGate instances on two, two-node clusters in a failover configuration.

Example Overview

The first cluster consists of the nodes psoma1 and psoma2 and the second cluster consists of the nodes vzsoma1a and vzsoma2a. The replication is configured from the soma cluster to the vzsoma cluster. For the Oracle GoldenGate agent, the clusters can either be global zones or zone clusters. In this example, the underlying database is MySQL.

Software and Cluster Configuration

The software configuration is as follows:

- MySQL 5.6.15
- GoldenGate 12.1.2

The following logical hosts are used.

- soma-3 - Source database on Cluster 1
- soma-1 - Target database on Cluster 2

Failover resource groups are as follows:

- db1-rg - Failover resource group on Cluster 1
- db2-rg - Failover resource group on Cluster 2

Assumptions

- The MySQL databases are installed and configured as `mys-rs` in the resource groups `db1-rg` and `db2-rg`.
- The MySQL owner has the default group `dba` and the MySQL data has group permissions for `dba`.
- The logical hosts `soma-1` and `soma-3` are also used by MySQL. Hence, the are also configured in the resource groups, `db1-rg` and `db2-rg`.
- The Oracle GoldenGate resources are configured in the same resource groups.
- On Cluster 1, the MySQL database is installed in the `/mys2/data` directory. On Cluster 2, the MySQL database is installed in the `/mys1/data` directory.
- On each cluster, a zpool is configured for Oracle GoldenGate. Zpool `gg2` is on Cluster 1, and zpool `gg1` is on Cluster 2.

Database Preparation

Use the following instructions to prepare the database:

- On both clusters, make sure the following lines are in the MySQL `my.cnf` configuration file. On Cluster 1:

```
# GoldenGate requirements
log-bin=/mys2/data/logs/bin-log
log-bin-index=/mys2/data/logs/bin-log.index
max_binlog_size=4096
binlog-format=ROW
#  GoldenGate requirements end
```

On Cluster 2

```
# GoldenGate requirements
log-bin=/mys1/data/logs/bin-log
log-bin-index=/mys1/data/logs/bin-log.index
max_binlog_size=4096
binlog-format=ROW
#  GoldenGate requirements end
```

- In both clusters, set the socket to the following value in the `my.cnf` file:

```
socket=/tmp/mysql.sock
```

As of the time of creating this deployment example, `/tmp/mysql.sock` must be used as the socket for the MySQL database. Check your Oracle GoldenGate documentation for the latest information.

- Restart the database resource as shown:

```
psoma1 # clrs disable mys-rs
psoma1 # clrs enable mys-rs
vzsoma1a # clrs disable mys-rs
vzsoma1a # clrs enable mys-rs
```

- On both clusters, create a database to be replicated and an User for Oracle GoldenGate.

  For for the sake of simplicity, in a lab install make the `gg` user as the administrator.

```
root@psoma1:~# /usr/local/mysql/bin/mysql -S /tmp/mysql.sock -uroot -proot
mysql> use mysql
mysql> GRANT ALL ON *.* TO 'gg'@'localhost' IDENTIFIED BY 'gg';
mysql> GRANT ALL ON *.* TO 'gg'@'psoma1' IDENTIFIED BY 'gg';
mysql> GRANT ALL ON *.* TO 'gg'@'psoma2' IDENTIFIED BY 'gg';
mysql> GRANT ALL ON *.* TO 'gg'@'soma-3' IDENTIFIED BY 'gg';
mysql> create database golden;
mysql> exit

root@psoma1:~# /usr/local/mysql/bin/mysql -h soma-3 -ugg -pgg
mysql> use golden
mysql> create table tab1 (row1 int not null, row2 char(64));
mysql> create unique index ind1 on tab1(row1);

root@vzsoma1a:~# /usr/local/mysql/bin/mysql -S /tmp/mysql.sock -uroot -proot
mysql> use mysql
mysql> GRANT ALL ON *.* TO 'gg'@'localhost' IDENTIFIED BY 'gg';
mysql> GRANT ALL ON *.* TO 'gg'@'vzsoma2a' IDENTIFIED BY 'gg';
mysql> GRANT ALL ON *.* TO 'gg'@'vzsoma2a' IDENTIFIED BY 'gg';
mysql> GRANT ALL ON *.* TO 'gg'@'soma-1' IDENTIFIED BY 'gg';
mysql> create database golden;
mysql> exit

root@vzsoma1a:~# /usr/local/mysql/bin/mysql -h soma-3 -ugg -pgg
mysql> use golden
mysql> create table tab1 (row1 int not null, row2 char(64));
mysql> create unique index ind1 on tab1(row1);
mysql> exit
```

Host Preparation for Oracle GoldenGate

Follow the instructions in this section to prepare the host for Oracle GoldenGate.

1. Configure the storage for Oracle GoldenGate.

    On Cluster 1:

    ```
    psomal # clrs create -t SUNW.HAStoragePlus -p zpools=/gg2 gg-hsp-rs
    psoma1 # clrs enable gg-hsp-rs
    ```

    On Cluster 2:

    ```
    vzsoma1a # clrs create -t SUNW.HAStoragePlus -p zpools=/gg2 gg-hsp-rs
    vzsoma1a # clrs enable gg-hsp-rs
    ```

2. Add the users.

    ```
    root@psoma1:~# useradd -u 2000 -g dba -m -s /usr/bin/bash -d /export/home/gg gg
    root@psoma2:~# useradd -u 2000 -g dba -m -s /usr/bin/bash -d /export/home/gg gg
    root@vzsoma1a:~# useradd -u 2000 -g dba -m -s /usr/bin/bash -d /export/home/gg gg
    root@vzsoma1a:~# useradd -u 2000 -g dba -m -s /usr/bin/bash -d /export/home/gg gg
    ```

3. Add the LD_LIBRARY_PATH to the environment of gg on all nodes.

    In this example, the /gg1 or the /gg2 entry is the only LD_LIBRARY_PATH entry in the .profile file.

    On psoma1 and psoma2:

    ```
    LD_LIBRARY_PATH=/gg2
    export LD_LIBRARY_PATH
    ```

    On vzsoma1a and vzsoma2a:

    ```
    LD_LIBRARY_PATH=/gg1
    export LD_LIBRARY_PATH
    ```

4. Set the permissions on the shared file system.

    On Cluster 1:

    ```
    root@psoma1:~# chown -R gg:dba/gg2
    ```

    On Cluster 2:

    ```
    root@vzsoma1a:~# chown -R gg:dba/gg1
    ```

5. Install the Oracle GoldenGate software.

    As user gg, untar the Oracle GoldenGate binaries in /gg1 on vzsoma1a and /gg2 on psoma1, assuming these are the nodes which own the shared storage.

    ```
    gg@psoma1:/gg2$ tar xvf /temp/ggs_Solaris_sparc_MySQL_64bit.tar
    ```

```
gg@vzsoma1a:/gg1$ tar xvf /temp/ggs_Solaris_sparc_MySQL_64bit.tar
```

6. Configure the Oracle GoldenGate instances.

   As user gg perform the following tasks:

   a. Create the Oracle GoldenGate instance subdirectories.

   ```
   gg@vzsoma1a:/gg1$ ./ggsci
   GGSCI(vzsoma1a) 1> create subdirs
   gg@psoma1:/gg2$ ./ggsci
   GGSCI(psoma1) 1> create subdirs
   ```

   b. Configure the manager on Cluster 1.

   ```
   GGSCI(psoma1) 3> edit params mgr
   PORT 7809
   DYNAMICPORTLIST  7810-7820
   AUTORESTART  ER *, RETRIES 4, WAITMINUTES 4
   STARTUPVALIDATIONDELAY  5
   PURGEOLDEXTRACTS  /gg2/dirdat/mm*, USECHECKPOINTS, MINKEEPHOURS 2
   GGSCI (psoma1) 3>
   ```

   c. Configure the manager on Cluster 2.

   ```
   GGSCI(vzsoma1a) 3> edit params mgr
   PORT 7909
   DYNAMICPORTLIST  7910-7920
   AUTORESTART  ER *, RETRIES 4, WAITMINUTES 4
   STARTUPVALIDATIONDELAY  5
   PURGEOLDEXTRACTS  /gg1/dirdat/mm*, USECHECKPOINTS, MINKEEPHOURS 2
   GGSCI(vzsoma1a) 3>
   ```

   d. Configure the extract on Cluster 1.

   ```
   GGSCI(psoma1) 9> edit param mysqlext
   EXTRACT  mysqlext
   DBOptions  Host soma-3
   SourceDB  golden UserID gg Password gg
   TranLogOptions  AltLogDest /mys2/data/logs/bin-log.index
   DISCARDFILE  /gg2/mysqlext.dsc, PURGE
   RMTHOST  soma-1, MGRPORT 7909
   RMTTRAIL  /gg1/dirdat/XX
   TABLE  golden.*;
   ```

   e. Add the extract on Cluster 1.

   ```
   GGSCI(psoma1) > Add Extract MySQLEXT TranLog Begin now
   ```

   f. Add the remote `trail` on Cluster 1.

   ```
   GGSCI(psoma1) >Add RmtTrail /gg1/dirdat/XX extract MySQLEXT
   ```

g. Configure the replicat on Cluster 2.

```
GGSCI(vzsoma1a) 11> edit param mysqlrep
REPLICAT  mysqlrep
DBOptions  Host soma-1
TargetDB  golden USERID gg PASSWORD gg
DISCARDFILE  /gg1/mysqlrep.dsc, PURGE
REPERROR  DEFAULT, ABEND
AssumeTargetDefsMap  golden.*, Target golden.*;
```

h. Add the replicat on Cluster 2.

```
GGSCI(vzsoma1a) 11> Add Replicat MySQLREP \
exttrail /gg1/dirdat/XX  nodbcheckpoint
```

i. Define the Ipv6 policy on both the clusters.

```
GGSCI(psoma1) >edit params ./GLOBALS
USEIPV6
GGSCI(vzsoma1a) >edit params ./GLOBALS
USEIPV6
```

j. Start the manager on both the clusters.

```
GGSCI(psoma1) > start mgr
GGSCI (vzsoma1a) >start mgr
```

k. Start the extract and replicat processes on both the clusters.

```
GGSCI(psoma1) > start er *
GGSCI(vzsoma1a) > start er *
```

l. Check if all the processes are running on both the clusters.

```
GGSCI(psoma1) > info all
GGSCI(vzsoma1a) > info all
```

m. Stop the extract and replicat processes on both the clusters

```
GGSCI(psoma1) > stop er *
GGSCI(vzsoma1a) > stop er *
```

n. Stop the manager process on both the clusters.

```
GGSCI(psoma1) > stop mgr !
GGSCI(psoma1) > exit
GGSCI(vzsoma1a) > stop mgr !
GGSCI(vzsoma1a) > exit
```

o. Kill the server process on the cluster with replicats configured.

```
vzsoma1a # pkill -f server
```

7. Create the Oracle GoldenGate resources.

```
psoma1 # clrt register ORCL.GoldenGate
psoma1 # clrs create -d -g db1-rg \
-t GoldenGate \
-p user=gg \
-p Basepath=/gg2 \
-p port=7809 \
-p parameter_file=/gg2/dirprm/mgr.prm \
-p database_resource=mys-rs \
-p resource_dependencies=mys-rs,soma-3 \
-p resource_dependencies_offline_restart=gg-hsp-rs \
gg2-rs
psoma1 # clrs enable gg2-rs


vzsoma1a # clrt register ORCL.GoldenGate
vzsoma1a # clrs create -d -g db2-rg \
-t GoldenGate \
-p user=gg \
-p Basepath=/gg1 \
-p port=7909 \
-p parameter_file=/gg1/dirprm/mgr.prm \
-p database_resource=mys-rs \
-p resource_dependencies=mys-rs,soma-1 \
-p resource_dependencies_offline_restart=gg-hsp-rs \
gg1-rs
vzsoma1a # clrs enable gg1-rs
```

3

Deployment Example: Configuring the Oracle GoldenGate Agent in a Multiple Master Configuration

This chapter provides a deployment example of how to install two Oracle GoldenGate instances on a single, two-node cluster in a multiple master configuration.

Example Overview

The cluster consists of the nodes `psoma1` and `psoma2`. The replication is configured from a different cluster to the `soma` cluster. The replication source must be configured from another cluster, which is not shown in this example. The nodes `psoma1` and `psoma2` contain the replication target databases.

For the Oracle GoldenGate agent, the clusters can be either global zones or zone clusters. In this example, the underlying database is MySQL.

Software and Cluster Configuration

The software configuration is as follows:

- MySQL 5.6.15
- GoldenGate 12.1.2

The following logical hosts are used.

- `psoma-1` - Target database on Node 1
- `psoma-2` - Target database on Node 2

Multiple master resource group on Cluster 1, is as follows:

- `db1-rg`

Assumptions

- The MySQL databases are installed and configured as multiple master resource `mys-rs` in the resource group `db1-rg`.
- The MySQL owner has the default group `dba` and the MySQL data has group permissions for `dba`.
- On both nodes, the MySQL database is installed in the `/mys/data` directory.
- On each node, the local file system is configured and mounted under `/gg` for Oracle GoldenGate.

Database Preparation

Use the following instructions to prepare the database:

- On both nodes, make sure the following lines are in the MySQL `my.cnf` configuration file.

  On `psoma1`:

  ```
  # GoldenGate requirements
  log-bin=/mys/data/logs/bin-log
  log-bin-index=/mys/data/logs/bin-log.index
  max_binlog_size=4096
  binlog-format=ROW
  #  GoldenGate requirements end
  ```

  On `psoma2`

  ```
  # GoldenGate requirements
  log-bin=/mys/data/logs/bin-log
  log-bin-index=/mys/data/logs/bin-log.index
  max_binlog_size=4096
  binlog-format=ROW
  #  GoldenGate requirements end
  ```

- In both nodes, set the socket to the following value in the `my.cnf` file:

  ```
  socket=/tmp/mysql.sock
  ```

  As of the time of creating this deployment example, `/tmp/mysql.sock` must be used as the socket for the MySQL database. Check your Oracle GoldenGate document for the latest information.

- Restart the database resource as shown:

  ```
  psoma1 # clrs disable mys-rs
  ```

```
psoma1 # clrs enable mys-rs
```

- On both nodes, create a database to be replicated and an User for Oracle GoldenGate.

  For the sake of simplicity, in a lab install, make the gg user as the administrator.

```
root@psoma1:~# /usr/local/mysql/bin/mysql -S /tmp/mysql.sock -uroot -proot
mysql> use mysql
mysql> GRANT ALL ON *.* TO 'gg'@'localhost' IDENTIFIED BY 'gg';
mysql> GRANT ALL ON *.* TO 'gg'@'psoma1' IDENTIFIED BY 'gg';
mysql> GRANT ALL ON *.* TO 'gg'@'psoma2' IDENTIFIED BY 'gg';
mysql> create database golden;
mysql> exit

root@psoma1:~# /usr/local/mysql/bin/mysql -h /psoma1 -ugg -pgg
mysql> use golden
mysql> create table tab1 (row1 int not null, row2 char(64));
mysql> create unique index ind1 on tab1(row1);

root@psoma2:~# /usr/local/mysql/bin/mysql -S /tmp/mysql.sock -uroot -proot
mysql> use mysql
mysql> GRANT ALL ON *.* TO 'gg'@'localhost' IDENTIFIED BY 'gg';
mysql> GRANT ALL ON *.* TO 'gg'@'psoma1' IDENTIFIED BY 'gg';
mysql> GRANT ALL ON *.* TO 'gg'@'psoma2' IDENTIFIED BY 'gg';
mysql> create database golden;
mysql> exit

root@psoma2:~# /usr/local/mysql/bin/mysql -h psoma2 -ugg -pgg
mysql> use golden
mysql> create table tab1 (row1 int not null, row2 char(64));
mysql> create unique index ind1 on tab1(row1);
mysql> exit
```

Host Preparation for Oracle GoldenGate

Follow the instructions in this section to prepare the host for Oracle GoldenGate.

1. Add the users.

   Create the user gg on all nodes. Ensure that the gg user is in the same group as the MySQL user mys, because the Oracle GoldenGate user must have access to the MySQL binary log files.

```
root@psoma1:~# useradd -u 2000 -g dba -m -s /usr/bin/bash -d /export/home/gg gg
root@psoma2:~# useradd -u 2000 -g dba -m -s /usr/bin/bash -d /export/home/gg gg
```

2. Add the LD_LIBRARY_PATH to the environment of gg on all nodes.

In this example, the `/gg` entry is the only `LD_LIBRARY_PATH` entry in the `.profile` file.

On `psoma1` and `psoma2`:

```
LD_LIBRARY_PATH=/gg
export LD_LIBRARY_PATH
```

3. Set the permissions on the Oracle GoldenGate file system.

   On Node 1:

   ```
   root@psoma1:~# chown -R gg:dba/gg
   ```

   On Node 2:

   ```
   root@psoma2:~# chown -R gg:dba /gg
   ```

4. Install the Oracle GoldenGate software.

   As user `gg`, untar the Oracle GoldenGate binaries in `/gg` on `psoma1` and `psoma2`.

   ```
   gg@psoma1:/gg$ tar xvf /temp/ggs_Solaris_sparc_MySQL_64bit.tar
   gg@psoma2:/gg$ tar xvf /temp/ggs_Solaris_sparc_MySQL_64bit.tar
   ```

5. Configure the Oracle GoldenGate instances.

   As user `gg` perform the following tasks:

   a. Create the Oracle GoldenGate instance subdirectories.

   ```
   gg@psoma1:/gg$  ./ggsci
   GGSCI (psoma1) 1> create subdirs
   gg@psoma2:/gg$  ./ggsci
   GGSCI (psoma2) 1> create subdirs
   ```

   b. Configure the manager on Node 1.

   ```
   GGSCI(psoma1) 3> edit params mgr
   PORT 7809
   DYNAMICPORTLIST  7810-7820
   AUTORESTART  ER *, RETRIES 4, WAITMINUTES 4
   STARTUPVALIDATIONDELAY 5
   PURGEOLDEXTRACTS  /gg/dirdat/mm*, USECHECKPOINTS, MINKEEPHOURS 2
   GGSCI (psoma1) 3>
   ```

   c. Configure the manager on Node 2.

   ```
   GGSCI(psoma2) 3> edit params mgr
   PORT 7909
   DYNAMICPORTLIST  7910-7920
   AUTORESTART  ER *, RETRIES 4, WAITMINUTES 4
   STARTUPVALIDATIONDELAY 5
   PURGEOLDEXTRACTS  /gg/dirdat/mm*, USECHECKPOINTS, MINKEEPHOURS 2
   GGSCI(psoma2) 3>
   ```

d. Configure the replicat on Node 1.

```
GGSCI(psoma1) 11> edit param mysqlrep
REPLICAT  mysqlrep
DBOptions  Host psoma1
TargetDB  golden USERID gg PASSWORD gg
DISCARDFILE  /gg1/mysqlrep.dsc, PURGE
REPERROR  DEFAULT, ABEND
AssumeTargetDefsMap
Map golden.*, Target golden.*;
GGSCI(psoma1) 3>
```

e. Add the replicat on Node 1.

```
GGSCI(psoma1) 11> Add Replicat MySQLREP exttrail /gg1/dirdat/XX  nodbcheckpoint
```

f. Configure the replicat on Node 2.

```
GGSCI(psoma2) 11> edit param mysqlrep
REPLICAT  mysqlrep
DBOptions  Host psoma2
TargetDB  golden USERID gg PASSWORD gg
DISCARDFILE  /gg1/mysqlrep.dsc, PURGE
REPERROR  DEFAULT, ABEND
AssumeTargetDefsMap
Map golden.*, Target golden.*;
GGSCI(psoma2) 3>
```

g. Add the replicat on Node 2.

```
GGSCI(psoma2) 11> Add Replicat MySQLREP exttrail /gg1/dirdat/XX  nodbcheckpoint
```

h. Define the Ipv6 policy on both the nodes.

```
GGSCI(psoma1) >edit params ./GLOBALS
USEIPV6
GGSCI(psoma2) >edit params ./GLOBALS
USEIPV6
```

i. Start the manager on both the nodes.

```
GGSCI(psoma1) > start mgr
GGSCI(psoma2) >start mgr
```

j. Start the replicat processes on both the nodes.

```
GGSCI(psoma1) > start er *
GGSCI(psoma2) > start er *
```

k. Check if all the processes are running on both the nodes.

```
GGSCI(psoma1) > info all
```

```
GGSCI(psoma2) > info all
```

l. Stop the replicat processes on both the clusters

```
GGSCI(psoma1) > stop er *
GGSCI(psoma2) > stop er *
```

m. Stop the manager process on both the clusters.

```
GGSCI(psoma1) > stop mgr !
GGSCI(psoma1) > exit
GGSCI(psoma2) > stop mgr !
GGSCI(psoma2) > exit
```

n. Kill the server process on the nodes with replicats configured.

```
psoma1 # pkill -f server
psoma2 # pkill -f server
```

6. Create the Oracle GoldenGate resources.

```
psoma1 # clrt register ORCL.GoldenGate
psoma1 # clrs create -d -g db1-rg \
-t GoldenGate \
-p user{psoma1}=gg \
-p user{psoma2}=gg \
-p Basepath{psoma1}=/gg \
-p Basepath{psoma2}=/gg
-p port{psoma1}=7809 \
-p port{psoma2}=7909
-p parameter_file{psoma1}=/gg/dirprm/mgr.prm \
-p parameter_file{psoma1}=/gg/dirprm/mgr.prm \
-p database_resource{psoma1}=mys-rs \
-p database_resource{psoma2}=mys-rs \
-p resource_dependencies=mys-rs{LOCAL_NODE} \
gg1-rs
psoma1# clrs enable gg-rs
```

A

HA for Oracle GoldenGate Extension Properties

The ORCL.GoldenGate resource type inherits the algorithms and properties from the `ORCL.gds` resource type. Only the additional extension properties for the ORCL.GoldenGate resource type are described in this Appendix. For a description for all other properties, see "Oracle Solaris Cluster Generic Data Service (GDS) Guide ".

`ORCL.GoldenGate` Extension Properties

The additional extension properties of this resource type are as follows:

User

This property indicates the owner of a Oracle GoldenGate installation.

Data type	String
Default	None defined
Range	Not applicable
Tunable	When disabled

Basepath

This property indicates the Oracle GoldenGate installation directory.

Data type	String
Default	None defined
Range	Not applicable
Tunable	When disabled

Degraded_code

This property indicates the return code of the probes, if all of the extract or replicat processes are abended.

Data type	Integer
Default	10
Range	1–50
Tunable	When disabled

Port

This property indicates the port on which the manager responds.

Data type	Integer
Default	0
Range	0–65535
Tunable	When disabled

Parameter_file

This property indicates the parameter file for a manager process to run on the Oracle GoldenGate instance.

Data type	String
Default	None defined
Range	Not applicable
Tunable	When disabled

Database_resource

This property indicates the resource name which protects the database to be replicated by the Oracle GoldenGate instance.

Data type	String
Default	None defined
Range	Not applicable
Tunable	When disabled

Start_pattern

This property indicates a list of pattern describing the extract and the replicat names to be started by the Oracle GoldenGate agent.

Data type	String
Default	`"'er \\\*'"`

Note - Precede the asterisk by three backward slash (\) characters, and surround the entire string by single quotation marks (') as well as double quotation marks (").

Range	Not applicable
Tunable	When disabled

Index

B

basepath extension property
   ORCL.GoldenGate resource type, 43

D

database_resource extension property
   ORCL.GoldenGate resource type, 44
degraded_code extension property
   ORCL.GoldenGate resource type, 43

E

extension properties
   ORCL.GoldenGate resource type, 43
extract
   process, 9

F

fault monitors, 24

H

HA for Oracle GoldenGate
   deployment example, 29, 37
   failover configuration example, 29
   fault monitor, 24
   multiple master configuration example, 37

M

manager

process, 9

O

Oracle GoldenGate software, 9
ORCL.GoldenGate resource type
   extension properties, 43

P

parameter_file extension property
   ORCL.GoldenGate resource type, 44
port extension property
   ORCL.GoldenGate resource type, 44

R

replicat
   process, 9

S

server
   process, 9
start_pattern extension property
   ORCL.GoldenGate resource type, 44

U

user extension property
   ORCL.GoldenGate resource type, 43