

Oracle® AI Database

GDSCTL Command-Line Utility Reference



Release 26ai
G45181-01
March 2026

ORACLE®

Oracle AI Database GDSCTL Command-Line Utility Reference, Release 26ai
G45181-01

Copyright © 2025, 2026, Oracle and/or its affiliates.

Primary Authors: Virginia Beecher, Douglas Williams

Contributing Authors: Jim Womack

Contributors: Ravi Sharma

Contents

Preface

| | |
|-------------------|---|
| Audience | i |
| Related Documents | i |
| Conventions | i |

1 Using GDSCTL

| | |
|---|---|
| GDSCTL Operation | 1 |
| Starting GDSCTL | 1 |
| Running GDSCTL Commands Interactively | 1 |
| Running GDSCTL Batch Operations | 2 |
| GDSCTL Help Text | 2 |
| GDSCTL Connections | 2 |
| GDSCTL Catalog Connections | 3 |
| GDSCTL Global Service Manager Connections | 3 |

2 Global Data Services Control Utility (GDSCTL) Command Reference

| | |
|-------------------------------------|----|
| add brokerconfig | 7 |
| add cdb | 9 |
| add credential | 10 |
| add database | 10 |
| add file | 12 |
| add gdspool | 13 |
| add gsm | 14 |
| add invitednode (add invitedsubnet) | 16 |
| add region | 17 |
| add service | 18 |
| add shard | 25 |
| add shardgroup | 28 |
| add shardspace | 29 |
| alter move | 30 |
| alter task | 32 |
| config | 33 |

| | |
|---------------------|----|
| config backup | 34 |
| config cdb | 36 |
| config chunks | 37 |
| config credential | 38 |
| config database | 39 |
| config file | 40 |
| config gdspool | 41 |
| config gsm | 42 |
| config region | 43 |
| config sdb | 43 |
| config service | 44 |
| config shard | 46 |
| config shardgroup | 47 |
| config shardspace | 48 |
| config table family | 49 |
| config task | 49 |
| config vncr | 50 |
| configure | 51 |
| connect | 52 |
| copy ru | 53 |
| create gdscatalog | 54 |
| create restorepoint | 56 |
| create shardcatalog | 57 |
| databases | 61 |
| delete backup | 62 |
| delete catalog | 63 |
| deploy | 64 |
| disable backup | 65 |
| disable service | 65 |
| enable backup | 66 |
| enable service | 67 |
| exit | 68 |
| export catalog | 68 |
| help | 69 |
| import catalog | 69 |
| list backup | 70 |
| list restorepoint | 72 |
| modify catalog | 73 |
| modify cdb | 75 |
| modify credential | 76 |
| modify database | 77 |
| modify file | 78 |

| | |
|---|-----|
| modify gdspool | 78 |
| modify gsm | 79 |
| modify region | 81 |
| modify service | 81 |
| modify shard | 89 |
| modify shardgroup | 90 |
| modify shardspace | 91 |
| move chunk | 92 |
| move ru | 93 |
| quit | 93 |
| recover shard | 93 |
| relocate chunk | 94 |
| relocate service | 95 |
| remove brokerconfig | 96 |
| remove cdb | 97 |
| remove credential | 98 |
| remove database | 99 |
| remove file | 100 |
| remove gdspool | 100 |
| remove gsm | 101 |
| remove invitednode (remove invitedsubnet) | 102 |
| remove region | 102 |
| remove ru | 103 |
| remove service | 104 |
| remove shard | 104 |
| remove shardgroup | 105 |
| remove shardspace | 106 |
| restore backup | 106 |
| resume services | 109 |
| rman | 109 |
| run backup | 111 |
| services | 112 |
| set dataguard_property | 113 |
| set gsm | 114 |
| set inbound_connect_timeout | 115 |
| set log_status | 116 |
| set outbound_connect_timeout | 117 |
| set trace_level | 117 |
| set trc_level | 118 |
| show ddl | 119 |
| split chunk | 120 |
| sql | 121 |

| | |
|--|-----|
| start gsm | 122 |
| start observer | 122 |
| start ru | 123 |
| start service | 124 |
| status | 125 |
| status backup | 126 |
| status database | 128 |
| status gsm | 129 |
| status routing | 130 |
| status ru | 131 |
| status service | 134 |
| stop gsm | 135 |
| stop ru | 135 |
| stop service | 136 |
| suspend services | 138 |
| switchover ru | 138 |
| sync brokerconfig (synchronize brokerconfig) | 139 |
| sync database (synchronize database) | 140 |
| sync ru | 141 |
| sync schema (synchronize schema) | 141 |
| validate backup | 143 |
| validate catalog | 145 |
| validate | 146 |

3 GDSCTL Commands Used For Oracle Globally Distributed AI Database

4 GDSCTL Commands Used For Global Data Services

Index

Preface

This guide provides a command reference for the Global Data Services Control Utility (GDSCTL), which you can use for management of Oracle Globally Distributed AI Database and Oracle Global Data Services (GDS) to maximize the availability, performance, and scalability of your Oracle databases.

- [Audience](#)
- [Related Documents](#)
- [Conventions](#)

Audience

This document is intended for database administrators, database and system architects, and engineers who manage replicated database environments.

Related Documents

For more information, see the following documents:

- *Oracle AI Database Global Data Services Concepts and Administration Guide* GSMUG
- *Oracle AI Database Net Services Administrator's Guide*
- *Oracle AI Database Net Services Reference*
- *Oracle Globally Distributed AI Database*
- *Oracle True Cache User's Guide*

Conventions

The following text conventions are used in this document:

| Convention | Meaning |
|-----------------|--|
| boldface | Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary. |
| <i>italic</i> | Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values. |
| monospace | Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter. |

1

Using GDSCTL

Global Data Services GDSCTL commands are used for setting up and deploying a Global Data Services or an Oracle Globally Distributed AI Database configuration. Learn how to use the GDSCTL command-line tool in the following topics.

- [GDSCTL Operation](#)
Learn how to start GDSCTL, run commands, and get command help text.
- [GDSCTL Connections](#)
Some GDSCTL commands require a connection to the shard catalog, and for certain operations, GDSCTL must connect to a shard director.

GDSCTL Operation

Learn how to start GDSCTL, run commands, and get command help text.

- [Starting GDSCTL](#)
To start GDSCTL, enter `gdctl` at the operating system prompt.
- [Running GDSCTL Commands Interactively](#)
You can run GDSCTL commands interactively at either the operating system prompt or the GDSCTL command prompt.
- [Running GDSCTL Batch Operations](#)
You can gather all the GDSCTL commands in one file and run them as a batch with GDSCTL.
- [GDSCTL Help Text](#)
You can display help for GDSCTL and GDSCTL commands.

Starting GDSCTL

To start GDSCTL, enter `gdctl` at the operating system prompt.

```
$ gdctl
```

GDSCTL starts and displays the GDSCTL command prompt.

```
GDSCTL>
```

Running GDSCTL Commands Interactively

You can run GDSCTL commands interactively at either the operating system prompt or the GDSCTL command prompt.

Run a GDSCTL command at the system prompt.

```
$ gdctl add gsm -gsm gsm1 -catalog 127.0.0.1:1521:db1
```

Run a GDSCTL command at the GDSCTL command prompt.

```
GDSCTL> add gsm -gsm gsm1 -catalog 127.0.0.1:1521:db1
```

Both of these methods achieve the same result. The command syntax examples in this document use the GDSCTL command prompt.

Running GDSCTL Batch Operations

You can gather all the GDSCTL commands in one file and run them as a batch with GDSCTL.

The following command starts GDSCTL and runs the commands contained in the specified script file.

```
$ gdsctl @script_file_name
```

GDSCTL Help Text

You can display help for GDSCTL and GDSCTL commands.

The GDSCTL `HELP` command displays a summary of all GDSCTL commands.

```
GDSCTL> help
```

If you specify a command name after `HELP`, then the help text for that command is shown.

```
GDSCTL> help start gsm
```

You can also use the `-h` option with any GDSCTL command to show the help text for the specified command.

```
GDSCTL> start gsm -h
```

GDSCTL Connections

Some GDSCTL commands require a connection to the shard catalog, and for certain operations, GDSCTL must connect to a shard director.

- [GDSCTL Catalog Connections](#)
If you run GDSCTL commands that require a connection to the catalog database, then you must run the GDSCTL `CONNECT` command before the first command that requires the connection.
- [GDSCTL Global Service Manager Connections](#)
For certain operations, GDSCTL must connect to a global service manager (GSM) instance.

GDSCTL Catalog Connections

If you run GDSCTL commands that require a connection to the catalog database, then you must run the GDSCTL `CONNECT` command before the first command that requires the connection.

The `CONNECT` command only needs to be run once in a GDSCTL session.

GDSCTL uses Oracle Net Services to connect to the catalog database or another database in the GDS (or distributed database) configuration. For these connections you can run GDSCTL from any client or host that has the necessary network configuration.

Unless specified, GDSCTL resolves connect strings with the current name resolution methods (such as `TNSNAMES`).

The GDSCTL operations that require a connection to a catalog database are noted in the usage notes for each command.

See also: `connect`

GDSCTL Global Service Manager Connections

For certain operations, GDSCTL must connect to a global service manager (GSM) instance.

Unless specified, GDSCTL resolves connect strings with the current name resolution methods (such as `TNSNAMES`). However, to resolve the GSM name, GDSCTL queries the `gsm.ora` file.

To connect to a GSM, GDSCTL must be running on the same host as the GSM instance. When connecting to a GSM, GDSCTL looks for the `gsm.ora` file associated with the local GSM.

The following are the GDSCTL operations that require a connection to a GSM.

- `ADD GSM` adds a GSM instance.
- `START GSM` starts the GSM instance.
- `STOP GSM` stops the GSM instance.
- `MODIFY GSM` modifies the configuration parameters of the GSM instance.
- `STATUS GSM` returns the status of a GSM instance.
- `SET INBOUND_CONNECT_LEVEL` sets the `INBOUND_CONNECT_LEVEL` listener parameter.
- `SET TRACE_LEVEL` sets the trace level for the listener associated with the specified GSM instance.
- `SET OUTBOUND_CONNECT_LEVEL` sets the timeout value for the outbound connections for the listener associated with a specific GSM instance.
- `SET LOG_LEVEL` sets the log level for the listener associated with a specific GSM instance.

2

Global Data Services Control Utility (GDSCTL) Command Reference

This appendix includes a complete reference of the Global Data Services utility (GDSCTL) commands for use with a Global Data Services or Oracle Globally Distributed AI Database configuration.

- [add brokerconfig](#)
Adds an Oracle Data Guard broker configuration to a Global Data Services pool.
- [add cdb](#)
Add a cdb to the shard catalog.
- [add credential](#)
Adds a credential which can be used by the remote scheduler agent to execute shard jobs.
- [add database](#)
Adds databases to a Global Data Services region and Global Data Services pool.
- [add file](#)
Adds the contents of a file to the catalog which can be used by subsequent GDSCTL commands.
- [add gds pool](#)
Adds a Global Data Services pool to the Global Data Services framework.
- [add gsm](#)
Adds a global service manager to the Global Data Services framework.
- [add invitednode \(add invitedsubnet\)](#)
Adds host address or subnet information to the valid node checking for registration (VNCR) list in the catalog, before starting the first global service manager, by establishing a direct connection to the Global Data Services catalog database.
- [add region](#)
Adds a region to a Global Data Services framework or an Oracle Globally Distributed AI Database configuration.
- [add service](#)
Adds a global service to a Global Data Services pool.
- [add shard](#)
Add a shard to the shard catalog.
- [add shardgroup](#)
Add a shardgroup to a shardspace.
- [add shardspace](#)
Add a shardspace to the shard catalog.
- [alter move](#)
Suspend, resume, or cancel scheduled chunk move operations on a sharded database.
- [alter task](#)
Suspend, resume, or cancel scheduled chunk or replication unit management operations on a sharded database.

- [config](#)
Displays the configuration data for all components defined for the configuration.
- [config backup](#)
Configure Sharded Database (SDB) Backup
- [config cdb](#)
Displays properties of a specified CDB.
- [config chunks](#)
Displays properties of a specified chunk.
- [config credential](#)
Displays remote credentials currently available for shard jobs.
- [config database](#)
Displays the static configuration data stored in the catalog for the specified database.
- [config file](#)
Displays file objects currently available that can be specified in GDSCTL commands.
- [config gds pool](#)
Displays the static configuration data that is stored in the catalog for the specified database pool.
- [config gsm](#)
Displays the static configuration data stored in the catalog for the specified global service manager.
- [config region](#)
Displays the static configuration data for the specified region.
- [config sdb](#)
Displays the static configuration data stored in the catalog for the sharded database.
- [config service](#)
Displays the static configuration data stored in the Global Data Services catalog for the specified services that are located in a database pool.
- [config shard](#)
Displays properties of a specified shard.
- [config shardgroup](#)
Displays properties of a specified shardgroup.
- [config shardspace](#)
Displays properties of a specified shardspace.
- [config table family](#)
Displays information about all table families in the sharded database.
- [config task](#)
Display chunk or replication unit management tasks and their statuses.
- [config vncr](#)
Displays the static configuration data stored in the catalog for valid node checking for registration (VNCR).
- [configure](#)
Sets the GDSCTL parameters.
- [connect](#)
Specifies the credentials to administer a global service management environment. Credentials must be specified to perform certain operations using GDSCTL.

- [copy ru](#)
To instantiate or repair a follower member of a replication unit you can copy another follower in that replication unit from another shard.
- [create gdscatalog](#)
Creates a Global Data Services catalog for global service management in a specific database.
- [create restorepoint](#)
Create Global Restore Points.
- [create shardcatalog](#)
Creates a shard catalog for the sharded database.
- [databases](#)
Displays the status of all databases.
- [delete backup](#)
Deletes sharded database (SDB) backups identified with specific tags from the recovery repository.
- [delete catalog](#)
Deletes the specified catalog.
- [deploy](#)
Deploys the sharded database.
- [disable backup](#)
Disable Sharded Database (SDB) Backup Jobs.
- [disable service](#)
Disables specified global services.
- [enable backup](#)
Enable Sharded Database (SDB) Backup Jobs.
- [enable service](#)
Enables the specified global services.
- [exit](#)
Quit GDSCTL utility.
- [export catalog](#)
Saves the current catalog configuration to a local file.
- [help](#)
Provides a list of the GDSCTL commands supported in the current release. When followed by a command name, it returns the help page associated with the command.
- [import catalog](#)
Restores the catalog configuration from the specified file, created using export catalog command.
- [list backup](#)
List Sharded Database (SDB) Backups
- [list restorepoint](#)
List Global Restore Points.
- [modify catalog](#)
Modifies the properties of the GDS catalog or shard catalog.
- [modify cdb](#)
Modify cdb attributes.

- [modify credential](#)
Modifies an existing credential which will be used by the remote Scheduler agent to execute shard jobs.
- [modify database](#)
Modifies the configuration parameters of the databases in a GDS pool, such as region, connect identifier, global service manager password, SCAN address, and ONS port.
- [modify file](#)
Updates the contents of a file in the catalog which can be used by subsequent GDSCTL commands.
- [modify gdspool](#)
Modifies the configuration parameters of GDS pools.
- [modify gsm](#)
Modifies the configuration parameters of the global service manager. The changes take effect after the global service manager is restarted.
- [modify region](#)
Modifies the configuration parameters for a region.
- [modify service](#)
Modifies the service attributes.
- [modify shard](#)
Modify shard attributes.
- [modify shardgroup](#)
Modify shardgroup attributes.
- [modify shardspace](#)
Modify shardspace parameters.
- [move chunk](#)
Moves a listed set of chunks from one shard to another shard or multiple shards.
- [move ru](#)
Move a member replica of a replication unit from one shard to another.
- [quit](#)
Quit GDSCTL utility.
- [recover shard](#)
Executes all DDL statements on the specified shard (database), starting from the one that was previously executed with errors. The command is intended to perform all skipped DDL changes after database administrator fixes shard issues.
- [relocate chunk](#)
This command moves a list of chunks in all the replicas of the specified source RU to all the replicas in the target ru.
- [relocate service](#)
Stops a service on one database and starts the service on a different database.
- [remove brokerconfig](#)
Removes an Oracle Data Guard broker configuration from a GDS pool.
- [remove cdb](#)
Removes one or more CDBs from the shard catalog, but does not destroy it.
- [remove credential](#)
Removes an existing credential.
- [remove database](#)
Removes databases from a GDS pool.

- [remove file](#)
Removes an existing file object from the catalog.
- [remove gdspool](#)
Removes a GDS pool from the current configuration.
- [remove gsm](#)
Removes a global service manager from the configuration.
- [remove invitednode \(remove invitedsubnet\)](#)
Remove host address information from the valid node checking for registration (VNCR) list in the Global Data Services catalog. This command removes either the specified invitednode or all invitednodes that correspond to an alias.
- [remove region](#)
Removes the specified regions from the global service management framework.
- [remove ru](#)
Remove empty replication unit from a sharded database configuration.
- [remove service](#)
Removes a service from a database pool.
- [remove shard](#)
Removes one or more shards from the sharded database.
- [remove shardgroup](#)
Removes a shardgroup from the shard catalog.
- [remove shardspace](#)
Removes a shardspace from the shard catalog.
- [restore backup](#)
The restore backup command is used to restore a sharded database to a specific global restore point.
- [resume services](#)
Resumes global service activity and traffic routing to the database, previously blocked by `SUSPEND SERVICES` command.
- [rman](#)
Allow users to submit RMAN commands to a list of shards for execution.
- [run backup](#)
Run Sharded Database (SDB) Backup Jobs.
- [services](#)
Retrieves information about the services that are registered with the specified global service manager.
- [set dataguard_property](#)
Dynamically updates the value of a specified property of a broker configuration or database.
- [set gsm](#)
Sets the global service manager for the current session.
- [set inbound_connect_timeout](#)
Sets the `INBOUND_CONNECT_TIMEOUT` listener parameter.
- [set log_status](#)
Sets the `LOG_STATUS` listener parameter.
- [set outbound_connect_timeout](#)
Sets the `OUTBOUND_CONNECT_TIMEOUT` listener parameter.

- [set trace_level](#)
Sets the trace level for the listener associated with the specified global service manager.
- [set trc_level](#)
Sets the `TRC_LEVEL` listener parameter.
- [show ddl](#)
Shows DDL statements execution status.
- [split chunk](#)
Splits each of the specified chunks into two chunks with an equal number of records. After the split, the chunks remain in the same shard.
- [sql](#)
Executes a SQL statement or a PL/SQL stored procedure against a sharded database.
- [start gsm](#)
Starts a specific global service manager.
- [start observer](#)
Starts specific services.
- [start ru](#)
Starts a specified replication unit.
- [start service](#)
Starts specific services.
- [status](#)
Displays the running status and runtime information for the global service manager.
- [status backup](#)
View the detailed state on the scheduled backup jobs in the specified shards.
- [status database](#)
Displays the runtime status of databases, such as registration information, services, and so on.
- [status gsm](#)
Displays the status of a specific global service manager.
- [status routing](#)
Displays the runtime routing information status.
- [status ru](#)
Displays runtime information about replication units for Oracle Globally Distributed AI Database native Raft replication.
- [status service](#)
Displays the status of a specific service.
- [stop gsm](#)
Stops a specific global service manager.
- [stop ru](#)
Stops a specified replication unit.
- [stop service](#)
Stops the specified global services.
- [suspend services](#)
This command allows users to block database on all GSM listeners.
- [switchover ru](#)
Switch leadership for the given replication unit to the specified database.

- [sync brokerconfig \(synchronize brokerconfig\)](#)
Synchronizes the Oracle Data Guard broker configuration in the global service manager with the configuration in the database pool. The `synchronize brokerconfig` command has the same options and usage.
- [sync database \(synchronize database\)](#)
Synchronizes attributes of global services and GDS related parameters of a GDS pool database with the contents of the GDS catalog. The `synchronize database` command has the same options and usage.
- [sync ru](#)
Synchronizes data of the specified replication unit on all shards, erases RAFT logs, and resets log index and term.
- [sync schema \(synchronize schema\)](#)
Allows common shared schemas across the existing databases to be retrieved. The command compares the schemas on all of the databases and retrieves those that are common.
- [validate backup](#)
The `validate backup` command provides sharded database (SDB) backup validation.
- [validate catalog](#)
Cross checks the Global Data Services catalog, global service manager run-time status, and pool databases, and reports inconsistencies and errors.
- [validate](#)
Cross checks the GDS catalog, global service manager run-time status, and databases from the GDS pool and reports any inconsistencies and errors.

add brokerconfig

Adds an Oracle Data Guard broker configuration to a Global Data Services pool.

Syntax

```
add brokerconfig -connect connect_identifier
                  [-pwd password]
                  [-gdspool gdspool_name]
                  [-region region_name]
                  [-savename]
                  [-force]
```

Options

Table 2-1 GDSCTL add brokerconfig Options

| Option | Description |
|---|--|
| <code>-connect <i>connect_identifier</i></code> | Specify an Oracle Net connect descriptor or net service name that resolves to a connect descriptor for a database in the broker configuration. |
| <code>-force</code> | If specified, the existing GDS configuration is deleted. Deletes an existing, running SDB, and should only be used if you want to get rid of the entire SDB. |

Table 2-1 (Cont.) GDSCTL add brokerconfig Options

| Option | Description |
|---|---|
| <code>-gdspool <i>gdspool_name</i></code> | The pool to which the databases of the Oracle Data Guard broker configuration are to be added. If the specified Global Data Services pool already contains databases or another configuration, GDSCTL returns an error. |
| <code>-pwd <i>password</i></code> | The password for the GSMUSER. If <code>-pwd</code> is not specified, then you are prompted for the password. |
| <code>-region <i>region_name</i></code> | The Global Data Services region to which the databases belong. If you specify a region, then all the databases are added to that region. If you do not specify a region, then all databases are added with a region of UNASSIGNED. If the region is UNASSIGNED, then you must use the modify database command to change the region. |
| <code>-savename</code> | Specify this option to store a net service name specified with the <code>-connect</code> option in the Global Data Services catalog, rather than the connect descriptor mapped to that net service name. |

Usage Notes

- You must connect to the Global Data Services catalog database as a user with the pool administrator privileges, the GSMUSER database account, using the [connect](#) command before running the `add brokerconfig` command. You should use the `CONNECT` command to connect to the GSMUSER for the database that you are adding the broker configuration for.
- If a GDS pool already has databases or another configuration, an error is returned. If `-region` is specified, it defines only the region of primary database. If there is more than one region in catalog, GDS region property of standbys will be unassigned. The user will have to use `modify database` to specify GDS region.

Examples

Add the Oracle Data Guard broker configuration for the DB1 database to the Global Data Services pool MYREADERFARM and the WEST region.

```
GDSCTL> add brokerconfig -connect 192.168.1.1:1521:sid -region west -gdspool myreaderfarm
```

Exceptions or Error Codes

GDSCTL returns the errors listed below if you use this command incorrectly.

Table 2-2 GDSCTL add brokerconfig Exceptions or Error Codes

| Exception | Description |
|-------------|--|
| ERROR-44866 | A pool can only contain one Data Guard broker configuration. If a Global Data Services pool already contains an Oracle Data Guard broker configuration, then GDSCTL returns error 44866 because a database must be added using Oracle Data Guard in this case. |

add cdb

Add a cdb to the shard catalog.

Syntax

```
add cdb -connect connect_identifier
        [-pwd gsmrootuser_pwd]
        [-savename]
        [-cpu_threshold cpu]
        [-disk_threshold disk]
        [-rack rack_id]
        [-force]
```

Options

Table 2-3 GDSCTL add cdb Options

| Option | Description |
|---|--|
| <code>-connect <i>connect_identifier</i></code> | Specify an Oracle Net connect descriptor or net service name that resolves to a connect descriptor for the database being added as the shard. |
| <code>-pwd <i>gsmrootuser_pwd</i></code> | Enter the GSMROOTUSER password. If not specified, the user is prompted for the password. |
| <code>-savename</code> | Store in the shard catalog a net service name specified with the <code>-connect</code> option rather than the connect descriptor mapped to that net service name. |
| <code>-force</code> | If specified, the existing GDS and sharding configuration on the shard and in the shard catalog with information about this shard will be rewritten. |
| <code>-cpu_threshold <i>cpu</i></code> | Specify the CPU Utilization percentage threshold. |
| <code>-disk_threshold <i>disk</i></code> | Specify the average latency in milliseconds of a synchronous single-block read. |
| <code>-rack <i>rack_id</i></code> | Specify an identifier of a rack (hardware cabinet), or another physical grouping of nodes with similar availability characteristics. If specified, GDS will enforce that databases that contain replicated data are not placed in the same rack. If this is not possible an error is raised. |

Usage Notes

ADD CDB adds metadata about a CDB to a sharding catalog. This command is only necessary if you intend to deploy a PDB as a shard with the `-cdb` option in the `ADD SHARD` command. CDBs can support multiple PDB shards from different sharded databases; however, this support is limited to only one PDB shard from a given sharded database for each CDB.

Examples

Adds a CDB called db11 to the shard catalog.

```
GDSCTL> add cdb -connect db11 -pwd gsmrootuser_pwd
```

add credential

Adds a credential which can be used by the remote scheduler agent to execute shard jobs.

Syntax

```
add credential -credential credential_name
               -osaccount account_name
               -ospassword password
               [-windows_domain domain_name]
```

Options

Table 2-4 GDSCTL add credential Options

| Option | Description |
|---|--|
| <code>-credential <i>credential_name</i></code> | Specify the name of the credential to add. |
| <code>-osaccount <i>account_name</i></code> | Specify the operating system account which will be used for remote jobs. |
| <code>-ospassword <i>password</i></code> | Specify the corresponding password for the account. |
| <code>-windows_domain <i>domain_name</i></code> | If a Windows account has been specified, specify the corresponding domain name for that account. |

Usage Notes

This command adds a credential which will be used to execute jobs on sharded hosts in response to administrative commands. The operating system account may be any valid account on the remote host which is in the OSDBA group; the account does not need to be enabled for interactive login unless it is used for other purposes. A specific non-interactive account may be created for use with the remote scheduler, if desired. The OS password must be a valid and current password for the specified account.

If the specified credential already exists, the command returns an error.

Examples

Add a credential named `east_region_cred`.

```
GDSCTL> add credential -credential east_region_cred -osaccount agent_user
               -ospassword password
```

add database

Adds databases to a Global Data Services region and Global Data Services pool.

Syntax

```
add database -connect connect_identifier
              [-region region_name]
              [-gds_pool gds_pool_name]
              [-pwd password]
```

```
[-savename]
[-cpu_threshold cpu]
[-disk_threshold disk]
[-validate_network]
```

Options

Table 2-5 GDSCTL add database Options

| Option | Description |
|---|--|
| <code>-connect <i>connect_identifier</i></code> | Specify an Oracle Net connect descriptor or net service name that resolves to a connect descriptor for the database being added. |
| <code>-cpu_threshold <i>cpu</i></code> | Specifies CPU Utilization percentage threshold. |
| <code>-disk_threshold <i>disk</i></code> | Specifies the average latency in milliseconds of a synchronous single-block read. |
| <code>-gdspool <i>gdspool_name</i></code> | The Global Data Services pool to which the database belongs. |
| <code>-pwd <i>password</i></code> | The password for the GSMUSER. If <code>-pwd</code> is not specified, then you are prompted for the password. |
| <code>-region <i>region_name</i></code> | The Global Data Services region to which the database belongs. |
| <code>-savename</code> | Specify this option to store a net service name specified with the <code>-connect</code> option in the Global Data Services catalog, rather than the connect descriptor mapped to that net service name. |
| <code>-validate_network</code> | This flag enables several network validation checks, including checking network connectivity between hosts and checking VNCR entries are valid. |

Usage Notes

Note

Only add a database to a GDS pool if its Oracle release, release update (RU), or recommended release update (RUR) and and datapatch level exactly match all existing database members. Never mix newer or older database binaries. If this is done, then it can cause errors and unstable services. Also, do not add or remove databases from the pool during rolling patching or upgrades. Onboard at the matched version first, verify services, and then perform rolling patching.

- You must connect to the Global Data Services catalog database as a user with the pool administrator privileges, using the [connect](#) command before running this command.
- If `-savename` is *not* specified, then GDSCTL replaces what you specify for the net service name with the full connection string before saving the configuration to the catalog.
- The default for GDSCTL is for `autovncr` to be enabled for the catalog. If `autovncr` has been disabled for the catalog, before configuring Global Data Services pools and adding databases to the Global Data Services configuration, the nodes where those databases run must be part of the valid node checking for registration (VNCR) list for database

registration. Use the [add invitednode \(add invitedsubnet\)](#) command to define the valid nodes.

Example

Adds database DB1 to the WEST region and Global Data Services pool MYREADERFARM.

```
GDSCTL> add database -connect 127.0.0.1:1521:db1 -region west -gdspool
myreaderfarm
```

Adds a database using *myalias* instead of the IP address connection string.

```
GDSCTL> add database -connect myalias -gdspool myreaderfarm
```

Exceptions or Error Codes

GDSCTL returns the errors listed below if you use this command incorrectly.

Table 2-6 GDSCTL add database Exceptions or Error Codes

| Exception | Description |
|-------------|---|
| ERROR-44866 | If a pool already contains an Oracle Data Guard broker configuration, then GDSCTL returns an error; you must add databases using Oracle Data Guard in this case. That is, if a pool contains an Oracle Data Guard broker configuration, then additional databases can only be added to the pool by adding them to that Data Guard broker configuration. |
| ERROR-44868 | If the database being added is part of a Oracle Data Guard broker configuration, then GDSCTL returns an error; you must use the add brokerconfig command in this case. |

add file

Adds the contents of a file to the catalog which can be used by subsequent GDSCTL commands.

Syntax

```
add file -file file_name
        -source local_filename
```

Options

Table 2-7 GDSCTL add file Options

| Option | Description |
|-------------------------------|--|
| -file <i>file_name</i> | Specify the name of the file object to add. |
| -source <i>local_filename</i> | Specify an operating system file name specifying a file local to the machine running GDSCTL. |

Usage Notes

This command creates a named file object in the catalog and associates the contents of an operating system file with that object by opening the file and storing its contents in the catalog. If the contents of the operating system file change, the MODIFY FILE command can be used to reload the contents into the catalog.

If the specified file object already exists, the command returns an error.

Examples

Add a file named `east_region_db_params` from the local source file `/tmp/dbca_params.txt`

```
GDSCTL> add file -file east_region_db_params -source /tmp/dbca_params.txt
```

add gdspool

Adds a Global Data Services pool to the Global Data Services framework.

Syntax

```
add gdspool -gdspool gdspool_name_list
           [-users user_list]
```

Options

Table 2-8 GDSCTL add gdspool Options

| Option | Description |
|--|---|
| <code>-gdspool <i>gdspool_name_list</i></code> | A comma-delimited list of Global Data Services pool names. A Global Data Services pool must have a unique name within its GDS configuration. If you do not specify a name for the pool when you create it, then the name defaults to <code>oradbpool</code> . The pool name can be up to 30 bytes long and can be any valid identifier (an alphabetical character followed by zero or more alphanumeric ASCII characters or the underscore (_)). |
| <code>-users <i>user_list</i></code> | A comma-delimited list of users that are granted the pool administrator role. |

Usage Notes

- A default GDS pool, `DBPOOLORA`, will be created automatically when a GDS catalog is created using [create gds catalog](#).
- You must connect to the Global Data Services catalog database as a user with the Global Data Services administrator privileges, using the [connect](#) command before running this command.
- The default for GDSCTL is for `autoovncr` to be enabled for the catalog. If `autoovncr` has been disabled for the catalog, then before configuring Global Data Services pools and adding databases to the Global Data Services configuration, the nodes where those

databases run must be part of the valid node checking for registration (VNCR) list for database registration. Use the [add invitednode \(add invitedsubnet\)](#) command to define the valid nodes.

Example

Add a Global Data Services pool named MYREADERFARM to the configuration:

```
GDSCTL> add gdspool -gdspool myreaderfarm
```

add gsm

Adds a global service manager to the Global Data Services framework.

Syntax

```
add gsm -gsm gsm_name
        -catalog connect_id
        [-pwd password]
        [-wpwd password]
        [-region region_name]
        [-localons ons_port]
        [-remoteons ons_port]
        [-listener listener_port]
        [-endpoint gmsendpoint]
        [-remote_endpoint remote_endpoint]
        [-trace_level level]
        [-encryption encryption]
        [-validate_network]
```

Options

Table 2-9 GDSCTL add gsm Options

| Option | Description |
|--------------------------------|---|
| -catalog <i>connect_id</i> | Specify the connect identifier for the Global Data Services catalog database. If a network service name is specified, it must be resolvable by the local naming method to a connect descriptor that allows the global service manager being added to connect to the catalog database. |
| -endpoint <i>gmsendpoint</i> | Specifies the protocol address that the global services manager listens on for client connection requests. If you use this option, the value that you specify overrides the default endpoint. |
| -gsm <i>gsm_name</i> | Specify the name of the global service manager that you want to add. If you do not specify a name, then GDSCTL uses the current global service manager name for the session (specified with the command set gsm). |
| -listener <i>listener_port</i> | Specify the listener port. The default port is 1522. |

Table 2-9 (Cont.) GDSCTL add gsm Options

| Option | Description |
|--|---|
| <code>-localons <i>ons_port</i></code> | Specify the local ONS port. If you do not specify this option, then GDSCTL uses the default ONS port (which is 6123 on most platforms). |
| <code>-pwd <i>password</i></code> | Specify the password for the GSMCATUSER. If you do not specify a password, then you are prompted to enter one. |
| <code>-region <i>region_name</i></code> | Specify the region to which the global service manager belongs. The value for <i>region_name</i> must match the name of an existing Global Data Services region. If you do not specify a region, then GDSCTL adds the global service manager without assigning a region. |
| <code>-remote_endpoint <i>remote_endpoint</i></code> | Specifies the protocol address that is used by the global service manager to receive database registration requests and communicate with other global service managers in the configuration. If you use this option, the value that you specify overrides the default endpoint. |
| <code>-remoteons <i>ons_port</i></code> | Specify the remote ONS port. If you do not specify this option, then GDSCTL uses the default ONS port (which is 6234 on most platforms). |
| <code>-trace_level <i>level</i></code> | Specify the global service manager trace level (to be used as directed by Oracle Support Services). |
| <code>-wpwd <i>password</i></code> | Specify a password to protect the global service manager wallet. If a wallet password is not specified, a system-generated password is used instead. Note that if a password is specified with this option, the wallet cannot be modified without supplying that password. |
| <code>-encryption <i>encryption</i></code> | Encryption protocol for Advanced Network Option (ANO) used between GSM, GDSCTL and databases. OFF means that ANO is disabled. (AES256 AES192 OFF). |
| <code>-validate_network</code> | This flag enables several network validation checks, including checking network connectivity between hosts and checking VNCR entries are valid. |

Usage Notes

- You must specify the Global Data Services catalog database when using this command.
- You must run this command, locally, on the computer where you want to add the global service manager.
- You must have operating system privileges on the computer where you want to add the global service manager to run this command.
- When you run this command, GDSCTL connects to the Global Data Services catalog as the GSMCATUSER user and prompts you for the GSMCATUSER password.

Example

Add a global service manager named `gsm1`, specifying the location of the Global Data Services catalog database, `DB1`.

```
GDSCTL> add gsm -gsm gsm1 -catalog 127.0.0.1:1521:db1
```

add invitednode (add invitedsubnet)

Adds host address or subnet information to the valid node checking for registration (VNCR) list in the catalog, before starting the first global service manager, by establishing a direct connection to the Global Data Services catalog database.

Syntax

```
add {invitednode | invitedsubnet}
    [-group group_name]
    [-catalog catalog_dbname [-user user_name/password]]
    vncr_id
    [-validate_network]
```

Options

Table 2-10 GDSCTL add invitednode (add invitedsubnet) Options

| Option | Description |
|--|---|
| <code>-catalog <i>catalog_dbname</i></code> | Specify the Global Data Services catalog database net alias or connect string. If you enter an invalid address or connect string, then GDSCTL uses the pre-established connection created with the connect command. |
| <code>-group <i>group_name</i></code> | Specify an alias which defines a group of invited nodes. This alias can be referenced in other commands related to invited nodes. |
| <code>-user <i>user_name[/password]</i></code> | Specify the user credentials for the Global Data Services administrator in the catalog database. If you do not specify a user or a password, then GDSCTL prompts you this information. |
| <code><i>vncr_id</i></code> | Specify the list of nodes that can register with the global service manager. The list can include host names or CIDR notation for IPv4 and IPv6 addresses. The wildcard format (*) is supported for IPv4 addresses. The presence of a host name in the list results in the inclusion of all IP addresses mapped to the host name. The host name should be consistent with the public network interface. |
| <code>-validate_network</code> | This flag enables several network validation checks, including checking network connectivity between hosts and checking VNCR entries are valid. |

Usage Notes

- You must connect to the Global Data Services catalog database as a user with the pool administrator privileges, using the [connect](#) command before running this command.
- The default for GDSCTL is that `autovncr` is enabled for the catalog. If `autovncr` has been disabled for the catalog, before configuring Global Data Services pools and adding

databases to the Global Data Services configuration, then the nodes where those databases run must be part of the valid node checking for registration (VNCR) list for database registration. Use the [add invitednode \(add invitedsubnet\)](#) command to define the valid nodes.

- VNCR enables or denies access from specified IP addresses to Oracle services. See *Oracle AI Database Net Services Administrator's Guide* for more information about VNCR.

Examples

Add the netmask 255.255.255.248 to the catalog.

```
GDSCTL> add invitednode 255.255.255.248
```

Add the server east1.example.com to the catalog in the alias group EAST_SRV.

```
GDSCTL> add invitednode east1.example.com
```

Add the server east2.example.com to the catalog in the alias group EAST_SRV.

```
GDSCTL> add invitednode east2.example.com
```

add region

Adds a region to a Global Data Services framework or an Oracle Globally Distributed AI Database configuration.

Syntax

```
add region -region region_list
          [-buddy region_name]
```

Options

Table 2-11 GDSCTL add region Options

| Option | Description |
|----------------------------|--|
| -buddy <i>region_name</i> | Specify the name of the buddy region. |
| -region <i>region_list</i> | Specify a comma-delimited list of Global Data Services region names. A Global Data Services region should have a name that is unique within the corresponding Global Data Services configuration. If no name is specified at the first region creation time, the default name, oraregion, is given to the region. The region name can be up to 30 characters long and can be any valid identifier - an alphabetical character followed by zero or more alphanumeric ASCII characters or underscore (_). |

Usage Notes

- When the Global Data Services catalog is created using the [create gdscatalog](#) command, the default REGIONORA region is created automatically.

- You must connect to the Global Data Services catalog database as a user with the Global Data Services administrator privileges, using the command [connect](#) before running this command

Example

Add two Global Data Services regions, EAST and WEST to the current configuration:

```
GDSCTL> add region -region east,west
```

add service

Adds a global service to a Global Data Services pool.

Syntax

```
add service
    [-gdspool gdspool_name]
    -service service_name
    (-preferred_all | (-preferred dbname_list [-available
dbname_list]))
    [-locality {ANYWHERE | LOCAL_ONLY [-region_failover]}]
    [-role {PRIMARY | PHYSICAL_STANDBY [-failover_primary] |
    LOGICAL_STANDBY | SNAPSHOT_STANDBY | TRUE_CACHE}]
    [-lag {lag_value | ANY}]
    [-notification {TRUE | FALSE}]
    [-rlbgoal {SERVICE_TIME | THROUGHPUT}]
    [-dtp {TRUE | FALSE}]
    [-sql_translation_profile stp_name]
    [-clbgoal {SHORT | LONG}]
    [-tafpolicy {BASIC | NONE | PRECONNECT}]
    [-policy policy]
    [-failovertype {NONE | SESSION | SELECT | TRANSACTION | AUTO}]
    [-failovermethod {NONE | BASIC}]
    [-failoverretry failover_retries]
    [-failoverdelay failover_delay]
    [-edition edition_name]
    [-commit_outcome {TRUE | FALSE}]
    [-retention retention_seconds]
    [-session_state {DYNAMIC | STATIC | AUTO}]
    [-replay_init_time replay_init_time]
    [-pdbname pdbname]
    [-drain_timeout]
    [-stop_option {NONE,IMMEDIATE, TRANSACTIONAL}]
    [-failover_restore {NONE|LEVEL1|AUTO}]
    [-table_family family]
    [-failover_restore {NONE|LEVEL1|AUTO}]
    [-reset_state {NONE|LEVEL1}]
```

Options

Table 2-12 GDSCTL add service Options

| Option | Description |
|---|---|
| <code>-available dbname_list</code> | <p>Specify a comma-delimited list of available databases on which the service runs if the preferred databases are not available. You <i>cannot</i> specify a list of available instances, only databases. You can use the modify service command with the <code>-server_pool</code> parameter to specify instance-level preferences.</p> <p>The list of available databases must be mutually exclusive with the list of preferred databases.</p> <p>You <i>cannot</i> use this option with the <code>-preferred_all</code> option.</p> <p>For True Cache implementations, specify a comma-delimited list of available databases or True Caches on which the service runs if the preferred ones are not available. You can't specify a list of available instances, only databases and True Caches. You can use the <code>modify service</code> command with the <code>-server_pool</code> parameter to specify instance-level preferences.</p> <p>The list of available databases or True Caches must be mutually exclusive with the list of preferred ones.</p> <p>You cannot use this option with the <code>-preferred_all</code> option.</p> |
| <code>-clbgoal {SHORT LONG}</code> | <p>Connection Load Balancing Goal. Use a value of <code>SHORT</code> for this parameter for run-time load balancing, or if using an integrated connection pool. Use a value of <code>LONG</code> for this parameter for long running connections, such as batch jobs, that you want balanced by the number of sessions for each node for the service.</p> <p>The default value for this option, if not specified, is <code>SHORT</code>.</p> <p>For True Cache services, set the connection load balancing goal to <code>SHORT</code> for runtime load balancing.</p> |
| <code>-commit_outcome {TRUE FALSE}</code> | <p>Enable Transaction Guard; when set to <code>TRUE</code>, the commit outcome for a transaction is accessible after the transaction's session fails due to a recoverable outage.</p> |
| <code>-drain_timeout</code> | <p>Set drain time in seconds.</p> |
| <code>-dtp {TRUE FALSE}</code> | <p>Indicates whether Distributed Transaction Processing should be enabled for this service. This service can either be a service in a policy-managed database or a preferred service on a single node in an administrator-managed database.</p> |

Table 2-12 (Cont.) GDSCTL add service Options

| Option | Description |
|--|--|
| <code>-edition <i>edition_name</i></code> | <p>Specify the initial session edition of the service.</p> <p>When an edition is specified for a service, all subsequent connections that specify the service use this edition as the initial session edition. However, if a session connection specifies a different edition, then the edition specified in the session connection is used for the initial session edition.</p> <p>GDSCTL does not validate the specified edition name. During connection, the connect user must have <code>USE</code> privilege on the specified edition. If the edition does not exist or if the connect user does not have <code>USE</code> privilege on the specified edition, then an error is raised.</p> |
| <code>-failover_primary</code> | <p>If you set the <code>-role</code> option to <code>PHYSICAL_STANDBY</code>, then you can use this option to enable the service for failover to the primary database.</p> <p>If the <code>-role</code> is <code>TRUE_CACHE</code>, you can use this option to enable the service to failover to the primary database if no True Caches are available. Also include the primary database in the <code>-preferred</code> or <code>-available</code> list, or use the <code>-preferred_all</code> option.</p> |
| <code>-failoverdelay <i>failover_delay</i></code> | For Application Continuity and TAF, this parameter specifies the time delay (in seconds) between reconnect attempts for each incident at failover. |
| <code>-failovermethod {NONE BASIC}</code> | <p>TAF failover method (for backward compatibility only).</p> <p>If the failover type (<code>-failovertime</code>) is set to a value other than <code>NONE</code>, then you should choose <code>BASIC</code> for this parameter.</p> |
| <code>-failoverretry <i>failover_retries</i></code> | For Application Continuity and TAF, this parameter determines the number of attempts to connect after an incident. |
| <code>-failovertime {NONE SESSION SELECT TRANSACTION}</code> | <p>Specify the failover type.</p> <p>To enable Application Continuity for Java, set this parameter to <code>TRANSACTION</code>. To enable Transparent Application Failover (TAF) for OCI, set this parameter to <code>SELECT</code> or <code>SESSION</code>.</p> |
| <code>-gdspool <i>gdspool_name</i></code> | Specify the name of the Global Data Services pool to which you want to add a service. If the pool name is not specified and there is only one <code>gdspool</code> with access granted to the user, then this the <code>gdspool</code> with access granted is used as the default <code>gdspool</code> . |

Table 2-12 (Cont.) GDSCTL add service Options

| Option | Description |
|-----------------------------------|---|
| -lag {lag_value ANY} | <p>Specify the lag for the service in seconds. You can use the keyword <code>ANY</code> to indicate that there is no upper threshold on the lag time. This parameter specifies the maximum lag that a provider of this service may have. The service cannot be provided by a database whose lag exceeds this value.</p> <p>If a True Cache falls behind the primary database beyond the specified lag time, the service stop forwarding requests to that True Cache until it catches up.</p> <p>The default value for <code>lag</code>, if not specified, is <code>ANY</code>.</p> |
| -locality {ANYWHERE LOCAL_ONLY} | <p>Specify the service region locality. If you do not specify this option, then GDSCTL uses the default value of <code>ANYWHERE</code> for the service.</p> <p>For True Cache services, set the locality to <code>LOCAL_ONLY</code>. This indicates that GDS only routes to True Caches in the same region, regardless of load. Use this option together with <code>-region_failover</code> so that client connections and requests are routed to another region if all True Caches in a region have failed.</p> |
| -notification {TRUE FALSE} | <p>Enable Fast Application Notification (FAN) for OCI connections.</p> |
| -pdbname <i>pdb_name</i> | <p>Specify the pluggable database name.</p> |
| -policy {AUTOMATIC MANUAL} | <p>Specify the management policy for the service.</p> <p>If you specify <code>AUTOMATIC</code> (the default), then the service automatically starts when the database restarts, either by a planned restart or after a failure. Automatic restart is also subject to the service role.</p> <p>If you specify <code>MANUAL</code>, then the service is never automatically restarted upon planned restart of the database. A <code>MANUAL</code> setting does not prevent the global service manager from monitoring the service when it is running and restarting it if a failure occurs.</p> |
| -preferred <i>dbname_list</i> | <p>Specify a comma-delimited list of preferred databases or True Caches on which the service runs. You can't specify preferred instances, only databases and True Caches. You can use the modify service command to specify instance-level preferences.</p> <p>The list of preferred databases must be mutually exclusive with the list of available databases.</p> <p>You <i>cannot</i> use this option with the <code>-preferred_all</code> option.</p> |

Table 2-12 (Cont.) GDSCTL add service Options

| Option | Description |
|--|---|
| -preferred_all | <p>Specifies that all the databases in the Global Data Services pool are preferred databases. Any databases you later add to the pool are configured as preferred databases for this service.</p> <p>You <i>cannot</i> use this option with the <code>-preferred</code> and <code>-available</code> options.</p> <p>For True Cache services, indicates that this service will be started on all existing and future True Caches and, if used with <code>-failover_primary</code>, the primary database.</p> |
| -region_failover | <p>Indicates that the service is enabled for region failover. You can only use this option when you specify <code>LOCAL_ONLY</code> for the <code>-locality</code> option.</p> |
| -replay_init_time <i>replay_init_time</i> | <p>For Application Continuity, this parameter specifies the time (in seconds) after which replay cannot be initiated. The default value is 300 seconds.</p> |
| -retention <i>retention_seconds</i> | <p>If <code>commit_outcome</code> is set to <code>TRUE</code>, then this parameter determines the amount of time (in seconds) that the commit outcome is retained in the database.</p> |
| -rlbgoal {SERVICE_TIME THROUGHPUT} | <p>Run-time Load Balancing Goal (for the Load Balancing Advisory). Set this parameter to <code>SERVICE_TIME</code> to balance connections by response time. Set this parameter to <code>THROUGHPUT</code> to balance connections by throughput.</p> <p>For True Cache services, set the runtime load balancing goal to <code>SERVICE_TIME</code> to balance connections by response time instead of by throughput.</p> <p>If you do not use this option, then the value defaults to <code>SERVICE_TIME</code> for the run-time load balancing goal.</p> |
| -role {[PRIMARY] [PHYSICAL_STANDBY] TRUE_CACHE} [-failover_primary] [LOGICAL_STANDBY] [SNAPSHOT_STANDBY]} | <p>Specify the database role that the database must be for this service to start on that database. This applies only to Global Data Services pools that contain an Oracle Data Guard broker configuration.</p> <p>For True Cache services, use <code>TRUE_CACHE</code>. This ensure that the service only runs on other True Caches, not on the primary database</p> <p>See Also: <i>Oracle Data Guard Concepts and Administration</i> for more information about database roles</p> |

Table 2-12 (Cont.) GDSCTL add service Options

| Option | Description |
|---|--|
| <code>-service <i>service_name</i></code> | <p>Specify the name of the global service.</p> <p>The service name specified in the <code>add service</code> command can be domain qualified (for example, <code>sales.example.com</code>) or not (for example, <code>sales</code>). If the specified name is not domain qualified, the service is created with the default domain name <code><GDS_pool_name>.<GDS_configuration_name></code>, however the shorter non-domain qualified name can be used with subsequent <code>gdctl</code> commands to manage the service. If the specified name is domain qualified, the full domain qualified service name must be used in all subsequent <code>gdctl</code> commands used to manage the service.</p> <p>A global service name must be unique within a GDS pool and when qualified by domain, must also be unique within a GDS configuration. A global service cannot be created at a database if a local or global service with the same name already exists at that database.</p> <p>A global service name can contain alphanumeric characters, underscore (<code>_</code>), and period (<code>.</code>). The first character must be alphanumeric. The maximum length of a global service name is 64 characters. The maximum length of a domain qualified global service name is 250 characters.</p> <p>An Oracle Net connect descriptor used to connect to a global service must contain a domain qualified service name.</p> |
| <code>-session_state {DYNAMIC STATIC}</code> | <p>For Application Continuity, this parameter specifies whether the session state that is not transactional is changed by the application. A setting of <code>DYNAMIC</code> is recommended for most applications.</p> |
| <code>-sql_translation_profile <i>stp_name</i></code> | <p>Use this option to specify a SQL translation profile for a service that you are adding after you have migrated applications from a non-Oracle database to an Oracle database.</p> <p>This option corresponds to the SQL translation profile parameter in the <code>DBMS_SERVICE</code> service attribute.</p> <p>Notes:</p> <ul style="list-style-type: none"> • Before using the SQL translation feature, you must migrate all server-side application objects and data to the Oracle database. • Use the command config service to display the SQL translation profile. <p>See Also: <i>Oracle AI Database SQL Translation and Migration Guide</i> for more information about SQL translation</p> |
| <code>-stop_option</code> | <p>Set the default stop option to <code>NONE</code>, <code>IMMEDIATE</code>, or <code>TRANSACTIONAL</code></p> |

Table 2-12 (Cont.) GDSCTL add service Options

| Option | Description |
|---|---|
| <code>-table_familyfamily</code> | Specifies the name of the table family as a property of the service. This parameter takes one of the table family values (root table schema.name) as shown in the CONFIG TABLE FAMILY output. If the schema name or the table name is case-sensitive, use two-level quotes (single quotes outside, double quotes inside) around the whole string, for example, ' "TESTUSER1.Customers6" '. No quotes are needed if neither name is case sensitive. If this parameter is not specified, but there is currently only one table family, the service created with the add service command is automatically associated with that table family. |
| <code>-tafpolicy {BASIC NONE }</code> | TAF policy specification (for administrator-managed databases only). |
| <code>-failover_restore</code> | Session state restoration for Application Continuity (NONE LEVEL1 AUTO) |
| <code>-reset_state</code> | Reset session state (NONE -default LEVEL1 AUTO) |

Usage Notes

Database-specific options cannot be set at this level. The `modify service` command must be used to set database-specific options.

One of `-preferred_all` or `-preferred` must be specified. If `-preferred_all` is specified, then all databases in the pool are preferred for this global service (databases inserted into the pool will also have this global service added).

Services cannot be added to systems that run Clusterware unless Clusterware is started on the system, even if the target database is not under CRS control. Ensure that Clusterware is started on Clusterware enabled systems before adding a new global service.

In Oracle Sharding, note that when there is no `table_family` parameter specified, the service is not associated with any table family, and the value of the property is set to NULL. This is the case for user-defined and composite sharding, where there is always only one table family, and can also be the case when there is only one table family in system-managed sharding. When a table family is deleted (that is, the root table of the table family is dropped) the `table_family` property of the service is reset to NULL.

Note

Starting with Oracle AI Database 26ai, the session attribute values `FAILOVER_TYPE = TRANSACTION` with `SESSION_STATE_CONSISTENCY = STATIC` are no longer a supported service attribute combination. Instead use one of the following combinations in the service configuration: `FAILOVER_TYPE = AUTO` with `SESSION_STATE_CONSISTENCY = AUTO` or `FAILOVER_TYPE = TRANSACTION` with `SESSION_STATE_CONSISTENCY = DYNAMIC`. These configurations enforce session state tracking in the Oracle database ensuring that session state is preserved at session migration and session failover.

Examples

Add a service named `sales_report` to the Global Data Services pool `MYREADERFARM` with a value of `ANYWHERE` for the locality.

```
GDSCTL> add service -gdspool myreaderfarm -service sales_report -locality
ANYWHERE
```

Add a service named `daily_sales_rept` to the Global Data Services pool `MYDGPOOL` with preferred instance set to `DB1` and the available instances set to `DB3` and `DB4`. The service should use the basic transaction failover policy.

```
GDSCTL> add service -gdspool mydgpool -s daily_sales_rept -preferred db1
    -available db3,db4 -tafpolicy BASIC
```

In a system-managed sharded database, the table family ID parameter is specified as a property of the service.

```
GDSCTL> add service -gdspool shdpool -table_family sales.customer -service
sales -preferred_all -locality ANYWHERE
```

Related Topics

- [Creating a Global Service](#)

add shard

Add a shard to the shard catalog.

Syntax

```
add shard -connect connect_identifier
           [-pwd password]
           [-savename]
           [-region region_name]
           [-force]
           [-cdb cdb_name]
           [-cpu_threshold cpu]
           [-disk_threshold disk]
           [{-shardgroup shardgroup_name | -shardspace shardspace_name}]
           [-deploy_as {PRIMARY | STANDBY | ACTIVE_STANDBY}]
           [-rack rack_id]
           [-replace old_db_name]
           [-gg_service (http|https):ogg_host:sm_port/GGHOME_directory]
           [-validate_network]
```

Options

Table 2-13 GDSCTL add shard Options

| Option | Description |
|--|---|
| <code>-connect connect_identifier</code> | Specify an Oracle Net connect descriptor or net service name that resolves to a connect descriptor for the database being added as the shard. |
| <code>-pwd password</code> | Enter the GSMUSER password. If not specified, the user is prompted for the password. |
| <code>-savename</code> | Store in the shard catalog a net service name specified with the <code>-connect</code> option rather than the connect descriptor mapped to that net service name. |
| <code>-region region_name</code> | Specify the GDS region that this shard belongs to. This parameter is only valid for user-defined sharding. For other sharing methods it is specified per shardgroup. |
| <code>-force</code> | If specified, the existing GDS and sharding configuration on the shard and in the shard catalog with information about this shard will be rewritten. |
| <code>-cdb cdb_name</code> | If this parameter is used, the shard must be a PDB and the CDB must already exist in the catalog. |
| <code>-cpu_threshold cpu</code> | Specify the CPU Utilization percentage threshold. |
| <code>-disk_threshold disk</code> | Specify the average latency in milliseconds of a synchronous single-block read. |
| <code>{-shardgroup shardgroup_name -shardspace shardspace_name}</code> | Specify the name of the shardgroup or shardspace that this shard is being added to. Use <code>-shardspace</code> when using this command in a user-defined sharding configuration. Use <code>-shardgroup</code> with system-managed and composite sharding configurations. |
| <code>-deploy_as {PRIMARY STANDBY ACTIVE_STANDBY}</code> | Specify the role that is assigned to a shard added to the shardgroup after deployment. This parameter is only used with Data Guard replication. The specified role will be assigned to the shard database after deployment. The valid values are: <ul style="list-style-type: none"> PRIMARY – the shard should be deployed as the primary database STANDBY – the shard should be deployed as a Data Guard standby (mounted) ACTIVE_STANDBY – the shard should be deployed as an Active Data Guard standby If the parameter is not specified, the default value is STANDBY |
| <code>-rack rack_id</code> | Specify an identifier of a rack (hardware cabinet), or another physical grouping of nodes with similar availability characteristics. If specified, GDS will enforce that databases that contain replicated data are not placed in the same rack. If this is not possible an error is raised. |

Table 2-13 (Cont.) GDSCTL add shard Options

| Option | Description |
|---|--|
| <code>-replace old_db_name</code> | This parameter specifies <code>db_unique_name</code> of the old shard when replacing it. The existing parameters of the <code>ADD SHARD</code> command (such as <code>connect</code>) must refer to attributes for the new (replacement) shard. This parameter is not supported in an Oracle GoldenGate environment. |
| <code>-gg_service (http https):ogg_host:sm_port/deployment</code> | This parameter is mandatory for Oracle GoldenGate replication and specifies the URI for the GoldenGate Admin Server that will manage the GoldenGate replication at this shard. The format will be as follows Example: <code>shard1.example.com:9005/shard1.</code> |
| <code>-validate_network</code> | This flag enables several network validation checks, including checking network connectivity between hosts and checking VNCR entries are valid. |

Usage Notes

Before running `add shard`, you must validate the shard by running the `validateShard` procedure as described in *Oracle Globally Distributed AI Database*

- The shard will become part of the sharded database after a `DEPLOY` command is executed, except in the case of `-replace`.
- Any databases added to a sharding configuration using `ADD SHARD` must not have ever been deployed as a shard in another configuration (unless `-replace` is specified). Re-adding a previously deployed shard will cause the `ADD SHARD` command to succeed, but the shard will be unable to successfully deploy and register with the shard director (GSM) when the `DEPLOY` command is eventually run.
- `ADD SHARD` only registers the database (shard) with GDS. Replication is not configured on a newly added database and data from other databases is not distributed to it until `DEPLOY` is run.
- With the introduction of Oracle AI Database, The default behavior when adding a shard to the topology has changed so that a current version of the schema is captured and applied on the new shard. Previously, all accumulated DDL on the sharded database was applied to the new shard, one by one in order, some of which might be negated by later DDL
- With Data Guard replication, a shard can be added as a standby to a preexisting Data Guard configuration. There is no need to re-shard the data. It is expected that the shard being added is in a the correct state for configuration; the standbys should be cloned from the primary and have the same DBID. When you run `DEPLOY`, the existing primary and standby databases are matched with each other, using the DBID to form a broker configuration. If the broker has not been configured, it is configured, otherwise it is validated that it has been configured correctly. Once the broker is configured, Data Guard does its work, and it should be able to perform catch-up on the standbys if needed before bringing them online. Note that the broker is not configured for a federated database.
- The `-DEPLOY_AS` option cannot be used in conjunction with `-SHARDGROUP` for system/composite sharding.
- When using the `-replace` parameter, see *Oracle Globally Distributed AI Database* for more information about its usage.

Note: Oracle GoldenGate replication support for Oracle Sharding High Availability is deprecated in Oracle Database 21c.

Examples

Add the shard to shardgroup GROUP1 of the DB11 database.

```
GDSCTL> add shard -connect db11 -shardgroup group1
```

Replace shard SH1 with database DB11.

```
GDSCTL> add shard -replace sh1 -connect db11
```

add shardgroup

Add a shardgroup to a shardspace.

Syntax

```
add shardgroup -shardgroup shardgroup_name
                [-region region_name]
                [-shardspace shardspace_name]
                [-deploy_as {PRIMARY | STANDBY | ACTIVE_STANDBY}]
                [-refactor number]
```

Options

Table 2-14 GDSCTL add shardgroup Options

| Option | Description |
|--|--|
| <code>-shardgroup <i>shardgroup_name</i></code> | Specify the name of the shardgroup. The name must be unique across all shardspaces. The shardgroup name can be up to 30 characters long and can be an alphabetical character followed by zero or more alphanumeric ASCII characters or an underscore (_). |
| <code>-region <i>region_name</i></code> | Specify the name of the region. If not specified, a default region will be used. |
| <code>-shardspace <i>shardspace_name</i></code> | Specify the name of the shardspace to which to add the shardgroup. |
| <code>-deploy_as {PRIMARY STANDBY ACTIVE_STANDBY}</code> | Specify the role that is assigned to a shard added to the shardgroup after deployment. This parameter is only used with Data Guard replication. The valid values are: If the parameter is not specified, the default value is STANDBY <ul style="list-style-type: none"> PRIMARY – the shard should be deployed as the primary database STANDBY – the shard should be deployed as a Data Guard standby (mounted) ACTIVE_STANDBY – the shard should be deployed as an Active Data Guard standby |

Table 2-14 (Cont.) GDSCTL add shardgroup Options

| Option | Description |
|--------------------------------|---|
| <code>-repfactor number</code> | Specify the replication factor - the number of replicas for each piece of data stored in this shardgroup. This parameter can only be used with Oracle GoldenGate replication and is mandatory unless the default value of replication factor was specified in <code>CREATE SHARDCATALOG</code> command. This parameter does not apply to user-defined sharding because GoldenGate does not support that sharding methods. |

Usage Notes

This command can only be used with system-managed or composite sharding, not user-defined sharding.

Note: Oracle GoldenGate replication support for Oracle Sharding High Availability is deprecated in Oracle Database 21c.

Examples

Add the GROUP1 shardgroup in the WEST region within the GOLD shardspace.

```
GDSCTL> add shardgroup -shardgroup group1 -region west -shardspace gold
```

add shardspace

Add a shardspace to the shard catalog.

Syntax

```
add shardspace -shardspace shardspace_name
                [-chunks number]
                [-protectmode dg_protection_mode]
                [repfactor][repunits]
```

Options**Table 2-15 GDSCTL add shardspace Options**

| Option | Description |
|---|--|
| <code>-shardspace <i>shardspace_name</i></code> | Specify the name of the shardspace. The shardspace name can be up to 30 characters long and can be an alphabetical character followed by zero or more alphanumeric ASCII characters or an underscore (_). |

Table 2-15 (Cont.) GDSCTL add shardspace Options

| Option | Description |
|---|---|
| <code>-chunks <i>number</i></code> | Specify the number of unique chunks in the shardspace. The value of <code>-chunks</code> must be greater than 2 times the size of the largest shardgroup in any shardspace. This parameter does not apply to user-defined sharding or a federated database. All shardgroups in a shardspace have the same number of chunks. If this parameter is not specified, the default number of chunks is determined at the time of execution of the first <code>DEPLOY</code> command and is 120 per database of the shardgroup with the biggest number of databases. |
| <code>-protectmode <i>dg_protection_mode</i></code> | Specify the Data Guard protection mode: <code>MAXPROTECTION</code> , <code>MAXAVAILABILITY</code> , or <code>MAXPERFORMANCE</code> (default). This parameter does not apply to Oracle GoldenGate replication. |
| <code>-repfactor</code> | Replication factor (the number of replicas for each piece of data stored in a shardgroup). This parameter can only be used with <code>NATIVE</code> replication and system-managed or composite sharding, and is mandatory in these cases. It doesn't apply to user-defined sharding or a federated database since there are no shardgroups in this case. |
| <code>-repunits</code> | Total number of replication units (SNR only). |

Usage Notes

The command is applicable to user-defined sharding, composite sharding that assumes multiple shardspaces, and system managed sharding when there are no other shardspaces present in the current configuration.

Examples

Add the `GOLD` shardspace with Data Guard `MAXAVAILABILITY` protection mode.

```
GDSCTL> add shardspace -shardspace gold -protectmode maxavailability
```

alter move

Suspend, resume, or cancel scheduled chunk move operations on a sharded database.

Syntax

```
ALTER MOVE { -RESUME | -SUSPEND | -CANCEL }
           [-CHUNK {chunk_id_list}]
           [-SHARD shd_lst]
           [-verbose]
```

Options

Table 2-16 GDSCTL ALTER MOVE Options

| Option | Description |
|-----------------------------|---|
| -cancel | Removes chunks from the pending chunk moves list for the specified scope. |
| -chunk <i>chunk_id_list</i> | List comma separated numeric chunk identifiers or use ALL for all chunks in the sharded database. |
| -resume | Resumes (reschedules) chunk moves which are in the "move failed" state for the specified scope. |
| -shard <i>shd_lst</i> | A comma separated list of shards. |
| -suspend | Suspends pending chunk moves for the specified scope. |
| -verbose | Enables verbose output mode. |

Usage Notes

Suspending chunk moves: Use `ALTER MOVE -SUSPEND` to postpone a pending chunk move.

Resuming chunk moves: Use `ALTER MOVE -RESUME` to place specified chunk moves that are suspended or stalled (in the "move failed" state) into the "scheduled" state, effectively rescheduling them.

Canceling chunk moves: Use `ALTER MOVE -CANCEL` to cancel pending chunk moves. Once a chunk move operation is canceled it cannot be resumed or suspended.

Use the `-chunk` and `-shard` options to filter the scheduled chunk move operations. Use the `-chunk` option to target specific chunk IDs, and use the `-shard` option to target all chunk moves scheduled to occur to and from the specified database.

You can use the `CONFIG CHUNKS` command to get a list of pending chunk moves.

Chunk moves that are already in process cannot be suspended or canceled. If any chunk in scope for the `ALTER MOVE` operation is already in any state other than "scheduled" a warning is returned indicating that the move operation for that chunk was not altered.

Examples

Suspend all pending chunk moves:

```
ALTER MOVE -SUSPEND
```

Resume (reschedule) move operations for chunks 3 and 4:

```
ALTER MOVE -RESUME -CHUNK 3,4
```

Cancel all pending chunk moves to and from shard SH_1:

```
ALTER MOVE -CANCEL -SHARD SH_1
```

alter task

Suspend, resume, or cancel scheduled chunk or replication unit management operations on a sharded database.

Syntax

```
ALTER TASK
    {-RESUME|-SUSPEND|-CANCEL}
    [ -TASK task | [[-CHUNK chunk_id_list] | [-RU ru_lst] | [-SHARD
shard]]]
    [-verbose]]
```

Options

Table 2-17 GDSCTL alter task Options

| Option | Description |
|----------|---|
| -cancel | Removes chunks for the specified scope from the scheduled list. "-chunk" specifies that all listed chunks will be removed, "-shard" specifies that all chunk moves to/from this database will be removed. If any chunk in scope is already in a move a warning is returned indicating that the chunk was not removed. |
| -chunk | List of numeric chunk identifiers or ALL for all chunks. |
| -resume | Restarts stalled move process, optional parameter "-shard" provides a list of databases that will have their "move failed" flags reset before move restarts. |
| -ru | A comma separated list of replication units. |
| -shard | A comma separated list of shards. |
| -suspend | Suspends move for the specified scope. -chunk specifies chunks to suspend, "-shard specifies that all chunk moves to/from that database will be suspended. If any chunk in scope is already in a move (any state other than "scheduled") a warning is returned indicating that move for that chunk was not suspended. |
| -task | A comma separated list of tasks. |
| -verbose | Enable verbose output mode. |

Usage Notes

RESUME option is used to restart stalled task. SUSPEND is used to postpone task, and CANCEL cancels task.

CHUNK, RU or SHARD is used to filter the list of scheduled migration. Use CONFIG TASK command to retrieve the list of scheduled tasks.

Examples

Suspend all pending tasks:

```
GDSCTL> alter task -suspend
```

Delete all tasks involving (ru 3 or 4) AND (shards shard1 or shard2):

```
GDSCTL> alter task -cancel -ru 3,4 -shard shard1,shard2
```

Suspend any task involving ru 5 in any way:

```
GDSCTL> alter task -suspend -ru 5
```

Resumes specified tasks:

```
GDSCTL> alter task -resume -task 34,35,36
```

config

Displays the configuration data for all components defined for the configuration.

Syntax

```
config [-support] [-verbose]
```

Options

Table 2-18 GDSCTL config Options

| Option | Description |
|----------|---|
| -support | If specified, GDSCTL output displays additional information for support purposes. |
| -verbose | Enable verbose mode. |

Usage Notes

When using the command, it does not matter if the components (except for the catalog database) are started. The configuration data displayed is retrieved from the catalog database.

Example

Display the configuration data for all components defined for the configuration.

```
GDSCTL> config
```

config backup

Configure Sharded Database (SDB) Backup

Syntax

```
config backup -rccatalog rc_connstr
               [-target (PRIMARY|STANDBY)]
               [(-destination (ALL|CATALOG|
<shard_list>):<deviceconfig>:<channelconfig>)+]
               [-frequency #level0_days|[#level0_days]:[#level1-days]]
               [-incremental (DIFFERENTIAL|CUMULATIVE)]
               [-retention #recovery_window_days]
               [-cdb sc_cdb_connstr] [-catpwd password]
               [-shard shard_list] [-async]
               [-encryption encryption][-REMOVE]
```

Options

Table 2-19 GDSCTL config backup Options

| Option | Description |
|--|---|
| -rccatalog rc_connstr | A connection string to the recovery catalog database. |
| -target (PRIMARY STANDBY) | For shards in Data Guard configurations, database backup can be done at the primary or the standby. The default is the standby. The value for this option is either PRIMARY or STANDBY. |
| -destination {ALL CATALOG shard_list:deviceconfig:channelcon fig} | Definition of device types and channels for target databases. It consists of three components: <i>shard-list</i> , <i>deviceconfig</i> and <i>channelconfig</i> . <ul style="list-style-type: none"> <i>shard-list</i>: specifies a comma separated list of shard identifiers. They can be shard space, shard group or shard names. <i>deviceconfig</i>: configures a device type for the prefixed target databases. It must be a valid device type configuration statement for RMAN. <i>channelconfig</i>: configures a channel for the prefixed target databases. It must be a valid channel configuration statement for RMAN. |
| -starttime ALL CATALOG shard_list:hh:mm | Backup start time for individual shards and the sharded database (SDB) catalog database. It must be a local time in a day in the time zone where the target database is located and specified in 24-hour format. The smallest unit for the time is minute. For example, CATALOG:22:30 specifies that the scheduled backup for the SDB catalog database should be started at 10:30 PM. The default backup start time for a target database is the midnight (00:00). Internally the start time is converted into a UTC time before it's passed to DBMS Scheduler to be set as the job start time |

Table 2-19 (Cont.) GDSCTL config backup Options

| Option | Description |
|--|--|
| <code>-frequency #level0_days #level0_days:#level1_days</code> | Backup repeat intervals for incremental level 0 and level 1 backups in days. The first number is the interval for incremental level 0 backups and the second is that for level 1 backups. If a single number is specified without a following comma, it defines the interval for level 0 backups. If a comma appears in the parameter, then the number to its left is the backup interval for level 0 backups and that to its right is the interval for level 1 backups, but both numbers are optional in this case. The default intervals for level 0 and level 1 backups are respectively 7 and 1 day. |
| <code>-incremental DIFFERENTIAL CUMULATIVE</code> | Incremental level 1 backup type to either <code>DIFFERENTIAL</code> or <code>CUMULATIVE</code> . The default is <code>DIFFERENTIAL</code> . |
| <code>-retention #recovery_window_days</code> | A recovery window for backup files. It must be a positive integer and specified in days. The default is 30 days. |
| <code>-cdb sc_cdb_connstr</code> | Required if the SDB catalog database is a PDB. It specifies a connect string for the container database of the SDB catalog database. |
| <code>-catpwd password</code> | Password for user <code>GSMCATUSER</code> . Prompted if not specified. This password only needs to be specified once for this command in an entire GDSCTL session. |
| <code>-shard shard_list</code> | <code>shard_list</code> specifies a comma separated list of shard identifiers. They can be shard space, shard group or shard names. If the same name is used for a shard space, shard group or shard, shard space takes the highest precedence followed by shard group and then shard. There are two special words for this option: <code>ALL</code> and <code>CATALOG</code> . <code>ALL</code> represents the SDB catalog database and all the shards in the SDB while <code>CATALOG</code> represents only the SDB catalog database. The default is <code>all shards</code> . |
| <code>-async</code> | When specified, all tasks to configure the backup for the shards will run in background. By default, the task will run in foreground. The task for the SDB catalog database will always run in foreground regardless of this flag setting. |
| <code>-encryption</code> | Encryption protocol for Advanced Network Option (ANO) used between GSM, GDSCTL and databases. <code>OFF</code> means that ANO is disabled. (<code>AES256</code> <code>AES192</code> <code>OFF</code>). |
| <code>-remov</code> | If specified, it removes the backup configuration from the specified shards. Other provided options for the command are ignored. |

Examples

The following example configures a backup channel of `DISK` type for the SDB catalog database, two parallel channels of `DISK` type for each of the shards (shard spaces `db1` and `db2` are used in the shard list), the backup retention window to 14 days, the level 0 and level 1 incremental backup repeat intervals to 7 and 1 day and the backup start time to 12:00 AM, leaving the incremental backup type to the default `DIFFERENTIAL` and the backup target type to the default `STANDBY`.

```
GDSCTL> config backup -rccatalog radmin/rman@inst6 -destination
"CATALOG::configure channel device type disk format '/tmp/rman/backups/
```

```
%d_%U'" -destination "dbs1,dbs2:configure device type disk parallelism
2:configure
channel 1 device type disk format '/tmp/rman/backups/1/%U';configure channel
2
device type disk format '/tmp/rman/backups/2/%U'" -starttime ALL:00:00 -
retention 14 -frequency 7:1 -catpwd gsm -cdb gsm_admin/gsm@instroot1;
```

When CONFIG BACKUP is not provided with any parameters, it shows the current backup configuration.

```
GDSCTL> config backup
Recovery catalog database user: radmin
Recovery catalog database connect descriptor:
(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=den02qxr)(PORT=1521))
(CONNECT_DATA=(SERVICE_NAME=cdb6_pdb1.regress.rdbms.dev.example.com)))
Catalog database root container user: gsm_admin
Catalog database root container connect descriptor:
(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=den02qxr)(PORT=1521))
(CONNECT_DATA=(SERVICE_NAME=v1908.regress.rdbms.dev.example.com)))
Backup retention policy in days: 14
Level 0 incremental backup repeat interval in minutes: 10080
Level 1 incremental backup repeat interval in minutes: 1440
Level 1 incremental backup type : DIFFERENTIAL
Backup target type: STANDBY
Backup destinations:
catalog::channel device type disk format '/tmp/rman/backups/%d_%u'
dbs1,dbs2:device type disk parallelism 2:channel 1 device type disk format
'/tmp/rman/backups/1/%u';channel 2 device type disk format '/tmp/rman/
backups/2/%u'
catalog::configure channel device type disk format '/tmp/rman/backups/%d_%u'
dbs1,dbs2:configure device type disk parallelism 2:configure channel 1 device
type disk format '/tmp/rman/backups/1/%u';configure channel 2 device type
disk format '/tmp/rman/backups/2/%u'
Backup start times:
all:00:00
```

config cdb

Displays properties of a specified CDB.

Syntax

```
config cdb [-cdb cdb_name] [-support][-verbose]
```

Options

Table 2-20 GDSCTL config cdb Options

| Option | Description |
|----------------------|--|
| -cdb <i>cdb_name</i> | Specify the name of the cdb. |
| -support | If specified, GDSCTL output displays additional information. |

Table 2-20 (Cont.) GDSCTL config cdb Options

| Option | Description |
|----------|----------------------|
| -verbose | Enable verbose mode. |

Examples

Display information about CDB called cdb1.

```
GDSCTL> config cdb -cdb cdb1
```

```
Name: tstdsbyb
Connection string: (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=cdb1host)
(PORT=1521))
(CONNECT_DATA=(SERVICE_NAME=cdb1.example.com)))
SCAN address:
ONS remote port: 0
Disk Threshold, ms: 20
CPU Threshold, %: 75
Version: 18.0.0.0
Rack:
```

config chunks

Displays properties of a specified chunk.

Syntax

```
config chunks [-support]
               ( [-shard shd] | [-shardgroup sh] | [-show_reshard] | [-
cross_shard] )
               ( [-chunk chunk_id] | [-key key [-superkey superkey] ] )
               [-table_family table family][-verbose]
```

Options**Table 2-21 GDSCTL config chunks Options**

| Option | Description |
|---------------------------|--|
| -chunk <i>chunk_id</i> | Specify a numeric chunk ID. |
| -cross_shard | Show cross-shard placement. |
| -key <i>key</i> | Sharding key |
| -shard <i>shd</i> | The name of the shard. |
| -shardgroup <i>sh</i> | The name of the shardgroup. |
| -show_reshard | Display information about ongoing chunk management operations. |
| -superkey <i>superkey</i> | Sharding super key. This is only needed for the composite sharding method. |

Table 2-21 (Cont.) GDSCTL config chunks Options

| Option | Description |
|---------------|---|
| -support | If specified, GDSCTL output displays additional information. |
| -table_family | Table family name in the form of <schema name>.<root table name>. |
| -verbose | Enable verbose mode. |

Usage Notes

The `config chunks` command lists all of the database shards and the chunks that they contain. Some chunks are listed more than once if there are standbys that contain replicated chunks.

If key or superkey type is date or timestamp then `gdctl config chunks -key/superkey` should be passed in the format as shown below:

```
"YYYY-MM-DD HH24:MI:SS.FF6"
```

It must be in this order in terms of year/month/date. The example below shows key/super key with DATE type:

```
gdctl config chunks -key '2000-01-01'
```

For key/super key with timestamp type, pass the value in single quotes:

```
gdctl config chunks -key 1281 -superkey '"1992-04-09 05:00:00.0"'
```

Examples

The output from `config chunks` is shown below.

```
GDSCTL> config chunks
```

```
Chunks
-----
Database                From      To
-----
sh1a                    1         10
sh1b                    1         10
```

config credential

Displays remote credentials currently available for shard jobs.

Syntax

```
config credential [-support] [-verbose]
```

Options

Table 2-22 GDSCTL config credential Options

| Option | Description |
|----------|--|
| -support | If specified, GDSCTL output displays additional information. |
| -verbose | Enable verbose mode |

Usage Notes

This command displays all existing remote credentials that can be used to execute sharding jobs.

Examples

Display credentials.

```
GDSCTL> config credential
```

```
Name                Username Windows domain
-----
CREDENTIAL_ONE      OraUser
CREDENTIAL_TWO      OraUser2
```

config database

Displays the static configuration data stored in the catalog for the specified database.

Syntax

```
config database [-support]
                [-database db_name] [-verbose]
```

Options

Table 2-23 GDSCTL config database Options

| Syntax | Description |
|--------------------------|---|
| -database <i>db_name</i> | Specify the name of a database. If you do not specify a database name, then GDSCTL displays the configuration data for all databases in the Global Data Services configuration. |
| -support | GDSCTL output displays additional support information. |
| -verbose | Enable verbose mode |

Usage Notes

You must connect to the catalog database as a user with the pool administrator privileges, using the [connect](#) command before running this command

Examples

Display the static configuration data stored in the catalog for all the databases in the Global Data Services configuration.

```
GDSCTL> config database
```

The `gdsctl config database` command returns information similar to the following:

| Name | Pool | Status | Region |
|--------|-------|--------|--------|
| ----- | ----- | ----- | ----- |
| dbcat | sales | Ok | east |
| dbcat1 | sales | Ok | west |
| dbcat3 | sales | Ok | west |

config file

Displays file objects currently available that can be specified in GDSCTL commands.

Syntax

```
config file [-support]
            [-file file_name] [-verbose]
```

Options

Table 2-24 GDSCTL config file Options

| Option | Description |
|-------------------------------------|--|
| <code>-file <i>file_name</i></code> | The name of the file object. |
| <code>-support</code> | If specified, GDSCTL output displays additional information. |
| <code>-verbose</code> | Enable verbose mode |

Usage Notes

If the specified file object does not exist, the command returns an error.

Example

Display the list of files defined in the catalog database.

```
GDSCTL> config file
Name
-----
dbcfg1
```

config gdspool

Displays the static configuration data that is stored in the catalog for the specified database pool.

Syntax

```
config gdspool [-support]
               [-gdspool gdspool_name] [-verbose]
```

Options

Table 2-25 GDSCTL config gdspool Options

| Syntax | Description |
|---|--|
| <code>-gdspool <i>gdspool_name</i></code> | Specify the name of a database pool. If you do not specify a database pool name, then GDSCTL displays the configuration data for all database pools. |
| <code>-support</code> | GDSCTL output displays additional support information. |
| <code>-verbose</code> | Enable verbose mode |

Usage Notes

You must connect to the catalog database as a user with the pool administrator privileges, using the [connect](#) command before running this command

Example

Display the static configuration data stored in the catalog for all Global Data Services pools.

```
GDSCTL> config gdspool
```

The `gdsctl config gdspool` command returns output similar to the following:

```
Name          Broker
-----
dbpoolora     No
mkt           No
sales         No
marketing     No
```

The following command shows the configuration detail of Global Data Services pool `marketing`.

```
GDSCTL> config gdspool -gdspool marketing
```

The above example returns output similar to the following:

```
GDS Pool administrators
-----
```

```

Databases
-----
dbcat2
dbcat1
dbcat3

Services
-----
sales_report
sales_analysis
sales_estimation
sales_peragent
sales_global

```

config gsm

Displays the static configuration data stored in the catalog for the specified global service manager.

Syntax

```

config gsm [-gsm gsm_name]
           [-support]
           [-verbose]

```

Options

Table 2-26 GDSCTL config gsm Options

| Syntax | Description |
|-----------------------------------|---|
| <code>-gsm <i>gsm_name</i></code> | Specify the name of a global service manager. If you do not specify a global service manager name, then GDSCTL displays the static configuration data for all global service managers in the cloud. |
| <code>-support</code> | If specified, GDSCTL output displays additional information. |
| <code>-verbose</code> | Enable verbose mode |

Usage Notes

You must connect to the catalog database as a user with the Global Data Services administrator privileges, using the [connect](#) command before running this command

Example

Display the static configuration data stored in the catalog for the global service manager `mygsm`:

```
GDSCTL> config gsm -gsm mygsm
```

The `gdsctl config gsm` command returns output similar to the following:

```
Name: mygsm
Endpoint 1: (ADDRESS=(HOST=stcal.us.hq.com)(PORT=1523)(PROTOCOL=tcp))
Endpoint 2: (ADDRESS=(HOST=stcal.us.hq.com)(PORT=1523)(PROTOCOL=tcp))
Local ONS port: 6123
Remote ONS port: 6234
Region: east
Buddy
-----
```

config region

Displays the static configuration data for the specified region.

Syntax

```
config region [-region region_name]
              [-support]
              [-verbose]
```

Options

Table 2-27 GDSCTL config region Options

| Syntax | Description |
|--------------------------------------|--|
| <code>-region <i>gsm_name</i></code> | Specify the name of a global service manager. |
| <code>-support</code> | If specified, GDSCTL output displays additional information. |
| <code>-verbose</code> | Enable verbose mode |

Example

Displays the static configuration data for the specified region.

```
GDSCTL> config region -region east
```

Displays the following output:

```
Name                Buddy
----                -
east
```

config sdb

Displays the static configuration data stored in the catalog for the sharded database.

Syntax

```
config sdb [-support] [-verbose]
```

Options

Table 2-28 GDSCTL config sdb Options

| Option | Description |
|----------|--|
| -support | If specified, GDSCTL output displays additional information. |
| -verbose | Enable verbose mode |

Examples

The output for `config sdb` is similar to the following.

```
GDSCTL> config sdb

GDS Pool administrators
-----

Replication Type
-----
Data Guard

Shard type
-----
System-managed

Shard spaces
-----
shardspaceora

Services
-----
oltp_ro_srvc
oltp_rw_srvc
```

config service

Displays the static configuration data stored in the Global Data Services catalog for the specified services that are located in a database pool.

Syntax

```
config service [-gdspool gdspool_name]
               [-service service_name]
               [-support]
               [-verbose]
```

Options

Table 2-29 GDSCTL config service Options

| Syntax | Description |
|---|--|
| <code>-gdspool <i>gdspool_name</i></code> | Specify the name of the database pool that contains the services. If the name is not specified, and there is only one <i>gdspool</i> with access granted to the user, it is used as the default <i>gdspool</i> . |
| <code>-service <i>service_name</i></code> | Specify a comma-delimited list of service names. If you do not use this option, then GDSCTL displays the configuration data for all services in the specified database pool. |
| <code>-support</code> | If specified, GDSCTL output displays additional information. |
| <code>-verbose</code> | Enable verbose mode |

Usage Notes

You must connect to the catalog database as a user with the pool administrator privileges, using the [connect](#) command before running this command

Examples

Show all the services in the user's Global Data Services pool:

```
GDSCTL> config service
```

The `gdsctl config service` command returns information similar to the following:

| Name | Network name | Pool | Started | Preferred | all |
|------------|-----------------------------|-------|---------|-----------|------|
| ---- | ----- | ---- | ----- | ----- | ---- |
| sales_svc1 | sales_svc1.sales.oradbcloud | sales | Yes | Yes | |
| sales_svc2 | sales_svc2.sales.oradbcloud | sales | NO | Yes | |
| sales_svc3 | sales_svc3.sales.oradbcloud | sales | Yes | Yes | |
| mkt_svc1 | mkt_svc1.mkt.oradbcloud | mkt | NO | Yes | |

Display the static configuration data stored in the Global Data Services catalog for sales:

```
GDSCTL> config service -service sales
```

```
Name: sales
Network name: sales.sdhdpool.oradbcloud
Pool: shdpool
Started: Yes
Preferred all: Yes
Locality: ANYWHERE
Region Failover: No
Role: NONE
Primary Failover: No
Lag: ANY
Runtime Balance: SERVICE_TIME
Connection Balance: LONG
```

```

Notification: Yes
TAF Policy: NONE
Policy: AUTOMATIC
DTP: No
Failover Method: NONE
Failover Type: NONE
Failover Retries:
Failover Delay:
Edition:
PDB:
Commit Outcome:
Retention Timeout:
Replay Initiation Timeout:
Session State Consistency:
SQL Translation Profile:
Stop option: NONE
Drain timeout:
Table Family: sales.customer

```

Supported services

```

-----
Database          Preferred Status
-----
shdb              Yes          Enabled
shdc              Yes          Enabled

```

config shard

Displays properties of a specified shard.

Syntax

```

config shard -shard shard_name
              [-support]
              [-verbose]

```

Options

Table 2-30 GDSCTL config shard Options

| Option | Description |
|--------------------------|--|
| -shard <i>shard_name</i> | Specify the name of the shard. |
| -support | GDSCTL output displays additional support information. |
| -verbose | Enable verbose mode |

Examples

```
GDSCTL> config shard
```

```

Name   Shard Group Status State   Region Availability
----   -
den17b dbs1      Ok     Deployed east   ONLINE
den17c dbs2      Ok     Deployed east   READ ONLY

```

The State column in the results can have the following values:

- **Created:** Indicates that `add shard` or `create shard` was run, but `deploy` has not yet been run for that shard.
- **Replicated:** Indicates that `deploy` was run and the Data Guard broker configuration was created. No other metadata (chunks, for example) are on the shard and the shard has not yet registered with the shard director
- **Sharded:** Indicates that the database has successfully registered with the shard director. Creates chunk metadata for new shards, but does not start any automatic rebalancing. To manually get from Replicated to Sharded and beyond, run `GDSCCTL sync -database <shard_name>`. This is what is happening internally in this step.
- **Deployed:** Indicates that all DDL catchup is completed and the shard is ready for operations. At this point, any scheduled chunk moves are begun in the background. A shard can be Deployed without having been rebalanced because rebalancing is a background operation.

config shardgroup

Displays properties of a specified shardgroup.

Syntax

```
config shardgroup [-shardgroup shardgroup_name]
                  [-support]
                  [-verbose]
```

Options

Table 2-31 GDSCCTL config shardgroup Options

| Option | Description |
|---|---|
| <code>-shardgroup <i>shardgroup_name</i></code> | Specify the name of the shardgroup. |
| <code>-support</code> | GDSCCTL output displays additional support information. |
| <code>-verbose</code> | Enable verbose mode |

Examples

The `config shardgroup` command generates the following output.

```
GDSCCTL> config shardgroup -shardgroup northeast
```

```
Shard Group Chunks Region Shard space
-----
dbs1        10     east  shd1
dbs2        10     east  shd1
```

By specifying a shardgroup, you get the following output.

```
GDSCCTL> config shardgroup -shardgroup dbs1
```

```
Shard Group: dbs1
```

```

Chunks: 10
Replicas:
Region: east
Shard space: shd1
Shards
-----
Shard Chunks
-----
den17b 10

```

config shardspace

Displays properties of a specified shardspace.

Syntax

```

config shardspace [-shardspace shardspace_name]
                  [-support]
                  [-verbose]

```

Options

Table 2-32 GDSCTL config shardspace Options

| Option | Description |
|------------------------------------|---|
| -shardspace <i>shardspace_name</i> | Specify the name of the shardspace. Optional for system-managed sharding. |
| -support | GDSCTL output displays additional support information. |
| -verbose | Enable verbose mode |

Usage Notes

The output varies depending on whether the command is issued on a shardspace configured in a user-defined SDB.

Examples

The `config shardspace` command generates the following output

```

GDSCTL> config shardspace

Shard space Chunks
-----
shd1          10

```

When a shardspace is specified, the output is returned in the following format.

```

GDSCTL> config shardspace -shardspace silver

Shard Group Region Role
-----
dbs1          east  Primary
dbs2          east  Standby
PROTECTION_MODE Chunks

```

```
-----
MaxProtection  10
```

config table family

Displays information about all table families in the sharded database.

Syntax

```
config table family
    [-verbose]
```

Examples

The config table family command generates the following output

```
GDSCTL> config table family

Schema  Name          ID      Shard Type
-----  -
sales   customer      1       System
hr       department    25      System
```

config task

Display chunk or replication unit management tasks and their statuses.

Syntax

```
CONFIG TASK
    [-oid oid_number]
    [-shard shard_name]
```

Options

Table 2-33 GDSCTL config task Options

| Option | Description |
|--------|--|
| -oid | Numeric object identifier (shard or replication unit). |
| -shard | The name of the shard. |

Examples

Display all tasks:

```
GDSCTL> CONFIG TASK
      task ID  status          GDS command
      -----  -
      43       started        switchover ru -ru 3 -database
cksrd3_ckshard3
```

```

          44      scheduled      switchover ru -ru 5 -database
cksr2_ckshard2
          45      scheduled      switchover ru -ru 6 -database
cksr3_ckshard3

```

Display task OID 43:

```

GDSCTL> CONFIG TASK -oid 43
      task ID  status      GDS command
      -----  -----  -----
          43      started      switchover ru -ru 3 -database
cksr3_ckshard3

```

Display tasks for shard cksrd2_ckshard2:

```

GDSCTL> CONFIG TASK -shard cksrd2_ckshard2
      task ID  status      GDS command
      -----  -----  -----
          44      scheduled      switchover ru -ru 5 -database
cksr2_ckshard2

```

config vncr

Displays the static configuration data stored in the catalog for valid node checking for registration (VNCR).

Syntax

```

config vncr [-group group_name]
            [-support]
            [-verbose]

```

Options

Table 2-34 GDSCTL config vncr Options

| Syntax | Description |
|---------------------------------------|--|
| <code>-group <i>group_name</i></code> | A group alias that defines a group of VNCRs. The same alias can be used in multiple ADD calls. |
| <code>-support</code> | GDSCTL output displays additional support information. |
| <code>-verbose</code> | Enable verbose mode |

Usage Notes

You must connect to the catalog database as a user with the pool administrator privileges, using the [connect](#) command before running this command

Example

The `config vncr` command returns information similar to the following:

```
GDSCTL> config vncr

Name          Group ID
-----
192.0.2.1     group_name
```

configure

Sets the GDSCTL parameters.

Syntax

```
configure [-gsmport port]
          [-timeout seconds]
          [-show]
          [-driver {THIN | OCI}]
          [-resolve {IP | HOSTNAME | QUAL_HOSTNAME}]
          [-log {ALL|OFF|INFO|FINE|FINER|FINEST|SEVERE|WARNING}]
          [-log_file log_file]
          [-gsm gsm_name]
          [-showtime ON|OFF]
          [-verbose ON|OFF]
          [-save_config]
          [-gsmdebug (1|0)]
          [-spool]
          [-width]
          [default_check_time]
          [echo]
          [-encryption encryption]
```

Options

Table 2-35 GDSCTL configure Options

| Syntax | Description |
|---|---|
| <code>-driver THIN OCI</code> | Oracle JDBC driver. |
| <code>-gsm <i>gsm_name</i></code> | Set current global service manager. |
| <code>-gsmdebug (1 0)</code> | Global service manager debug mode. |
| <code>-gsmport <i>port</i></code> | Default global service manager port. |
| <code>-log {ALL OFF INFO FINE FINER FINEST SEVERE WARNING}</code> | Set the logging level. The default is OFF. |
| <code>-log_file <i>log_file</i></code> | Set the location of the log file. The default is <code>\$TNS_ADMIN/GDSTL.log</code> . |
| <code>-resolve IP HOSTNAME QUAL_HOSTNAME</code> | Default host resolution for global service manager endpoint. |

Table 2-35 (Cont.) GDSCTL configure Options

| Syntax | Description |
|-------------------------|---|
| -save_config | Store configuration changes to GSM.ORA. |
| -show | Show the configuration. |
| -showtime ON OFF | Print time stamps. |
| -spool | Enable spooling. Warning: prints security-sensitive information to log file. |
| -timeout <i>seconds</i> | Global service manager requests timeout in seconds. |
| -verbose ON OFF | Enable or disable verbose output. The default value is ON. |
| -width | Console width in number of characters (default 80). |
| -default_check_time | Automatic check timeout |
| -echo | Turn echo ON/OFF |
| -encryption | Encryption protocol for Advanced Network Option (ANO) used between GSM, GDSCTL and databases. OFF means that ANO is disabled. (AES256 AES192 OFF). |

Example

Set the `mygsm` driver to OCI:

```
configure -driver OCI mygsm
```

connect

Specifies the credentials to administer a global service management environment. Credentials must be specified to perform certain operations using GDSCTL.

Syntax

```
connect [user_name[/password]]@connect_identifier
```

Options**Table 2-36 GDSCTL connect Options**

| Syntax | Description |
|---------------------------|---|
| <i>connect_identifier</i> | Specify an Oracle Net connect descriptor or a net service name that maps to a connect descriptor (for example, a list of global service managers). |
| <i>password</i> | Specify the password for the specified user. If you do not specify a password, then you are prompted to enter a password. The password is obscured when entered. |
| <i>user_name</i> | Specify the name of the user to connect as. The user that you specify must have either the Global Data Services administrator or the pool administrator role. If you specify no user name, then you are prompted for a user name. |

Usage Notes

You must connect to the catalog database as a user with either the Global Data Services administrator or the pool administrator privileges, depending on which command you want to run after you connect

Warning

Specifying a password as a connect command option is a security risk. You can avoid this risk by omitting the password, and entering it only when the system prompts for it.

Examples

Connect as the `gsmadmin` user to the private cloud:

```
GDSCTL> connect gsmadmin@mycloud
Enter password:
```

Connect using a connect descriptor, without specifying a user name and password:

```
GDSCTL> connect (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=myhost)(PORT=1521)))
Enter username:
```

copy ru

To instantiate or repair a follower member of a replication unit you can copy another follower in that replication unit from another shard.

Syntax

```
COPY [RU|REPLICATION_UNIT] -RU ru_id
      [-SOURCE source_shard_name] -TARGET target_shard_name
      [-REPLACE repl_shard_name]
      [-timeout timeout]
      [-force]
```

Options

Table 2-37 GDSCTL copy ru Options

| Option | Description |
|----------|--|
| -force | Allow the operation to bypass RAFT replication role checks. |
| -replace | Removes the replication unit follower specified by <code>-ru</code> from the database specified in <code>-replace</code> . |
| -ru | Replication unit ID |
| -source | Name of the source shard. |

Table 2-37 (Cont.) GDSCTL copy ru Options

| Option | Description |
|----------|---|
| -target | Name of the target shard . |
| -timeout | Timeout of connection retention between the time FAN event is sent to clients and chunk going read-only/down (seconds). |
| -verbose | Enable verbose mode. |

Usage Notes

The source shard and target shard shouldn't be the replica leader for the given replication unit. If a member of the specified replication unit exists on the target shard, it is replaced by full replica of replication unit from the source shard.

If `-source` is not specified, then an existing follower of the replication unit will be chosen automatically as the source shard.

If the target database doesn't already contain a member of the specified replication unit, then the total number of replicas for the given replication unit should be below replication factor, unless `-replace` is specified.

If `-replace` is specified, the replication unit member is removed from the specified database.

Examples

```
GDSCTL> copy ru -ru 1 -source sh1 -target sh2
```

create gds catalog

Creates a Global Data Services catalog for global service management in a specific database.

Syntax

```
create gds catalog -database db_name
                  [-user user_name[/password]]
                  [-region region_name_list]
                  [-gdspool gdspool_name_list]
                  [-configname confname]
                  [-autovncr {ON | OFF}]
                  [-force]
                  [-encryption encryption]
                  [-validate_network]
```

Options

Table 2-38 GDSCTL create gds catalog Options

| Syntax | Description |
|---|--|
| <code>-autovncr {ON OFF}</code> | This option enables (ON) or disables (OFF) autovncr mode. The default value is ON. See the Usage Notes below for important information about this option. |
| <code>-configname <i>confname</i></code> | Specify the name of the GDS configuration. The default configuration name is ORADBCLOUD. The configuration name can be up to 32 bytes long and can contain an alphabetical character followed by zero or more alphanumeric ASCII characters, '_', or '#' and possibly separated by periods if there are multiple identifiers. |
| <code>-database <i>db_name</i></code> | Specify the connect identifier for the database in which you want to create catalog. |
| <code>-force</code> | Rewrites existing global service manager configuration on catalog database. |
| <code>-gdspool <i>gdspool_name_list</i></code> | Specify a comma-delimited list of database pool names. When you use this option, the specified database pools are created as part of the catalog creation. If you do not specify this option, then GDSCTL creates a default database pool named DBPOOLORA. |
| <code>-region <i>region_name_list</i></code> | Specify a comma-delimited list of region names. This command creates each region and adds the regions to the catalog. If you do not specify a region, then a default region named REGIONORA is created. |
| <code>-user <i>user_name</i>[/<i>password</i>]</code> | Specify a user (and optionally, the password) that has the Global Data Services administrator privileges on the catalog database. If you do not use this option, then GDSCTL prompts you for the name and the password of a user with Global Data Services administrator privileges. If you specify a user name but not the password for the user, then GDSCTL prompts you for the password. |
| <code>-encryption</code> | Encryption protocol for Advanced Network Option (ANO) used between GSM, GDSCTL and databases. OFF means that ANO is disabled. (AES256 AES192 OFF). |
| <code>-validate_network</code> | This flag enables several network validation checks, including checking network connectivity between hosts and checking VNCR entries are valid. |

Usage Notes

The `create gds catalog` command generates a pair of PKI public and private keys and stores them in the catalog, along with a fixed string "GSM" that is encrypted with a private key. It uses the `GSMCATUSER` password.

You must have the Global Data Services administrator privileges on the computer where you want to create the Global Data Services catalog.

Auto VNCR is best used in environments with simple private networks where ease of configuration is the most important consideration. To have the highest level of control over

which hosts can participate in a GDS configuration, disable Auto VNCR and explicitly add the IP addresses of each database host to the VNCR configuration.

When `-autovncr` is enabled, Oracle attempts to find the host name of the target database when it is validated during `add shard` or `add database` execution. This host is then automatically added to the VNCR list in the catalog as an invited node. This mechanism is not compatible with all network configurations and may not work in the following cases:

- The catalog or global service manager host does not know how to translate the host name discovered on the target database host to a real IP address. This can happen if they have different names in the hosts file or DNS on the catalog or global service manager host, or if they just do not exist on those hosts.
- The target database host has multiple public network addresses, and Oracle chooses an address that is different than the address used when the database registers with the global service manager. This can happen if the host has multiple network cards or has configured virtual network interfaces.
- The database is running Oracle RAC, and other Oracle RAC instances run on a different subnet. This is not a recommended configuration for Oracle RAC. Refer to the Oracle RAC documentation for recommended configurations. With Oracle RAC, Oracle Database connects to a single database host to validate the target, and returns a subnet mask that includes the entire subnet that the host is on. If other instances are on a different subnet, they have no valid VNCR entry, and registration is rejected.

When `-autoVNCR` is not enabled, or, if any of the above cases apply, new hosts should be added manually using [add invitednode \(add invitedsubnet\)](#).

Example

Create a Global Data Services catalog for global service management in the database named DB1. Also create the regions EAST and WEST, and the database pool READERFARM.

```
GDSCTL> create gdscatalog -database db1 -region west,east -gdspool readerfarm
```

create restorepoint

Create Global Restore Points.

Syntax

```
create restorepoint [-name restore_point_name]
```

Options

Table 2-39 GDSCTL create restorepoint Options

| Syntax | Description |
|--|---|
| <code>-name <i>restore_point_name</i></code> | The name of the global restore point to create in the sharded database. If it is not provided, a name is generated for the restore point. |

Examples

The example below creates a global restore point in the sharded database with name GRP_MANUAL_1.

```
GGDSCTL> CREATE RESTOREPOINT -NAME GRP_MANUAL_1
```

create shardcatalog

Creates a shard catalog for the sharded database.

Syntax

```
create shardcatalog -database connect_identifier
                    [-user username[/password]]
                    [-region region_name_list]
                    [-configname config_name]
                    [-autovncr {ON | OFF}]
                    [-force]
                    [-sdb sdb_name]
                    [-shardspace shardspace_name_list]
                    [-agent_password password]
                    [-repl DG]                                [-sharding
{system | composite | user}}                               [-chunks number]
                    [-protectmode dg_protection_mode]
                    [-agent_port port]
                    [-for_federated_database] [-encryption encryption]
                    [-repunits repunits]
                    [-encryption encryption]
                    [-validate_network]
```

Options

Table 2-40 GDSCTL create shardcatalog options

| Command Option | Description |
|---------------------------------|---|
| -agent_password <i>password</i> | Specify the password to be used for remote scheduler agent registration with the catalog database. |
| -agent_port <i>port</i> | Port number for XDB to use. If NULL and no current value is set, then it will default to 8080. Execute on catalog as well. |
| -autovncr {ON OFF} | This option enables (ON) or disables (OFF) auto VNCR mode. The default value is ON. See the Usage Notes below for important information about this option. |

Table 2-40 (Cont.) GDSCTL create shardcatalog options

| Command Option | Description |
|---|--|
| <code>-chunks <i>number</i></code> | <p>Specify the default number of unique chunks in a shardspace. The value of <code>-chunks</code> must be greater than 2 times the size of the largest shardgroup in any shardspace.</p> <p>This parameter does not apply to user-defined sharding or a federated database. It will apply to all shardspaces created for composite sharding if the number of chunks is not specified in the <code>ADD SHARDSPACE</code> command.</p> <p>All shardgroups in a shardspace have the same number of chunks. If this parameter is not specified, the default number of chunks is determined at the time of execution of the first <code>DEPLOY</code> command and is 120 per database of the shardgroup with the biggest number of shards</p> |
| <code>-configname <i>config_name</i></code> | <p>Specify the name of the GDS configuration. This name is used as the virtual <code>DB_DOMAIN</code> of the sharded database. The default configuration name is <code>ORADBCLLOUD</code>.</p> <p>The configuration name can be up to 32 bytes long and can contain an alphabetical character followed by zero or more alphanumeric ASCII characters, <code>'_'</code>, or <code>'#'</code> and possibly separated by periods if there are multiple identifiers.</p> |
| <code>-database <i>connect_identifier</i></code> | Specify the connect identifier for the database in which you want to create the catalog. |
| <code>-for_federated_database</code> | Create a catalog for a federated database. |
| <code>-force</code> | Before creating the new catalog, delete an existing shard or GDS catalog on this database. |
| <code>-protectmode <i>dg_protection_mode</i></code> | Specify the Data Guard protection mode: <code>MAXPROTECTION</code> , <code>MAXAVAILABILITY</code> , or <code>MAXPERFORMANCE</code> (default). For Data Guard replication this parameter applies to any shardspace created without specification of protection mode in the <code>ADD SHARDSPACE</code> command. |
| <code>-region <i>region_name_list</i></code> | Specify a comma-delimited list of region names. This command creates each region and adds the regions to the catalog. If you do not specify a region, then a default region named <code>REGIONORA</code> is created. |
| <code>-repl <i>DG NATIVE</i></code> | Specify the technology used to replicate data in the sharded database. Only one value can be specified for this parameter: <code>DG</code> for Data Guard and <code>NATIVE</code> for native RAFT replication. The default value is <code>DG</code> . This parameter cannot be modified after the catalog has been created. This parameter cannot be modified after the catalog has been created. |

Table 2-40 (Cont.) GDSCTL create shardcatalog options

| Command Option | Description |
|--|--|
| <code>-sdb <i>sdb_name</i></code> | Specify the virtual DB_UNIQUE_NAME for the sharded database. The default name is ORASDB. The sharded database (SDB) name can be up to 30 characters long and can be an alphabetical character followed by zero or more alphanumeric ASCII characters or an underscore (_). |
| <code>-sharding {system composite user}</code> | Specify the sharding type: SYSTEM for system-managed (default), USER for user-defined, and COMPOSITE. This parameter cannot be modified after the catalog has been created. Oracle GoldenGate does not support the user-defined sharding method. |
| <code>-repunits <i>reunits</i></code> | Total number of replication units (this setting only applies to native RAFT replication). By default, Oracle Sharding determines the number of replication units (RUs) in a shardspace and the number of chunks in an RU. Note that in system-managed sharding there is one shardspace named SHARDSPACEORA. |
| <code>-shardspace <i>shardspace_name_list</i></code> | Specify a comma-delimited list of shardspace names. This option creates specified shardspaces and adds them to the catalog. If you do not specify a shardspace, then a default shardspace named SHARDSPACEORA is created. |
| <code>-user <i>username[/password]</i></code> | Specify a user (and optionally, the password) that has the administrator privileges on the catalog database. If you do not use this option, then GDSCTL prompts you for the name and the password of a user with administrator privileges. If you specify a user name but not the password for the user, then GDSCTL prompts you for the password. |
| <code>-encryption</code> | Encryption protocol for Advanced Network Option (ANO) used between GSM, GDSCTL and databases. OFF means that ANO is disabled. (AES256 AES192 OFF). |
| <code>-validate_network</code> | This flag enables several network validation checks, including checking network connectivity between hosts and checking VNCR entries are valid. |

Usage Notes

The `create shardcatalog` command creates a GDS catalog specifically designed for a sharded database (SDB). This command cannot be used to create a conventional GDS catalog. Execution of this command is the first step required to create an SDB. The command is executed by the user with the GDS administrator or SYSDBA privileges.

Keep the following points in mind when using `create shardcatalog`:

- Only a single sharded database can be created using a shard catalog. A shard catalog cannot be used for a regular GDS configuration.

- Any arbitrary password can be specified for remote agent registration. If one is stipulated and an agent registration password already exists, it will be overridden with the new password. In order to successfully execute the `GDSCTL CREATE SHARD` command, an agent password must be set using `CREATE SHARDCATALOG` or `MODIFY CATALOG`.
- `CHUNKS` defines the default number of unique chunks in a shardspace. It is applied to all shardspaces created for composite sharding if the number of chunks is not specified in the `ADD SHARDSPACE` command.
- This command creates each region and adds the regions to the catalog. If you do not specify a region, then a default region named `REGIONORA` is created.
- The default replication factor specified by `REPFACOR` can be overridden for a particular shardgroup by specifying the replication factor in the corresponding `ADD SHARDGROUP` command. For automatically created default shardgroups the parameter cannot be overridden. A non-default shardgroup must be created to customize the replication factor.
- The `SHARDSPACE` option creates specified shardspaces and adds them to the catalog. If you do not specify a shardspace, then a default shardspace named `SHARDSPACEORA` is created.
- For Data Guard replication the `PROTECTMODE` parameter applies to any shardspace created without specification of protection mode in the `ADD SHARDSPACE` command.
- The `FOR_FEDERATED_DATABASE` option is mutually exclusive with the `SHARDING` option.
- Auto VNCR is best used in environments with simple private networks where ease of configuration is the most important consideration. To have the highest level of control over which hosts may participate in a GDS configuration, disable Auto VNCR and explicitly add the IP addresses of each database host to the VNCR configuration.

When `-autovnocr` is enabled, Oracle attempts to find the host name of the target shard when it is validated during `add shard` execution. This host is then automatically added to the VNCR list in the catalog as an invited node. This mechanism is not compatible with all network configurations and may not work in the following cases:

- The catalog or global service manager host does not know how to translate the host name discovered on the target shard host to an real IP address. This can happen if they have different names in the hosts file or DNS on the catalog or global service manager host, or if they just do not exist on those hosts.
- The target shard host has multiple public network addresses, and Oracle chooses an address that is different than the address used when the shard registers with the global service manager. This can happen if the host has multiple network cards or has configured virtual network interfaces.
- The shard is running Oracle RAC, and other Oracle RAC instances run on a different subnet. This is not a recommended configuration for Oracle RAC. Refer to the Oracle RAC documentation for recommended configurations. With Oracle RAC, Oracle Database connects to a single shard host to validate the target, and returns a subnet mask that includes the entire subnet that the host is on. If other instances are on a different subnet, they have no valid VNCR entry, and registration is rejected.

When `-autoVNCR` is not enabled, or, if any of the above cases apply, new hosts should be added manually using [add invitednode \(add invitedsubnet\)](#).

Note: Oracle GoldenGate replication support for Oracle Sharding High Availability is deprecated in Oracle Database 21c.

Examples

The following example creates a shard catalog on the mydb database.

```
GDSCTL> CREATE SHARDCATALOG -DATABASE mydb
```

databases

Displays the status of all databases.

Syntax

```
{status database | databases} [-gsm gsm_name]  
                               [-database db_name]  
                               [-gdspool gdspool_name]  
                               [-raw|-support|-verbose]
```

Options

Table 2-41 GDSCTL databases Options

| Option | Description |
|------------------------------|--|
| -database <i>db_name</i> | Specify the name of the database on which you want to start the service. If you do not specify this option, then GDSCTL starts the services on all preferred databases. |
| -gdspool <i>gdspool_name</i> | Specify the name of the database pool in which the services you want to start are located. If not specified and there is only one <i>gdspool</i> with access granted to the user, it is used as the default <i>gdspool</i> . |
| -gsm <i>gsm_name</i> | Specify the name of a global service manager to check. If the name is not specified, then GDSCTL uses the current global service manager name for the session (specified with the GDSCTL <code>set gsm</code> command). |
| -raw | If specified, GDSCTL output is presented in a raw non-parsed format. |
| -support | If specified, GDSCTL output displays additional information. |
| -verbose | Enable verbose mode. |

Usage Notes

You must connect to the catalog database as a user with the pool administrator privileges, using the command [connect](#) before running this command.

Example

Display the status of all databases:

```
GDSCTL> databases
```

The databases command returns output similar to the following:

```
Database: "dbcat1" Registered: Y State: Ok ONS: N. Role: PRIMARY Instances: 1
Region: east

Service: "sales_svc2" Globally started: N Started: N
Scan: Y Enabled: Y Preferred: Y
Service: "sales_svc1" Globally started: Y Started: Y
Scan: N Enabled: Y Preferred: Y
Registered instances:
sales%11
Database: "dbcat2" Registered: Y State: Ok ONS: N. Role: PRIMARY Instances: 1
Region: east
Service: "sales_svc2" Globally started: N Started: N
Scan: Y Enabled: Y Preferred: Y
Service: "sales_svc1" Globally started: Y Started: Y
Scan: N Enabled: Y Preferred: Y
Registered instances:
sales%1
```

delete backup

Deletes sharded database (SDB) backups identified with specific tags from the recovery repository.

Syntax

```
delete backup [-tag tag_list] [-obsolete] [-catpwd password]
              [-shard shard_list] [-async]
```

Options

Table 2-42 GDSCTL delete backup Options

| Option | Description |
|--------------------------|--|
| tag <i>tag_list</i> | A comma separated list of tags. The backups identified by these tags will be deleted. |
| -obsolete | If specified, all obsoleted backups are deleted. |
| -catpwd <i>password</i> | Password for user GSMCATUSER. Prompted if not specified. It needs to be specified once for the entire GDSCTL session. |
| -shard <i>shard_list</i> | <i>shard_list</i> specifies a comma separated list of shard identifiers. They can be shard space, shard group or shard names. The default is no shards. |
| -async | When specified, all tasks to configure the backup for the shards will run in background. By default, the task will run in foreground. The task for the SDB catalog database will always run in foreground regardless of this flag setting. |

Examples

The following example deletes backups with tag odb_200414205057124_0400 from shard v1908d_cdb2_pdb1:

```
GDSCTL> delete backup -shard v1908d_cdb2_pdb1 -tag ODB_200414205057124_0400 -
catpwd gsm
```

This will delete identified backups, would you like to continue [No]?y

Deleting backups for database "v1908d_cdb2_pdb1" ...

```
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=231 device type=DISK
allocated channel: ORA_DISK_2
channel ORA_DISK_2: SID=467 device type=DISK
deleted backup piece
backup piece handle=/tmp/rman/backups/2/2hutl9e9 RECID=13 STAMP=1037739466
Deleted 1 objects
```

```
deleted backup piece
backup piece handle=/tmp/rman/backups/1/2iutl9ed RECID=14 STAMP=1037739469
Deleted 1 objects
```

Recovery Manager complete.

delete catalog

Deletes the specified catalog.

Syntax

```
delete catalog [-connect [user/[password]@]conn_str]
               [-force]
```

Options

Table 2-43 GDSCTL delete catalog Options

| Syntax | Description |
|----------|--|
| -connect | Specify an Oracle Net connect descriptor or a net service name that maps to a connect descriptor for the database (or shard). If you do not use this option, then GDSCTL deletes the Global Data Services catalog that is used by the global service manager associated with the current session. |
| -force | Silently remove GDS metadata. No warnings are shown. |

Usage Notes

You must have the Global Data Services administrator privileges on the computer where the database resides from which you want to delete the Global Data Services catalog

If `-connect` is not specified, the catalog that belongs to currently connected database (if any) is deleted.

Example

Delete the Global Data Services catalog located in the database named DB1.

```
GDSCTL> delete catalog -connect db1
```

deploy

Deploys the sharded database.

Syntax

```
deploy [-no_rebalance] [-validate_network][--timeout timeout]
```

Options

Table 2-44 GDSCTL deploy Options

| Option | Description |
|--------------------------------|---|
| <code>-no_rebalance</code> | Skip automatic chunk migration during incremental deploy. |
| <code>-timeout</code> | Timeout of connection retention between FAN is sent to clients and chunk going read-only/down (seconds). |
| <code>-validate_network</code> | This flag enables several network validation checks, including checking network connectivity between hosts and checking VNCR entries are valid. |

Usage Notes

This command is executed after one or more shards have been added to the shard catalog. As the result of the command execution, a certain range of data is associated with a newly added database. If a database is part of a Data Guard Broker configuration, a role (primary or standby) is assigned to it. Then replication and/or migration of data to from other databases to newly deployed databases are initiated.

- Deploy runs almost entirely in parallel, and mostly in the background, and will not deploy any shards which do not have all their counterparts in other shardgroups. All undeployed shards that can be deployed are deployed as the result of execution of this command.
- Before configuring replication, this command cross-checks parameters of all databases included into the replication configuration. An error is returned if the cross-check finds inconsistency or ambiguity, for example, no primary shardgroup in a shardspace with Data Guard replication.
- If a `CREATE SHARD` command had previously been issued, these new shards will be created during deployment and added to the shard catalog. If a shard needs to be created, `DEPLOY` runs a job for each database which requires a remote credential (see [add credential](#)). This credential must be valid at the time of deployment.

- The `NO_REBALANCE` option allows to skip automatic rebalancing of chunks across shards during incremental `DEPLOY`. Use the [move chunk](#) command to perform manual chunk migration.

Examples

Deploy the sharded database.

```
GDSCTL> deploy
```

disable backup

Disable Sharded Database (SDB) Backup Jobs.

Syntax

```
disable backup [-catpwd password] [-shard shard_list]
```

Options

Table 2-45 GDSCTL disable backup Options

| Syntax | Description |
|---------------------------------------|---|
| <code>-catpwd <i>password</i></code> | Password for user GSMCATUSER. Prompted if not specified. It needs to be specified once for the entire GDSCTL session. |
| <code>-shard <i>shard_list</i></code> | <i>shard-list</i> is a comma separated list of shard identifiers. They can be shard space, shard group or shard names. The default is <i>all shards</i> |

disable service

Disables specified global services.

Syntax

```
disable service [-gdspool gdspool_name]
                [-service service_name_list]
                [-database db_name | [-override -connect conn_str [-pwd
password]]]
```

Options

Table 2-46 GDSCTL disable service Options

| Syntax | Description |
|---------------------------------------|---|
| <code>-connect <i>conn_str</i></code> | An Oracle Net connect descriptor or net service name that resolves to a connect descriptor for the database (or shard). |
| <code>-database <i>db_name</i></code> | Specify the name of the database on which to the service is located. If you do not specify this option, then the service is disabled, globally. |

Table 2-46 (Cont.) GDSCTL disable service Options

| Syntax | Description |
|--|---|
| <code>-gdspool <i>gdspool_name</i></code> | Specify the database pool in which the services are located. If not specified and there is only one <code>gdspool</code> with access granted to the user, then this one is used as the default <code>gdspool</code> . |
| <code>-override</code> | Skip the GDS catalog (used when the GDS catalog is not available). |
| <code>-pwd</code> | The GSMUSER password. |
| <code>-service <i>service_name_list</i></code> | Specify a comma-delimited list of global service names. If you do not use <code>-service</code> to specify an individual global service or to specify a list of global services, then all the services in the database pool are disabled. |

Usage Notes

You must connect to the catalog database as a user with the pool administrator privileges, using the [connect](#) command before running this command

A running service cannot be disabled. If `-override` is specified, the command is executed without going to the GDS catalog. Use this option when the GDS catalog is unavailable. It is not recommended for use under normal operation.

Example

Disable and stop the service `G_SALES_REPORT` on all databases in the database pool `READERFARM`.

```
GDSCTL> disable service -gdspool readerfarm -service g_sales_report -database dbl
```

See Also

[#unique_138](#)

enable backup

Enable Sharded Database (SDB) Backup Jobs.

Syntax

```
enable backup [-catpwd password] [-shard shard_list]
```

Options

Table 2-47 GDSCTL enable backup Options

| Syntax | Description |
|---------------------------------------|---|
| <code>-catpwd <i>password</i></code> | Password for user GSMCATUSER. Prompted if not specified. It needs to be specified once for the entire GDSCTL session. |
| <code>-shard <i>shard_list</i></code> | <i>shard_list</i> is a comma separated list of shard identifiers. They can be shard space, shard group or shard names. The default is <i>all shards</i> |

Usage Notes

All backup jobs are initially disabled. They can be enabled by running the `enable database` command. They should be run after SDB backup has been configured with command `CONFIG BACKUP`. An error is reported if this command is run before SDB backup is configured.

enable service

Enables the specified global services.

Syntax

```
enable service [-gdspool gdspool_name]
               [-service service_name_list]
               [-database db_name][[-override -connect conn_str [-pwd password]]]
```

Options

Table 2-48 GDSCTL enable service Options

| Syntax | Description |
|---|---|
| <code>-connect <i>conn_str</i></code> | An Oracle Net connect descriptor or net service name that resolves to a connect descriptor for the database (or shard). |
| <code>-database <i>db_name</i></code> | Specify the name of the database on which the service is located. If you do not specify this option, then the service is enabled globally. |
| <code>-gdspool <i>gdspool_name</i></code> | Specify the GDS pool in which the services are located. If not specified and there is only one gdspool with access granted to the user, it is used as the default gdspool. |
| <code>-override</code> | Skip the GDS catalog (used when the GDS catalog is not available). |
| <code>-pwd</code> | The GSMUSER password. |
| <code>-service <i>service_name</i></code> | Specify a comma-delimited list of global service names. If you do not use <code>-service</code> to specify an individual global service or to specify a list of global services, then all the services in the database pool are disabled. |

Usage Notes

You must connect to the catalog database as a user with the pool administrator privileges, using the [connect](#) command before running this command

ENABLE SERVICE will start the global service if it is preferred_all or cardinality is not reached.

If -override is specified, the command is executed without going to the GDS catalog. Use this option when the GDS catalog is unavailable. It is not recommended for use under normal operation.

Example

Enable the service G_SALES_REPORT on the database DB1 in the database pool READERFARM.

```
GDSCTL> enable service -gdspool readerfarm -service g_sales_report -database db1
```

See Also

[#unique_139](#)

exit

Quit GDSCTL utility.

Syntax

```
quit | exit
```

export catalog

Saves the current catalog configuration to a local file.

Syntax

```
export catalog [-force] source
```

Options

Table 2-49 GDSCTL export catalog Options

| Syntax | Description |
|--------|---|
| -force | If not specified, export will be cancelled if there are ongoing GDS operations. |
| source | Name of a file on the same computer where the command is being executed. The configuration will be saved to this file. If the file already exists, it will be overwritten without a prompt. If the file is not writable (for example the path does not exist), you will get an error. |

Usage Notes

You must connect to the catalog database as a user with GDS Administrator privileges before running this command.

It is recommended that you validate the catalog, using the [validate catalog](#) command before exporting it.

Example

Save the catalog backup to your home directory.

```
GDSCTL> export catalog /home/user/cat-201307.backup
```

help

Provides a list of the GDSCTL commands supported in the current release. When followed by a command name, it returns the help page associated with the command.

Syntax

```
help [gdsctl_command]
```

Options

Table 2-50 GDSCTL help Options

| Option | Description |
|-----------------------|---|
| <i>gdsctl_command</i> | Enter any GDSCTL command name to return a help page with syntax, options, usage notes and examples. |

import catalog

Restores the catalog configuration from the specified file, created using export catalog command.

Syntax

```
import catalog [-database catalog_db_name]  
               [-catpwd gsmcatusrpwd]  
               [-user gsmadminname[/password]]  
               source
```

Options

Table 2-51 GDSCTL import catalog Options

| Syntax | Description |
|----------------------------------|---|
| <i>-catpwd gsmcatusrpwd</i> | GSMCATUSER password. |
| <i>-database catalog_db_name</i> | The connect identifier for the database in which to create catalog. |

Table 2-51 (Cont.) GDSCTL import catalog Options

| Syntax | Description |
|--|---|
| <i>source</i> | Name of a file on the same computer where the command is being executed. The configuration will be restored from this file. If the file is not readable, you will get an error. |
| <code>-user gsmadminname[/password]</code> | Credentials of the user that has the GDS administrator privileges on the catalog database. |

Usage Notes

If `-database` is not specified, the GDS catalog that the current global service manager is associated with will be used. The `-catpwd` option should be specified in case you need to perform cleanup of databases in the existing catalog that are not found in imported file.

When restoring to a new catalog database, catalog must be created first, using the [create gds catalog](#) command.

You must connect to the catalog database as a user with GDS Administrator privileges before running this command.

The import procedure can be considered finished only when there are no pending requests after import. Use the [config](#) command to get the list of pending requests.

Example

Load the catalog backup from your home directory.

```
GDSCTL> import catalog /home/user/cat-201307.backup
```

list backup

List Sharded Database (SDB) Backups

Syntax

```
list backup [-restorepoint restorepoint] [-controlfile]
           [-summary] [-catpwd password] [-shard shard_list]
```

Options**Table 2-52 GDSCTL list backup Options**

| Option | Description |
|--|---|
| <code>-restorepoint <i>restorepoint</i></code> | An SDB global restore point. If specified, only backups that are usable to restore the specified shards to the specific restore point are listed. Otherwise, all backups for the specified shards are listed. |
| <code>-controlfile</code> | If specified, only backups usable to restore database control files to a specific restore point are listed. |
| <code>-summary</code> | If specified, the backups are listed in summary format. |

Table 2-52 (Cont.) GDSCTL list backup Options

| Option | Description |
|--------------------------------|---|
| <code>-catpwd password</code> | Password for user GSMCATUSER. Prompted if not specified. This password only needs to be specified once for this command in an entire GDSCTL session. |
| <code>-shard shard_list</code> | <i>shard_list</i> specifies a comma separated list of shard identifiers. They can be shard space, shard group or shard names. The default is no shards. |

Examples

The following example shows how to list the backups of control files from the SDB catalog database in summary.

```
GDSCTL> list backup -shard catalog -controlfile -summary -catpwd gsm
```

Listing backups for database "v1908" ...

List of Backups

=====

| Key | TY | LV | S | Device | Type | Completion Time | #Pieces | #Copies | Compressed | Tag |
|--------------------|----|----|---|--------|------|-----------------|---------|---------|------------|-----|
| 1366 | B | F | A | DISK | | 13-APR-20 | 1 | 1 | NO | |
| TAG20200413T234608 | | | | | | | | | | |
| 1851 | B | F | A | DISK | | 14-APR-20 | 1 | 1 | NO | |
| TAG20200414T000333 | | | | | | | | | | |
| 1996 | B | F | A | DISK | | 14-APR-20 | 1 | 1 | NO | |
| TAG20200414T001446 | | | | | | | | | | |
| 2057 | B | F | A | DISK | | 14-APR-20 | 1 | 1 | NO | |
| TAG20200414T001519 | | | | | | | | | | |
| 2151 | B | F | A | DISK | | 14-APR-20 | 1 | 1 | NO | |
| TAG20200414T012934 | | | | | | | | | | |
| 3205 | B | F | A | DISK | | 14-APR-20 | 1 | 1 | NO | |
| TAG20200414T202822 | | | | | | | | | | |

Recovery Manager complete.

The next example shows the use of the command to list the backups from shard v1908b_cdb2_pdb1 recoverable to restore point backup_before_db_maintenance.

```
GGDSCTL> list backup -shard v1908b_cdb2_pdb1 -restorepoint
BACKUP_BEFORE_DB_MAINTENANCE
"GSMCATUSER" password:
```

Listing backups for database "v1908b_cdb2_pdb1" ...

List of Backup Sets

=====

| BS Key | Type | LV | Size | Device | Type | Elapsed Time | Completion Time |
|--|------|----|---------|--------|------|--------------|-----------------|
| 2998 | Incr | 0 | 265.53M | DISK | | 00:00:06 | 14-APR-20 |
| BP Key: 3009 Status: AVAILABLE Compressed: NO Tag: | | | | | | | |
| BACKUP_BEFORE_DB_MAINTENANCE | | | | | | | |

```

        Piece Name: /tmp/rman/backups/2/0sutl6oa
List of Datafiles in backup set 2998
File LV Type Ckp SCN    Ckp Time  Abs Fuz SCN Sparse Name
-----
11  0  Incr 2678401  14-APR-20          NO  /ade/b/3998875997/
oracle/dbs/cdb2_pdb1_db.f

BS Key  Type LV Size      Device Type Elapsed Time Completion Time
-----
2999    Incr 0  191.61M  DISK          00:00:04    14-APR-20
        BP Key: 3010    Status: AVAILABLE Compressed: NO  Tag:
BACKUP_BEFORE_DB_MAINTENANCE
        Piece Name: /tmp/rman/backups/1/0tutl6oh
List of Datafiles in backup set 2999
File LV Type Ckp SCN    Ckp Time  Abs Fuz SCN Sparse Name
-----
12  0  Incr 2678425  14-APR-20          NO  /ade/b/3998875997/
oracle/dbs/cdb2_pdb1_ax.f
13  0  Incr 2678425  14-APR-20          NO  /ade/b/3998875997/
oracle/dbs/cdb2_pdb1_xdb.f

Recovery Manager complete.

```

list restorepoint

List Global Restore Points.

Syntax

```
list restorepoint [-start_time t1] [-end_time t2] [-start_scn_ s1] [-end_scn
s2]
```

Options

Table 2-53 GDSCTL list restorepoint Options

| Syntax | Description |
|--------------------------------|---|
| <code>-start_time time1</code> | If specified, the command lists restore points that were created at or after this time. It must be specified in the format "YYYY-MM-DD HH:MM:SS[.FFF]" where .FFF is a fraction of a second in the precision of milliseconds. |
| <code>-end_time time2</code> | If specified, the command lists restore points that were created at or before this time. Refer to the option "start_time" above for its format. |
| <code>-start_scn scn1</code> | If specified, the command lists restore points with SCNs equal to or greater than this SCN. |
| <code>-end_scn scn2</code> | If specified, the command lists restore points with SCNs equal to or less than this SCN. |

Examples

The example below lists the available restore points in the sharded database (SDB) with the SCN between 2600000 and 2700000.

```
GDSCTL> list restorepoint -start_scn 2600000 -end_scn 2700000
Name                               SCN           Create
Time
GRP_200726222838427_0400          2601938      2020-07-26
22:28:39.0
GRP_200726232837677_0400          2613192      2020-07-26
23:28:38.0
GRP_200727002838026_0400          2624200      2020-07-27
00:28:38.0
GRP_200727012838351_0400          2634360      2020-07-27
01:28:38.0
GRP_200727022837961_0400          2645399      2020-07-27
02:28:38.0
GRP_200727032838402_0400          2654898      2020-07-27
03:28:39.0
GRP_200727042837648_0400          2664398      2020-07-27
04:28:38.0
GRP_200727052837932_0400          2673905      2020-07-27
05:28:38.0
GRP_200727062838321_0400          2683840      2020-07-27 06:28:38.0
```

modify catalog

Modifies the properties of the GDS catalog or shard catalog.

Syntax

```
modify catalog [-autovncr {ON | OFF}]
               [-oldpwd oldpassword -newpwd newpassword]
               [-pwd password -newkeys]
               [-agent_password password]
               [-agent_port port]
               [-region region]
               [-recover]
```

Options

Table 2-54 GDSCTL modify catalog Options

| Syntax | Description |
|--|--|
| <code>-agent_password <i>password</i></code> | Specify the agent registration password in the catalog for the remote Scheduler agent. |
| <code>-agent_port <i>port</i></code> | Port number for XDB to use. If it is NULL and no current value is set, then it will default to 8080. Execute on catalog as well. |
| <code>-autovncr {ON OFF}</code> | This option enables (ON) or disables (OFF) autovncr mode. The default value is ON. |

Table 2-54 (Cont.) GDSCTL modify catalog Options

| Syntax | Description |
|---|--|
| <code>-newkeys</code> | Generates a new PKI key pair. |
| <code>-newpwd <i>newpassword</i></code> | Used along with <code>-oldpwd</code> , sets the GSMCATUSER password after changing it on the catalog database. |
| <code>-oldpwd <i>oldpassword</i></code> | Used along with <code>-newpwd</code> , sets the GSMCATUSER password after changing it on the catalog database. |
| <code>-pwd <i>password</i></code> | Provides the GSMCATUSER password to generate the PKI keys when using <code>-newkeys</code> . |
| <code>-recover</code> | Perform catalog recovery to the last consistent state. |
| <code>-region <i>region</i></code> | Region that the database, catalog, shard, shardgroup, or global service manager belongs to. |

Usage Notes

To use this command, there must be a least one global service manager running and a connection with the catalog database must have already been established (see the [connect](#) command).

You must connect to the catalog database as a user with the Global Data Services administrator privileges, using the [connect](#) command before running this command.

Auto VNCR is best used in environments with simple private networks where ease of configuration is the most important consideration. To have the highest level of control over which hosts may participate in a GDS configuration, disable Auto VNCR and explicitly add the IP address(es) of each database host to the VNCR configuration.

The GSMCATUSER password should be updated regularly for security reasons. Use the following command to perform this operation.

```
modify catalog -oldpwd oldpassword -newpwd newpassword
```

This command fetches the encrypted private key and encryption string, decrypts them using the old password, re-encrypts them with the new password and stores them again.

If the GSMCATUSER password is changed, you must execute `MODIFY CATALOG` to update catalog security scheme, with `-newpwd` and `-oldpwd` specified.

The PKI keys must be updated regularly, which is done using `modify catalog -oldpwd oldpassword -newkeys`. This command generates a new PKI key pair and replaces the corresponding fields in the database.

If you decide to replace the PKI keys, or just after the patchset upgrade from Oracle Database 12c Release 1 (12.1) on the catalog database, run this command:

```
modify catalog -pwd ** -newkeys
```

An arbitrary password can be stipulated for remote agent registration. If an agent registration password already exists, it will be overridden with the new password. In order to successfully execute the `GDSCTL CREATE SHARD` command, an agent password must be set using the `CREATE SHARDCATALOG` or `MODIFY CATALOG` command.

Examples

Turn off autovncr mode for the catalog database.

```
connect gsmadmin@mycloud
GDSCTL> modify catalog -autovncr off
```

Specify the remote Scheduler agent registration password.

```
connect gsmadmin@mycloud
GDSCTL> modify catalog -agent_password mypass
```

Update catalog security scheme.

```
GDSCTL> modify catalog -autovncr OFF -oldpwd opwd -newpwd npwd -pwd pwd -
newkeys
```

modify cdb

Modify cdb attributes.

Syntax

```
modify cdb -cdb cdbname_list
          [-connect connect_identifier]
          [-pwd gsmrootuser_pwd]
          [-scan scan_address [-ons port]]
          [-savename]
```

Options

Table 2-55 GDSCTL modify cdb Options

| Option | Description |
|------------------------------------|--|
| -cdb <i>cdbname_list</i> | Specify a comma-delimited list of cdb names. |
| -connect <i>connect_identifier</i> | Specify an Oracle Net connect descriptor or a net service name that maps to a connect descriptor for the database that is being modified. |
| -ons <i>port</i> | Specify the ONS port. |
| -pwd <i>gsmrootuser_pwd</i> | Specify the password for GSMROOTUSER. |
| -savename | Specify this option to store a net service name specified with the <code>-connect</code> option in the Global Data Services catalog, rather than the connect descriptor mapped to that net service name. |
| -scan <i>scan_address</i> | Specify the SCAN address for a cluster. |

Usage Notes

Some parameters are not supported after the CDB contains shards or contains shards that have been deployed.

Examples

Change a password on a CDB.

```
GDSCTL> modify cdb -cdb1 cdb1 -pwd new_gsmrootuser_password
```

modify credential

Modifies an existing credential which will be used by the remote Scheduler agent to execute shard jobs.

Syntax

```
modify credential -credential credential_name
                  -osaccount account_name
                  -ospassword password
                  [-windows_domain domain_name]
```

Options

Table 2-56 GDSCTL modify credential Options

| Option | Description |
|------------------------------------|--|
| -credential <i>credential_name</i> | Specify the name of the credential to modify. |
| -osaccount <i>account_name</i> | Specify the operating system account which will be used for remote jobs. |
| -ospassword <i>password</i> | Specify the corresponding password for the account. |
| -windows_domain <i>domain_name</i> | If a Windows account has been specified, specify the corresponding domain name for that account. |

Usage Notes

This command modifies credentials which will be used to execute jobs on sharded hosts in response to administrative commands.

If the specified credential does not exist, the command returns an error.

Examples

Modify a credential named east_region_cred.

```
GDSCTL> modify credential -credential east_region_cred -osaccount agent_user
                  -ospassword newpass
```

modify database

Modifies the configuration parameters of the databases in a GDS pool, such as region, connect identifier, global service manager password, SCAN address, and ONS port.

Syntax

```
modify database -database db_name_list
                [-gdspool gdspool_name]
                [-shard shard_name]
                [-deploy_as PRIMARY|STANDBY]
                [-region region_name]
                [-pwd password]
                [-connect connect_identifier]
                [-scan scan_address]
                [-ons port]
                [-savename]
                [-cpu_threshold cpu]
                [-disk_threshold disk]
                [-rack rack_id]
                [-NETPARAM net_parameter_file | -NETPARAMFILE
net_parameter_file]
                [-DBPARAM db_parameter | -DBPARAMFILE db_parameter_file]
                [-DBTEMPLATE db_template | -DBTEMPLATEFILE db_template_file]
```

Options

Table 2-57 GDSCTL modify database Options

| Option | Description |
|------------------------------------|---|
| -connect <i>connect_identifier</i> | Specify an Oracle Net connect descriptor or a net service name that maps to a connect descriptor for the database that is being modified. |
| -cpu_threshold <i>cpu</i> | Specifies CPU Utilization percentage threshold. |
| -database <i>dbname_list</i> | Specify a comma-delimited list of database names. |
| -disk_threshold <i>disk</i> | Specifies the average latency in milliseconds of a synchronous single-block read. |
| -gdspool <i>gdspool_name</i> | Specify the database pool to which the databases belong. |
| -ons <i>port</i> | Specify the ONS port. |
| -pwd <i>password</i> | Specify the password for the GSMUSER. |
| -region <i>region_name</i> | Specify the region to which the databases belong. |
| -savename | Specify this option to store a net service name specified with the -connect option in the Global Data Services catalog, rather than the connect descriptor mapped to that net service name. |
| -scan <i>scan_address</i> | Specify the SCAN address for a cluster. |

Usage Notes

You can multiple databases if the `region` property is specified.

For all parameters except for the GDS region, first the appropriate changes must be done by the database administrator and then the `modify database` command must be run to update the modified parameters in the GDS catalog. Alternatively, you can use the [sync database \(synchronize database\)](#) command for this purpose.

You must connect to the catalog database as a user with the pool administrator privileges, using the [connect](#) command before running this command

Example

Change the region of databases DB1 and DB3 to EAST.

```
GDSCTL> modify database -database db1,db3 -region east
```

modify file

Updates the contents of a file in the catalog which can be used by subsequent GDSCTL commands.

Syntax

```
modify file -file file_name
           -source local_filename
```

Options

Table 2-58 GDSCTL modify file Options

| Option | Description |
|-------------------------------------|--|
| <code>-file file_name</code> | Specify the name of the file object to update. |
| <code>-source local_filename</code> | Specify an operating system file name specifying a file local to the machine running GDSCTL. |

Usage Notes

This command updates a named file object in the catalog by reloading the contents of an operating system file into the catalog.

Examples

Update a file named `east_region_db_params` with content from the local source file `/tmp/dbca_params.txt`

```
GDSCTL> modify file -file east_region_db_params -source /tmp/dbca_params.txt
```

modify gdspool

Modifies the configuration parameters of GDS pools.

Syntax

```
modify gdspool -gdspool gdspool_name_list
              [-removeuser user_name | -adduser user_name]
```

Options

Table 2-59 GDSCTL modify gdspool Options

| Option | Description |
|---|---|
| <code>-adduser <i>user_name</i></code> | Specify the user to add to the list of GDS pool administrators. This option grants the pool administrator role to the specified user. |
| <code>-gdspool <i>database_pool_list</i></code> | Specify a comma-delimited list of GDS pool names. |
| <code>-removeuser <i>user_name</i></code> | Specify the user to remove from the list of GDS pool administrators. This option revokes the pool administrator role from the specified user. |

Usage Notes

You must connect to the catalog database as a user with the pool administrator privileges, using the [connect](#) command before running this command

Example

Add PETER to the list of database pool administrators for the pool MYREADERFARM:

```
GDSCTL> modify gdspool -gdspool myreaderfarm -adduser peter
```

modify gsm

Modifies the configuration parameters of the global service manager. The changes take effect after the global service manager is restarted.

Syntax

```
modify gsm -gsm gsm_name
    [-catalog connect_id [-pwd password]]
    [-region region_name]
    [-localons ons_port]
    [-remoteons ons_port]
    [-endpoint gmsendpoint [-remote_endpoint remote_endpoint]]
    [-listener listener_port]
    [-wpwd wallet_password]
    [-encryption encryption]
    [-timeout timeout]
```

Options

Table 2-60 GDSCTL modify gsm Options

| Option | Description |
|---|--|
| <code>-catalog connect_id</code> | Specify the connect identifier for the Global Data Services catalog database. If a network service name is specified, it must be resolvable by the local naming method to a connect descriptor that allows the global service manager being modified to connect to the catalog database. |
| <code>-endpoint gsmendpoint</code> | Specify the protocol address that the global services manager listens on for client connection requests. If you use this option, the value you specify overrides the default endpoint. |
| <code>-gsm gsm_name</code> | Enter the name of the global service manager that you want to modify. If you do not specify a name, then the current global service manager name for the session (specified with the set gsm command) is used. |
| <code>-listener listener_port</code> | Specify the new listener port. |
| <code>-localons ons_port</code> | Specify the new local ONS port. |
| <code>-pwd password</code> | Specify the password for the GSMCATUSER account. If you do not specify a password, then you are prompted to enter one. |
| <code>-region region_name</code> | Specify the region to which the global service manager belongs. |
| <code>-remote_endpoint remote_endpoint</code> | Specify the protocol address that is used by the global service manager to receive database registration requests and communicate with other global service managers in the configuration. If you use this option, the value you specify overrides the default endpoint. |
| <code>-remoteons ons_port</code> | Specify the new remote ONS port. |
| <code>-wpwd</code> | Specify the password for the global service manager wallet. |
| <code>-encryption</code> | Encryption protocol for Advanced Network Option (ANO) used between GSM, GDSCTL and databases. OFF means that ANO is disabled. (AES256 AES192 OFF). |
| <code>-timeout</code> | Restart timeout |

Usage Notes

- You must run this command locally on the computer where you want to modify the global service manager.
- This command can be run only by the operating system user who started the global service manager.
- When you run this command, GDSCTL connects to the Global Data Services catalog as the GSMCATUSER user and prompts you for the GSMCATUSER password.

Example

Modify the global service manager named `gsm1` so that it is in the EAST region.

```
GDSCTL> modify gsm -gsm gsm1 -region east
```

modify region

Modifies the configuration parameters for a region.

Syntax

```
modify region -region region_name_list
              [-buddy region_name]
              [-weights weight]
```

Options**Table 2-61 GDSCTL modify region Options**

| Option | Description |
|---|---|
| <code>-buddy <i>region_name</i></code> | Specify the name of the buddy region |
| <code>-region <i>region_list</i></code> | Specify a comma-delimited list of region names |
| <code>-weights <i>weight</i></code> | Used for static RLB distribution. format: name = value,...,name = value |

Usage Notes

You must connect to the catalog database as a user with the Global Data Services administrator privileges, using the [connect](#) command before running this command.

To clear buddy region or weight, call `MODIFY REGION` and specify empty quotes as the value. If `WEIGHTS` is specified, dynamic load balancing is replaced by static (not recommended).

Example

Modify two regions, EAST and WEST, as follows:

```
GDSCTL> modify region -region west -buddy east
```

modify service

Modifies the service attributes.

Syntax

To add more preferred or available databases to a global service:

```
modify service [-gdspool gdspool_name]
              -service service_name
              {-preferred db_name_list | -available db_name_list}
```

To modify the attributes of a global service:

```
modify service [-gdspool gdspool_name]
               -service service_name
               [-locality {ANYWHERE | LOCAL_ONLY}]
               [-region_failover]
               [-role {PRIMARY | PHYSICAL_STANDBY [-failover_primary] |
                     LOGICAL_STANDBY | SNAPSHOT_STANDBY}]
               [-lag {lag_value | ANY}]
               [-notification {TRUE | FALSE}]
               [-rlbgoal {SERVICE_TIME | THROUGHPUT}]
               [-clbgoal {SHORT | LONG}]
               [-tafpolicy {BASIC | NONE | PRECONNECT}]
               [-policy policy]
               [-failovertype {NONE | SESSION | SELECT | TRANSACTION}]
               [-failovermethod {NONE | BASIC}]
               [-dtp {TRUE | FALSE}]
               [-sql_translation_profile stp_name]
               [-failoverretry failover_retries]
               [-failoverdelay failover_delay]
               [-edition edition_name]
               [-commit_outcome {TRUE | FALSE}]
               [-retention retention_seconds]
               [-session_state {DYNAMIC | STATIC}]
               [-replay_init_time replay_init_time]
               [-table_family family]
```

To move a global service from one database to another database:

```
modify service [-gdspool gdspool_name]
               -service service_name
               -old_db db_name
               -new_db db_name
               [-force]
```

To change an available database to a preferred database for a service:

```
MODIFY SERVICE [-gdspool gdspool_name]
               -service service_name
               -available db_name_list
               -preferred
```

To change databases between preferred and available status:

```
modify service [-gdspool gdspool_name]
               -service service_name
               {-preferred_all |
               {-modifyconfig -preferred db_name_list [-available
               db_name_list]}}
```

To modify properties for a global service that are specific to an Oracle RAC database:

```
modify service [-gdspool gdspool_name]
               -service service_name
```

```

-database db_name
[-server_pool server_pool_name |
{-add_instances|-modify_instances} -preferred inst_list
 -available inst_list |
 -drop_instances inst_list
-cardinality {UNIFORM | SINGLETON}

```

Options

Table 2-62 GDSCTL modify service Options

| Option | Description |
|---|---|
| <code>-add_instances [-preferred <i>comma-delimited-list</i>] [-available <i>comma-delimited-list</i>]</code> | Provides a list of preferred and available instances for the given service on the given database. The provided list over-rides existing assigned instances, if any. Using the <code>-preferred</code> and <code>-available</code> options is optional, but at least one of these must be provided. |
| <code>-available <i>db_name_list</i></code> | Specify a comma-delimited list of available databases on which the service runs, if the preferred databases are not available. The list of available instances must be mutually exclusive with the list of preferred instances. If you attempt to add a preferred or available database to a service that was configured with <code>-preferred_all</code> , then GDSCTL returns an error. |
| <code>-cardinality {UNIFORM SINGLETON}</code> | Specify the cardinality option for a service running on a policy-managed Oracle RAC database. Services with cardinality set to <code>UNIFORM</code> are offered on all database instances. Services with cardinality set to <code>SINGLETON</code> are offered on only one database instance. |
| <code>-clbgoal {SHORT LONG}</code> | For connection load balancing goal: set to <code>SHORT</code> if using run-time load balancing, set to <code>LONG</code> for long running connections such as batch jobs or older SQL*Forms style. The default value for this option is <code>SHORT</code> . |
| <code>-commit_outcome {TRUE FALSE}</code> | Enable Transaction Guard; when set to <code>TRUE</code> , the commit outcome for a transaction is accessible after the transaction's session fails due to a recoverable outage. |
| <code>-database <i>db_name</i></code> | Specify the name of the database on which you want to modify the service. When <code>-database</code> is specified, you must specify exactly one of: <ul style="list-style-type: none"> <code>-server_pool</code> and/or <code>-cardinality</code>. Either is optional, but at least one must appear, both can be used at once. <code>-add_instances</code>, <code>-modify_instances</code>, or <code>-drop_instances</code>. Exactly one of these three options must be used. |
| <code>-dtp {TRUE FALSE}</code> | Indicates whether Distributed Transaction Processing should be enabled for this service. This ensures that the service is offered at exactly one instance at a time for XA affinity. |

Table 2-62 (Cont.) GDSCTL modify service Options

| Option | Description |
|--|---|
| <code>-drop_instances inst_list</code> | Provide a list of instances to be removed from the existing assigned instances for a given service on a given database. The provided list of instances will be removed from the existing assigned list. |
| <code>-edition edition_name</code> | Specify the initial session edition of the service. When an edition is specified for a service, all subsequent connections that specify the service use this edition as the initial session edition. However, if a session connection specifies a different edition, then the edition specified in the session connection is used for the initial session edition. GDSCTL does not validate the specified edition name. During connection, the connect user must have <code>USE</code> privilege on the specified edition. If the edition does not exist or if the connect user does not have <code>USE</code> privilege on the specified edition, then an error is raised. |
| <code>-failover_primary</code> | If you set the <code>-role</code> option to <code>PHYSICAL_STANDBY</code> , then you can use this option to enable the service for failover to the primary database. |
| <code>-failoverdelay failover_delay</code> | For Application Continuity and TAF, the time delay (in seconds) between reconnect attempts for each incident at failover. |
| <code>-failovermethod {NONE BASIC}</code> | TAF failover method (for backward compatibility only). If <code>failovertime</code> is set to either <code>SESSION</code> or <code>SELECT</code> , then choose <code>BASIC</code> for this option. |
| <code>-failoverretry failover_retries</code> | For Application Continuity and TAF, the number of attempts to connect after an incident. |
| <code>-failovertime {NONE SESSION SELECT TRANSACTION}</code> | Specify the failover type. To enable Application Continuity for Java, set this parameter to <code>TRANSACTION</code> . To enable Transparent Application Failover (TAF) for OCI, set this parameter to <code>SELECT</code> or <code>SESSION</code> . |
| <code>-force</code> | If you use this option, then all sessions are disconnected when the service is moved, requiring the sessions using the service to reconnect (potentially to a different instance). If you do not use this option, then the sessions that are connected to a database using this service stay connected, but all new sessions cannot be established to the service. |
| <code>-gdspool gdspool_name</code> | Specify the name of the database pool to which the service belongs. If not specified and there is only one <code>gdspool</code> with access granted to user, it is used as the default <code>gdspool</code> . |
| <code>-lag {lag_value ANY}</code> | Specify the lag for the service in seconds. You can use the keyword <code>ANY</code> to indicate that there is no upper threshold on the lag time. The default value for the <code>-lag</code> option is <code>ANY</code> . |

Table 2-62 (Cont.) GDSCTL modify service Options

| Option | Description |
|--|--|
| <code>-locality {ANYWHERE LOCAL_ONLY}</code> | The service region locality. If you do not use this option, then the default value of ANYWHERE is used for the service. |
| <code>-modifyconfig</code> | Use this option to indicate that you are changing the current list of preferred and available databases for the service. If you use this option, then any databases that are not specified in either the preferred or available list, but were previously assigned, are removed from the list of databases on which the service can run. |
| <code>-modify_instances [-preferred comma-delimited-list] [-available comma-delimited-list]</code> | <p>The provided <i>comma-delimited-list</i> of preferred and available instances is merged with the existing list currently stored in the catalog.</p> <p>If you specify an instance in the <i>comma-delimited-list</i> that is not already in the stored list, it is added to the stored list in its correct mode (preferred or available.)</p> <p>If you specify in <i>comma-delimited-list</i> an instance that is already in the stored list, then the mode of the instance in the stored list is modified to the provided mode (preferred or available). If the user provided mode is the same as the stored mode, then the mode of the instance will not be changed.</p> <p>Any instances already in the stored list that are not in the provided list remain unchanged in the stored list.</p> <p>Note that an instance cannot be both preferred and available, it can be in one mode only.</p> <p><code>-preferred</code> and <code>-available</code> are optional but at least one list must be provided.</p> |
| <code>-new_db database_name</code> | <p>Specify the name of the new database on which the service runs.</p> <p>If you attempt to move a service that was configured with <code>-preferred_all</code>, then GDSCTL returns an error.</p> |
| <code>-notification {TRUE FALSE}</code> | Enable Fast Application Notification (FAN) for OCI connections. |
| <code>-old_db database_name</code> | <p>Specify the name of the old database on which the service runs.</p> <p>If you attempt to move a service that was configured with <code>-preferred_all</code>, then GDSCTL returns an error.</p> |
| <code>-policy {AUTOMATIC MANUAL}</code> | <p>Specify the management policy for the service.</p> <p>If you specify <code>AUTOMATIC</code> (the default), then the service automatically starts when the database restarts, either by a planned restart or after a failure. Automatic restart is also subject to the service role.</p> <p>If you specify <code>MANUAL</code>, then the service is never automatically restarted upon planned restart of the database. A <code>MANUAL</code> setting does not prevent the global service manager from monitoring the service when it is running and restarting it if a failure occurs.</p> |

Table 2-62 (Cont.) GDSCTL modify service Options

| Option | Description |
|--|--|
| <code>-pdbname <i>pdb_name</i></code> | Specify the pluggable database name. |
| <code>-preferred <i>db_name_list</i></code> | Specify a comma-delimited list of preferred databases on which the service runs. When changing a database from available to preferred, you do not specify a value for the <code>-preferred</code> option. The list of preferred instances must be mutually exclusive with the list of available instances. If you attempt to add a preferred or available database to a service that was configured with <code>-preferred_all</code> , then GDSCTL returns an error. |
| <code>-preferred_all</code> | Specifies that all the databases in the database pool are preferred databases. Any new databases added to the pool are configured as preferred databases for this service. This option cannot be used with the <code>-preferred</code> and <code>-available</code> options. |
| <code>-region_failover</code> | Indicates that the service is enabled for region failover. You can only use this option when you specify <code>LOCAL_ONLY</code> for the <code>-locality</code> option. |
| <code>-replay_init_time <i>replay_init_time</i></code> | For Application Continuity, this parameter specifies the time (in seconds) after which replay is not initiated. Default value is 300 seconds. |
| <code>-retention <i>retention_seconds</i></code> | For Transaction Guard (<code>commit_outcome</code> set to <code>TRUE</code>), this parameter determines the amount of time (in seconds) that the commit outcome is retained in the database. |
| <code>-rlbgoal {SERVICE_TIME THROUGHPUT}</code> | Run-time Load Balancing Goal. Set this parameter to <code>SERVICE_TIME</code> to balance connections by response time. Set this parameter to <code>THROUGHPUT</code> to balance connections by throughput. If you do not use this option, then the value defaults to <code>SERVICE_TIME</code> for the run-time load balancing goal. |
| <code>-role {[PRIMARY] [PHYSICAL_STANDBY] [-failover_primary] [LOGICAL_STANDBY] [SNAPSHOT_STANDBY]}</code> | Specify the database role that the database must be for this service to start on that database. This applies only to database pools that contain an Oracle Data Guard broker configuration. See Also: <i>Oracle Data Guard Concepts and Administration</i> for more information about database roles |
| <code>-server_pool <i>server_pool_name</i></code> | Specify the name of the Oracle RAC server pool in the GDS pool database to which the service belongs (for a policy-managed Oracle RAC database). |
| <code>-service <i>service_name</i></code> | Specify the name of the global service. |
| <code>-session_state {DYNAMIC STATIC}</code> | For Application Continuity, this parameter specifies whether the session state that is not transactional is changed by the application. A value of <code>DYNAMIC</code> is recommended for most applications. |

Table 2-62 (Cont.) GDSCTL modify service Options

| Option | Description |
|--|--|
| <code>-sql_translation_profile stp_name</code> | <p>Use this option to specify a SQL translation profile for a service that you are adding after you have migrated applications from a non-Oracle database to an Oracle database.</p> <p>This option corresponds to the SQL translation profile parameter in the <code>DBMS_SERVICE</code> service attribute.</p> <p>Notes:</p> <ul style="list-style-type: none"> • Before using the SQL translation feature, you must migrate all server-side application objects and data to the Oracle database. • Use the config service command to display the SQL translation profile. <p>See Also: <i>Oracle AI Database SQL Translation and Migration Guide</i> for more information about SQL translation</p> |
| <code>-table_familyfamily</code> | <p>Specifies the table family name as a property of the service. This parameter takes one of the table family values (root table schema.name) as shown in the <code>CONFIG TABLE FAMILY</code> output.</p> <p>If the schema name or the table name is case-sensitive, use two-level quotes (single quotes outside, double quotes inside) around the whole string, for example, <code>' "TESTUSER1.Customers6" '</code>. No quotes are needed if neither name is case sensitive.</p> |
| <code>-tafpolicy {BASIC NONE }</code> | <p>TAF policy specification (for administrator-managed databases only).</p> |

Usage Notes

You must connect to the catalog database as a user with the pool administrator privileges, using the [connect](#) command before running this command.

Use this command to:

- Add databases to the preferred or available lists for the service
- Move a service from one database to another database
- Change an available database to a preferred database or a preferred database to an available database
- Modify the high availability attributes of the service

If you want to temporarily move a service from one database to a different database, then use the [relocate service](#) command.

Examples

Add the database DB3 as a preferred database for the service `G_SALES_REPORT` in the database pool `MYREADERFARM`.

```
GDSCTL> modify service -gdspool myreaderfarm -service g_sales_report -
preferred db3
```

Modify the service G_DAILY_SALES_REPT in the database pool MYREADERFARM to change the run-time load balancing goal to THROUGHPUT.

```
GDSCTL> modify service -gdspool myreaderfarm -service g_daily_sales_rept
        -rlbgoal THROUGHPUT
```

Move the service G_SALES_REPORT in the database pool MYREADERFARM from the database DB1 to DB4.

```
GDSCTL> modify service -gdspool myreaderfarm -service g_sales_report
        -old_db db1 -new_db db4
```

Upgrade the DB3 database from an available database to a preferred database for the service G_SALES_REPORT in the database pool READFARM.

```
GDSCTL> modify service -gdspool readfarm -service g_sales_report
        -available db3 -preferred
```

Assume the service G_SALES_REPORT currently has the databases DB1 and DB2 assigned as preferred databases, and the database DB3 assigned as an available database. Exchange the preferred and available databases DB1 and DB3, and remove the DB2 database for the service SALES_REPORT in the database pool READFARM.

```
GDSCTL> modify service -gdspool readfarm -service g_sales_report -modifyconfig
        -available db3 -preferred db1
```

Modify the properties of the service G_SALES_REPORT in the database pool READFARM to specify that it should run only in the server pool named SALESPOOL for the policy-managed Oracle RAC database DB1.

```
GDSCTL> modify service -gdspool readfarm -service g_sales_report -database db1
        -server_pool salespool
```

Supply the preferred and available instances for the given service on the given database.

```
GDSCTL> modify service -gdspool mypool -service mysvc -database mydb -
add_instances
        -preferred inst1,inst2 -available inst3,inst4
```

In a system-managed sharded database, the table family ID parameter is specified as a property of the service.

```
GDSCTL> modify service -GDSPPOOL shdpool -table_family sales.customer -service
sales
```

See Also

[#unique_140](#)

modify shard

Modify shard attributes.

Syntax

```
modify shard -shard shname_list
            [-region region_name]
            [-connect connect_identifier]
            [-pwd password]
            [-scan scan_address [-ons port]]
            [-savename]
            [-cpu_threshold cpu]
            [-disk_threshold disk]
            [-destination destination_name]
            [-credential credential_name |
            [[-osaccount account_name |
            [-ospassword password]
            [-windows_domain domain_name]]]]
```

Options

Table 2-63 GDSCTL modify shard Options

| Option | Description |
|---|--|
| <code>-connect <i>connect_identifier</i></code> | Specify an Oracle Net connect descriptor or a net service name that maps to a connect descriptor for the database that is being modified. |
| <code>-cpu_threshold <i>cpu</i></code> | Specifies CPU Utilization percentage threshold. |
| <code>-credential <i>credential_name</i></code> | Specify the credential to use on the remote machine which specifies the user name and password under which database creation will occur. |
| <code>-destination <i>destination_name</i></code> | Specify the name of the remote executable agent through which the database will be created. |
| <code>-disk_threshold <i>disk</i></code> | Specifies the average latency in milliseconds of a synchronous single-block read. |
| <code>-ons <i>port</i></code> | Specify the ONS port. |
| <code>-osaccount <i>account_name</i></code> | Specify the operating system account which will be used for remote jobs. |
| <code>-ospassword <i>password</i></code> | Specify the corresponding password for the account specified in <code>-osaccount</code> . |
| <code>-pwd <i>password</i></code> | Specify the password for the GSMUSER. |
| <code>-region <i>region_name</i></code> | Specify the region to which the databases belong. |
| <code>-savename</code> | Specify this option to store a net service name specified with the <code>-connect</code> option in the Global Data Services catalog, rather than the connect descriptor mapped to that net service name. |
| <code>-scan <i>scan_address</i></code> | Specify the SCAN address for a cluster. |
| <code>-shard <i>shname_list</i></code> | Specify a comma-delimited list of shard names. |

Table 2-63 (Cont.) GDSCTL modify shard Options

| Option | Description |
|--|--|
| <code>-windows_domain domain_name</code> | Specify the corresponding domain name if a Windows account has been specified in <code>-osaccount</code> . |

Usage Notes

The `REGION` parameter cannot be modified for a shard that belongs to a shardgroup. The modification has to be done at the shardgroup level.

The `DESTINATION` and `CREDENTIAL` parameters are only modifiable when the shard has not yet been deployed, since these parameters only have meaning for the deployment process and are no longer referenced once deployment has completed successfully.

Examples

```
GDSCTL> modify shard -shard shard1 -ons 23222
```

modify shardgroup

Modify shardgroup attributes.

Syntax

```
modify shardgroup -shardgroup shardgroup_name
                  [-region region_name]
                  [-shardspace shardspace_name]
                  [-repfactor number]
                  [-deploy_as {PRIMARY | STANDBY | ACTIVE_STANDBY}]
```

Options**Table 2-64 GDSCTL modify shardgroup Options**

| Option | Description |
|--|--|
| <code>-shardgroup shardgroup_name</code> | Specify the name of the shardgroup to be modified. |
| <code>-region region_name</code> | Specify the region the shardgroup resides in. |
| <code>-shardspace shardspace_name</code> | Specify the shardspace this shardgroup belongs to. |
| <code>-repfactor number</code> | Specify the number of replicas for each piece of data stored in this shardgroup. |
| <code>-deploy_as {PRIMARY STANDBY ACTIVE_STANDBY}</code> | Specify the initial role for a newly deployed database: PRIMARY, STANDBY, or ACTIVE_STANDBY. |

Usage Notes

All shardgroup attributes, except for `DEPLOY_AS`, can only be modified when the shardgroup does not contain any deployed shards. `DEPLOY_AS` can be modified at any time because it does not affect shards that were already added to the shardgroup.

Examples

Modify the GROUP1 shardgroup to have a replication factor of 3.

```
GDSCTL> modify SHARDGROUP -SHARDGROUP group1 -REPFACOR 3
```

modify shardspace

Modify shardspace parameters.

Syntax

```
modify shardspace -shardspace shardspace_name
                  [-chunks number]
                  [-protectmode dg_protection_mode]
                  [-repunits repunits]
                  [-refactor refactor]
```

Options

Table 2-65 GDSCTL modify shardspace Options

| Option | Description |
|--|--|
| -shardspace <i>shardspace_name</i> | Specify the name of the shardspace to be modified. |
| -chunks <i>number</i> | Specify the number of chunks in the shardspace. |
| -protectmode <i>dg_protection_mode</i> | Specify the Data Guard Protection mode: MAXPROTECTION, MAXAVAILABILITY, or MAXPERFORMANCE. This option can only be executed where Data Guard replication technology is used. |
| -refactor | Replication factor (the number of replicas for each piece of data stored in a shardgroup). This parameter can only be used with NATIVE replication and system-managed or composite sharding, and is mandatory in these cases. It doesn't apply to user-defined sharding or a federated database since there are no shardgroups in this case. |
| -repunits | Total number of replication units (SNR only). |

Usage Notes

The number of chunks can only be modified if a shardspace does not contain deployed shards. This command is not applicable to a federated database.

Examples

Modify the GOLD shardspace to have 6000 chunks.

```
GDSCTL> modify shardspace -shardspace gold -chunks 6000
```

move chunk

Moves a listed set of chunks from one shard to another shard or multiple shards.

Syntax

```
move chunk -chunk {chunk_id_list | ALL}
           -source shard_name
           [-target shard_name]
           [-timeout]
           [-verbose]
           [-copy]
```

Options

Table 2-66 GDSCTL move chunk Options

| Option | Description |
|--------------------------------------|--|
| -chunk { <i>chunk_id_list</i> ALL} | Specify a comma-separated list of chunk IDs. If -chunk ALL is specified without the -target option, all of the chunks are removed from the source shard and distributed to all of the remaining shards in a round-robin manner. |
| -copy | Copy the chunk instead of moving (OGG only). |
| -source <i>shard_name</i> | Specify the name of the source shard. |
| -target <i>shard_name</i> | Specify the name of the target shard. |
| -timeout | Specify a connection retention time-out for the interval between when FAN is sent to the clients and a chunk going into read-only mode or down. |
| -verbose | Enable verbose output mode. |

Usage Notes

Chunks cannot be moved between shards that belong to different shardgroups.

If -chunk ALL is specified without the -target option, all of the chunks are removed from the source shard and distributed to all of the remaining shards in a round-robin manner.

Examples

Move chunks 3 and 4 from SALE1 to SALE3.

```
GDSCTL> move chunk -chunk 3,4 -source sale1 -target sale3
```

Move all chunks from sale1 and distribute evenly among the remaining shards.

```
GDSCTL> move chunk -chunk ALL -source sale1
```

move ru

Move a member replica of a replication unit from one shard to another.

Syntax

```
[ru|replication_unit] -ru ru_id -source source_db
                        -target target_db
                        [-force]
```

Options

Table 2-67 GDSCTL move ru Options

| Option | Description |
|----------|--|
| -force | Allow operation to bypass RAFT replication role checks. |
| -ru | Replication unit ID |
| -source | Name of the source shard. |
| -target | Name of the target shard. |
| -timeout | Timeout of connection retention between FAN is sent to clients and chunk going read-only/down (seconds). |

Examples

```
MOVE RU -RU 1 -SOURCE sh1 -TARGET sh2
```

quit

Quit GDSCTL utility.

Syntax

```
quit | exit
```

recover shard

Executes all DDL statements on the specified shard (database), starting from the one that was previously executed with errors. The command is intended to perform all skipped DDL changes after database administrator fixes shard issues.

Syntax

```
recover shard -gdspool pool
              -shard shard_name
              [-skip_first|-ignore_first]
              [-full]
```

Options

Table 2-68 GDSCTL recover shard Options

| Option | Description |
|--------------------------|--|
| -full | Full recovery mode. |
| -gdspool <i>pool</i> | Specify the GDS pool. If not specified and there is only one GDS pool with access granted to user, it will be used by default. |
| -ignore_first | Make first failed DDL statement obsolete. |
| -shard <i>shard_name</i> | The name of the shard. |
| -skip_first | Skip the first failed DDL statement. |

Usage Notes

Use `SKIP_FIRST` to skip first DDL. This is typically required after manual fix done by database administrator. For example, if `CREATE TABLE` statement fails because of a lack of space, the database administrator fixes the issue and re-executes `CREATE TABLE`. To avoid `ORA-39151 (table exists)` in `RECOVER SHARD` the database administrator must specify `-SKIP_FIRST`.

Use `IGNORE_FIRST` to mark the first DDL as obsolete. This is required when the wrong DDL statement was specified and failed on all shards. In this case, you need to mark it down as obsolete. `FULL` mode performs a complete recovery, including DDL operations, failed chunk migration, tablespace sets reconstruction, and database parameters.

Examples

Recover shard `shd1`.

```
GDSCTL> recover shard -shard shd1
```

relocate chunk

This command moves a list of chunks in all the replicas of the specified source RU to all the replicas in the target ru.

Syntax

```
relocate chunk -chunk {chunk_id_list | all} -sourceru ru_id
                [-targetru ru_id]
                [-timeout timeout]
```

Options

Table 2-69 GDSCTL relocate chunk Options

| Option | Description |
|-----------|---|
| -chunk | List of numeric chunk identifiers or <code>ALL</code> for all chunks. |
| -sourceru | Source replication unit ID |
| -targetru | Target replication unit ID |

Table 2-69 (Cont.) GDSCTL relocate chunk Options

| Option | Description |
|----------|--|
| -timeout | Timeout of connection retention between FAN is sent to clients and chunk going read-only/down (seconds). |
| -verbose | Enable verbose mode |

Usage Notes

The source and target replication unit must be colocated in the same shard. leaders on the same shard and followers on the same shards. if not use SWITCHOVER to move the leader and MOVE RU to move the followers to colocated shards.

The specified chunks must be in the same source replication unit. If target ru is not specified, an empty target replication unit will be created.

Examples

```
GDSCTL> relocate chunk -chunk 3,4 -sourceru 1 -targetru 2
```

relocate service

Stops a service on one database and starts the service on a different database.

Syntax

```
relocate service [-gdspool gdspool_name]
                 -service service_name
                 -old_db db_name
                 -new_db db_name
                 [-force]
                 [-override [-oldpwd oldpassword] [-newpwd newpassword]]
```

Options**Table 2-70 GDSCTL relocate service Options**

| Option | Description |
|------------------------------|---|
| -force | If you use this option, then all sessions are disconnected when the service is moved, requiring the sessions using the service to reconnect (potentially to a different instance). If you do not use this option, then the sessions that are connected to a database using this service stay connected, but new sessions cannot be established to the service. |
| -gdspool <i>gdspool_name</i> | Specify the name of the database pool where the service is located. If not specified and there is only one <i>gdspool</i> with access granted to user, it is used as the default <i>gdspool</i> . |
| -new_db <i>db_name</i> | Specify the name of the database to which you want to move the service. |
| -newpwd <i>newpassword</i> | Specify the password for the GSMUSER in the database to which the service is being relocated (the target database). |

Table 2-70 (Cont.) GDSCTL relocate service Options

| Option | Description |
|------------------------------------|---|
| <code>-old_db db_name</code> | Specify the name of the database where the service is currently located. |
| <code>-oldpwd oldpassword</code> | Specify the password for the GSMUSER in the source database, or the database where the service is currently located. |
| <code>-override</code> | This option causes the command to execute without updating the global service manager catalog. You can use this option when the catalog database is unavailable. During normal operation, you should not use this option. |
| <code>-service service_name</code> | Specify the name of the global service you are relocating. |

Usage Notes

Unlike using the [modify service](#) command to change the location of a service, this command does not change the underlying configuration. This command temporarily relocates a service to run on another database.

If `-force` is not specified, then the global service must have been started on the old database and not running on the new database prior to command execution. If `-force` is not specified, then sessions already connected to this global service stay connected, but new sessions cannot be established.

If `-override` is specified the command will be executed without going to the GDS catalog. Use this option when the GDS catalog is unavailable. It is not recommended for use under normal operation.

If you attempt to use this command on a service that was previously configured with the `-preferred_all` option, then GDSCTL returns an error.

You must connect to the catalog database as a user with the pool administrator privileges, using the command [connect](#) before running this command

Example

Relocate the service SALES_REPORT in the READFARM database pool from the DB2 database to the DB3 database.

```
GDSCTL> relocate service -gdspool readfarm -service sales_report -old_db db1
-new_db db3
```

remove brokerconfig

Removes an Oracle Data Guard broker configuration from a GDS pool.

Syntax

```
remove brokerconfig [-gdspool gdspool_name]
```

Options

Table 2-71 GDSCTL remove brokerconfig Options

| Syntax | Description |
|---|--|
| <code>-gdspool <i>gdspool_name</i></code> | Specify the GDS pool from which you want to remove the Oracle Data Guard broker configuration (not required, however, if not specified and there is only one GDS pool with access granted to the user, and it is used by default). |

Usage Notes

You must connect to the catalog database as a user with the pool administrator privileges, using the command [connect](#) before running this command.

If a GDS pool does not contain a Data Guard Broker configuration, an error is returned.

Example

Remove the Oracle Data Guard broker configuration from the database pool MYDGPOOL.

```
GDSCTL> remove brokerconfig -gdspool myreaderfarm
```

remove cdb

Removes one or more CDBs from the shard catalog, but does not destroy it.

Syntax

```
remove cdb -cdb {cdb_name_list | ALL}
           [-force]
```


Options

Table 2-72 GDSCTL remove cdb Options


| Option | Description |
|--|--|
| <code>-cdb {<i>cdb_name_list</i> ALL}</code> | Specify a comma-delimited list of CDB names to remove, or specify ALL to remove all CDBs from the catalog. |

Table 2-72 (Cont.) GDSCTL remove cdb Options

| Option | Description |
|--------|---|
| -force | Remove one or more specified CDBs, even if they are inaccessible and/or contain PDB shards which may contain chunks. It might result in a lower number of replicas or total unavailability for a certain range of data. |

 **Warning**

No chunks are moved before removing the CDB which may result in data loss.

 **Warning**

Forced removal of a CDB will also cause the removal of all CDBs that are replicas of the CDB being forcibly removed.

Examples

Remove the cdb named cdb1.

```
GDSCTL> remove cdb -cdb cdb1
```

remove credential

Removes an existing credential.

Syntax

```
remove credential -credential credential_name
```

Options

Table 2-73 GDSCTL remove credential Options

| Option | Description |
|------------------------------------|---|
| -credential <i>credential_name</i> | Specify the name of the credential to remove. |

Usage Notes

This command removes an existing credential. When the credential is removed, the catalog may no longer be able to execute jobs on sharded hosts in response to administrative commands.

If the specified credential does not exist, the command returns an error.

Examples

Remove a credential named east_region_cred.

```
GDSCTL> remove credential -credential east_region_cred
```

remove database

Removes databases from a GDS pool.

Syntax

```
remove database [-gdspool gdspool_name]  
                {-all | -database db_name_list}  
                [-force]
```

Options

Table 2-74 GDSCTL remove database Options

| Option | Description |
|-------------------------------|---|
| -all | Removes all databases in the database pool. |
| -database <i>db_name_list</i> | Specify a comma-delimited list of database names that you want to remove from the database pool. You cannot specify a database that was added through an Oracle Data Guard broker configuration; you must use Oracle Data Guard to remove these databases. |
| -gdspool <i>gdspool_name</i> | Specify the GDS pool name. If not specified, and there is only one GDS pool with access granted to the user, it will be used by default. |
| -force | Removes the database from the catalog even if the database is not available. Using this option can result in global services not being removed from the database. |

Usage Notes

Note

Only add a database to a GDS pool if its Oracle release, release update (RU), or recommended release update (RUR) and and datapatch level exactly match all existing database members. Never mix newer or older database binaries. If this is done, then it can cause errors and unstable services. Also, do not add or remove databases from the pool during rolling patching or upgrades. Onboard at the matched version first, verify services, and then perform rolling patching.

You must connect to the catalog database as a user with the pool administrator privileges, using the command [connect](#) before running this command.

If a pool already contains a Data Guard Broker configuration, an error is returned because a database must be removed through DGMGRL in this case.

With Oracle Globally Distributed AI Database, only an undeployed database can be removed. If a database is offline or inaccessible, it has to be first undeployed with the `-force` option and then removed with the `-force` option.

Example

Remove the database DB1 from the global service management configuration.

```
GDSCTL> remove database -database db1 -gdspool pool1
```

remove file

Removes an existing file object from the catalog.

Syntax

```
remove file -file file_name
```

Options

Table 2-75 GDSCTL remove file Options

| Option | Description |
|-------------------------------------|---|
| <code>-file <i>file_name</i></code> | Specify the name of the file object to remove from the catalog. |

Usage Notes

If the specified file object does not exist, the command returns an error.

Examples

Remove a file named `east_region_db_params`.

```
GDSCTL> remove file -file east_region_db_params
```

remove gdspool

Removes a GDS pool from the current configuration.

Syntax

```
remove gdspool -gdspool gdspool_name_list
```

Options

Table 2-76 GDSCTL remove gdspool Options

| Option | Description |
|--|---|
| <code>-gdspool <i>gdspool_name_list</i></code> | Specify a comma-delimited list of GDS pool names. |

Usage Notes

You must connect to the catalog database as a user with the Global Data Services administrator privileges, using the command [connect](#) before running this command.

Example

Remove the GDS pools tempreaders and myfarm from the Global Data Services framework.

```
GDSCTL> remove gdspool -gdspool tempreaders,myfarm
```

remove gsm

Removes a global service manager from the configuration.

Syntax

```
remove gsm [-gsm gsm_name]
```

Options

Table 2-77 GDSCTL remove gsm Options

| Syntax | Description |
|-----------------------------------|---|
| <code>-gsm <i>gsm_name</i></code> | Specify the name of the global service manager that you want to remove. If the name is not specified, then the current global service manager is removed. |

Usage Notes

The removal of a global service manager requires at least one global service manager to be running to perform cleanup of Global Data Services databases. If there is only one global service manager in the Global Data Services configuration, then it has to be running to be removed.

You must connect to the catalog database as a user with the Global Data Services administrator privileges, using the command [connect](#) before running this command.

Example

Remove the global service manager named gsm5 from the configuration.

```
GDSCTL> remove gsm -gsm gsm5
```

remove invitednode (remove invitedsubnet)

Remove host address information from the valid node checking for registration (VNCR) list in the Global Data Services catalog. This command removes either the specified invitednode or all invitednodes that correspond to an alias.

Syntax

```
remove invitednode {[-group group_name]|vncr_id}
```

Options

Table 2-78 GDSCTL remove invitednode (remove invitedsubnet) Options

| Option | Description |
|--------------------------------|---|
| <code>-group group_name</code> | Specify an alias which defines a group of VNCRs. This alias can be referenced in other commands related to invited nodes. |
| <code>vncr_id</code> | The host address information, which can be an IPv4 or IPv6 address, a host name, a netmask, or other identifier for a server. The host address information cannot contain any spaces. |

Usage Notes

You must connect to the catalog database as a user with the pool administrator privileges, using the command [connect](#) before running this command

Examples

Remove the invitednode 198.51.100.22 from the catalog.

```
GDSCTL> remove invitednode 198.51.100.22
```

Remove the VNCR alias group EAST_SRV from the catalog.

```
GDSCTL> remove invitednode -group east_srv
```

remove region

Removes the specified regions from the global service management framework.

Syntax

```
remove region -region region_list
```

Options

Table 2-79 GDSCTL remove region Options

| Option | Description |
|---|--|
| <code>-region <i>region_list</i></code> | Specify a comma-delimited list of region names |

Usage Notes

You must connect to the catalog database as a user with the Global Data Services administrator privileges, using the [connect](#) command before running this command.

Example

Remove the region named SOUTH from the configuration.

```
GDSCTL> remove region -region south
```

remove ru

Remove empty replication unit from a sharded database configuration.

Syntax

```
remove [ru|replication_unit] -ru ru_id
        [-timeout timeout]
```

Options

Table 2-80 GDSCTL remove ru Options

| Option | Description |
|-----------------------|---|
| <code>-ru</code> | Replication unit ID |
| <code>-timeout</code> | GSM (shard director) requests timeout (in seconds). |

Usage Notes

Replication unit must be empty prior to its remove. Use `RELOCATE CHUNK` command to move chunks between replication units.

Examples

```
GDSCTL> remove ru -ru 1
```

remove service

Removes a service from a database pool.

Syntax

```
remove service [-gdspool gdspool_name]
               -service service_name
```

Options

Table 2-81 GDSCTL remove service Options

| Option | Description |
|------------------------------|--|
| -gdspool <i>gdspool_name</i> | Specify the name of the GDS pool from which you want to remove the service. If not specified and there is only one <i>gdspool</i> with access granted to user, then it is used as the default <i>gdspool</i> . |
| -service <i>service_name</i> | Specify the name of the service that you want to remove. |

Usage Notes

You must connect to the catalog database as a user with the pool administrator privileges, using the command [connect](#) before running this command

Example

Remove the service `sales_report` from the database pool MYREADERFARM.

```
GDSCTL> remove service -gdspool myreaderfarm -service sales_report
```

See Also

[#unique_141](#)

remove shard

Removes one or more shards from the sharded database.

Syntax

```
remove shard {-shard {shard_name_list | ALL} |
              -shardspace shardspace_list |
              -shardgroup shardgroup_list}
              [-force]
```

Options

Table 2-82 GDSCTL remove shard Options

| Option | Description |
|---|--|
| <code>-shard {shard_name_list ALL}</code> | Specify a comma-delimited list of shard names to remove, or specify <code>ALL</code> to remove all shards from the catalog. |
| <code>-shardspace shardspace_list</code> | Specify a comma-delimited list of names of shardspaces from which to remove all shards. |
| <code>-shardgroupshardgroup_list</code> | Specify a comma-delimited list of names of shardgroups from which to remove all shards. |
| <code>-force</code> | Remove one or more specified shards, even if they are inaccessible and/or contain chunks. It might result in a lower number of replicas or total unavailability for a certain range of data. |

 **Warning**

No chunks are moved before removing the shard which may result in data loss.

 **Warning**

Forced removal of a shard will also cause the removal of all shards that are replicas of the shard being forcibly removed.

Examples

Remove the shards from shardgroup GROUP1.

```
GDSCTL> remove shard -shardgroup group1
```

remove shardgroup

Removes a shardgroup from the shard catalog.

Syntax

```
remove shardgroup -shardgroup shardgroup_name
```

Options

Table 2-83 GDSCTL remove shardgroup Options

| Option | Description |
|---|---|
| <code>-shardgroupshardgroup_name</code> | Specify the name of the shardgroup to be removed. |

Usage Notes

Only a shardgroup that does not contain any shards can be removed.

Examples

Remove the GROUP1 shardgroup.

```
GDSCTL> remove shardgroup -shardgroup group1
```

remove shardspace

Removes a shardspace from the shard catalog.

Syntax

```
remove shardspace -shardspace shardspace_name
```

Options**Table 2-84 GDSCTL remove shardspace Options**

| Option | Description |
|---|---|
| <code>-shardspace <i>shardspace_name</i></code> | Specify the name of the shardspace to be removed. |

Usage Notes

Only a shardspace that does not contain any shards or shardgroups can be removed.

Examples

Remove the GOLD shardspace.

```
GDSCTL> remove shardspace -shardspace gold
```

restore backup

The restore backup command is used to restore a sharded database to a specific global restore point.

Syntax

```
restore backup [-restorepoint restore_point_name | -scn scn] [-cdb  
conn_str]  
[-catalog_name pdbrname] [-catalog_dbid dbid] [-restore_only | -  
recover_only]  
[-catpwd password] [-shard shard_list] [-async]
```

Options

Table 2-85 GDSCTL restore backup Options

| Option | Description |
|--|---|
| <code>-restorepoint <i>restore_point_name</i></code> | An sharded database global restore point that the specified list of shards will be restored to. |
| <code>-scn</code> | The SCN associated with a global restore point. This option cannot be used with option <code>"-restorepoint"</code> . However, to restore the SDB catalog to a specific restore point, the associated SCN must be used. Command <code>"LIST RESTOREPOINT"</code> can be used to list the available global restore points and their associated SCNs. |
| <code>-cdb</code> | A connect string to the CDB root of the catalog database. The provided user must have SYSDG privilege in the CDB root and SYSBACKUP privilege for all containers. This option should only be used to restore the SDB catalog. |
| <code>-catalog_name</code> | The PDB name of the SDB catalog. This option should only be used to restore the SDB catalog. |
| <code>-catalog_dbid</code> | The DBID of the SDB catalog container database. Both the catalog name and the DBID can be obtained from fixed view <code>v\$containers</code> . This option should only be used to restore the SDB catalog. |
| <code>-catpwd <i>password</i></code> | Password for user GSMCATUSER. Prompted if not specified. This password only needs to be specified once for this command in an entire GDSCTL session. |
| <code>-recover_only</code> | A flag. If specified, the command only recovers databases. This flag cannot be used with flag <code>"-restore_only"</code> . |
| <code>-restore_only</code> | A flag. If specified, the command restores the databases only without doing database recovery. This flag cannot be used with flag <code>"-recover_only"</code> . |
| <code>-shard <i>shard_list</i></code> | <i>shard_list</i> specifies a comma separated list of shard identifiers. They can be shard space, shard group or shard names. The default is <code>no shards</code> . |
| <code>-async</code> | When specified, all tasks to configure the backup for the shards will run in background. By default, the task will run in foreground. The task for the SDB catalog database will always run in foreground regardless of this flag setting. |

Examples

The following example restores the control files of shard `v1908c_cdb2_pdb1` to restore point `backup_before_db_maintenance`. The database must be in `NOMOUNT` state. This command alters the database to `MOUNT` state after it has restored the control file..

```
GDSCTL> restore backup -shard v1908c_cdb2_pdb1 -restorepoint
BACKUP_BEFORE_DB_MAINTENANCE -controlfile -catpwd gsm
executing command: SET until clause
```

```
Starting restore at 14-APR-20
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=441 device type=DISK
allocated channel: ORA_DISK_2
```

```

channel ORA_DISK_2: SID=202 device type=DISK

channel ORA_DISK_1: starting datafile backup set restore
channel ORA_DISK_1: restoring control file
channel ORA_DISK_1: reading from backup piece /ade/b/3998875997/oracle/dbs3/
V1908C/autobackup/2020_04_14/o1_mf_s_1037736781_h9dndfrd_.bkp
channel ORA_DISK_1: piece handle=/ade/b/3998875997/oracle/dbs3/V1908C/
autobackup/2020_04_14/o1_mf_s_1037736781_h9dndfrd_.bkp tag=TAG20200414T201301
channel ORA_DISK_1: restored backup piece 1
channel ORA_DISK_1: restore complete, elapsed time: 00:00:01
output file name=/ade/b/3998875997/oracle/dbs/ct_cf1.f
Finished restore at 14-APR-20

released channel: ORA_DISK_1
released channel: ORA_DISK_2

```

The next example restores the shard v1908c_cdb2_pdb1 to a restore point backup_before_db_maintenance.

```

GDSCTL> restore backup -shard v1908c_cdb2_pdb1 -restorepoint
BACKUP_BEFORE_DB_MAINTENANCE -catpwd gsm
executing command: SET until clause

Starting restore at 14-APR-20
starting full resync of recovery catalog
full resync complete
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=460 device type=DISK

channel ORA_DISK_1: starting datafile backup set restore
channel ORA_DISK_1: specifying datafile(s) to restore from backup set
...
channel ORA_DISK_1: restored backup piece 1
channel ORA_DISK_1: restore complete, elapsed time: 00:00:03
Finished restore at 14-APR-20

Starting recover at 14-APR-20
current log archived
using channel ORA_DISK_1

Creating automatic instance, with SID='yhox'
...
executing Memory Script
...

Oracle instance shut down

Removing automatic instance
Automatic instance removed
auxiliary instance file /ade/b/3998875997/oracle/dbs/V1908/datafile/
o1_mf_sysext_h9dx66s0_.dbf deleted
auxiliary instance file /ade/b/3998875997/oracle/dbs/V1908/datafile/
o1_mf_sysaux_h9dx66rp_.dbf deleted
auxiliary instance file /ade/b/3998875997/oracle/dbs/V1908/controlfile/
o1_mf_h9dx5klq_.ctl deleted

```

Finished recover at 14-APR-20

resume services

Resumes global service activity and traffic routing to the database, previously blocked by `SUSPEND SERVICES` command.

Syntax

```
GDSCTL> RESUME SERVICES -DATABASE target_db
```

Options

Table 2-86 GDSCTL resume services Options

| Option | Description |
|------------------|----------------------------------|
| <i>target_db</i> | Specify the name of the database |

Example

Resumes global service activity and traffic routing to the database:

```
GDSCTL> RESUME SERVICES -DATABASE dba
```

```
GDSCTL>
```

rman

Allow users to submit RMAN commands to a list of shards for execution.

Syntax

```
rman -shard shard_list [-check_syntax] [-from_cdb userid[/password]]
[-catpwd password] [-async] (-cmd_file cmdfile | <quote>rman-stmts<quote>)
```

Options

Table 2-87 GDSCTL rman Options

| Option | Description |
|---------------|---|
| -shard | Specifies a comma-delimited list of shard identifiers. Each shard identifier can be a region, shardspace, shardgroup or shard name. If the same name is used for a region, shardspace, shardgroup or shard, region takes the highest precedence followed by shardspace, shardgroup and then shard. For example, assume a sharded database has a shardspace and a shard both named "foo". When name "foo" is provided in the specified shard list, it is considered a shardspace and expanded to a list of the shards in the shardspace "foo". Two special words can be used for shard list: ALL and CATALOG. "ALL" means the sharded database catalog database and all the shards in the database while "CATALOG" means only the catalog database. For commands where this parameter is optional, if this value is not specified the value defaults to "ALL". |
| -check_syntax | If specified, it only checks syntax for the input RMAN commands. |
| -from_cdb | This option provides a CDB common user and a password in the form of "user/password". When this option is used, the provided RMAN commands will be run from the shard root container. The provided user must have SYSBACKUP privilege. |
| -catpwd | The GSMCATUSER password |
| -cmd_file | RMAN command file path |

Usage Notes

RMAN commands can be supplied either with a command file or directly in the command line. The syntax of the commands is the same as when they are entered at the RMAN prompt.

When the commands are supplied in the command line, they must be put in quotation marks. If the commands themselves contain single quotes, then double quotes should be used for the commands.

Some RMAN commands can only be executed from CDB root. If such a command is supplied, option `-FROM_CDB` must be used. Commands that should be executed from the shard PDBs and those that should be executed from shard CDB root cannot be supplied at the same time.

Some RMAN commands cannot be used inside command files, for example, `HOST`. These commands cannot be used here in the command line or through a command file.

Example

The command allows users to run various RMAN commands against the selected shards. The following example reports the objects needed to be backed up in the shards in shard group DBS1. Assuming that the shard PDB name is PDB1 for all the shards in shard group DBS1:

```
GDSCTL> rman -shard dbs1 -from_cdb 'report need backup of pluggable database pdb1';
```

Assuming that datafile 1 for shard SHARD1 needs to be backed up. The example below shows how to back up the datafile for shard SHARD1:

```
GDSCTL> rman -shard shard1 -from_cdb 'backup datafile 1';
```

run backup

Run Sharded Database (SDB) Backup Jobs.

Syntax

```
run backup [-tag tag] [-catpwd password] [-shard shard_list] [-async]
```

Options

Table 2-88 GDSCTL run backup Options

| Syntax | Description |
|--------------------------|--|
| -tag <i>tag</i> | A name for the backup. If not provided, a unique tag is automatically generated for the backup. The size limit for tag is 30. |
| -catpwd <i>password</i> | Password for user GSMCATUSER. Prompted if not specified. It needs to be specified once for the entire GDSCTL session. |
| -shard <i>shard_list</i> | <i>shard_list</i> is a comma separated list of shard identifiers. They can be shard space, shard group or shard names. The default is <i>no shards</i> |
| -async | When specified, all tasks to configure the backup for the shards will run in background. By default, the task will run in foreground. |

Examples

In the following example, a backup for the shards in shard space `db1` is created before a shard maintenance.

```
GDSCTL> run backup -tag backup_before_db_maintenance -shard db1 -catpwd gsm
```

```
Running on-demand backup for database "v1908b_cdb2_pdb1" ...
executing global script: full_backup_cdb
```

```
...
```

```
Starting Control File and SPFILE Autobackup at 14-APR-20
piece handle=/ade/b/3998875997/oracle/db3/V1908C/autobackup/2020_04_14/
ol_mf_s_1037736781_h9dndfrd_.bkp comment=NONE
Finished Control File and SPFILE Autobackup at 14-APR-20
```

```
Recovery Manager complete.
```

services

Retrieves information about the services that are registered with the specified global service manager.

Syntax

```
[status service | services] [-gsm gsm_name]
                               [-service service_name]
                               [-raw | -verbose | -support]
```

Options

Table 2-89 GDSCTL services Options

| Option | Description |
|---|--|
| <code>-gsm <i>gsm_name</i></code> | Specify the name of a global service manager. If the name is not specified, then GDSCTL uses the current global service manager name for the session (specified with the GDSCTL <code>set gsm</code> command). |
| <code>-raw</code> | If specified, GDSCTL output is presented in a raw, non-parsed format. |
| <code>-service <i>service_name</i></code> | Specify a fully qualified service name. If the service name is not specified, then the information about all the services registered with the global service manager is retrieved. |
| <code>-support</code> | If specified, GDSCTL output displays additional information. |
| <code>-verbose</code> | Enables verbose output mode. |

Usage Notes

You must run this command on the host where the global service manager for which you want to retrieve service information resides.

You must have the privileges of the user who started the global service manager to run this command.

If `-service` is not specified, then information for all global services is displayed.

Example

Display information about the services registered with global service manager `mygsm`:

```
GDSCTL> services -gsm mygsm
```

The `gdsctl services` command returns output similar to the following:

```
GDSCTL>services -gsm mygsm
Service "localsvc.dbpoolora.oradbcloud" has 2 instance(s). Affinity: LOCALPREF
  Instance "dbpoolora%1", name: "gdscat", db: "gdscat", region: "regionora",
  status: ready.
  Instance "dbpoolora%11", name: "gdscat2", db: "gdscat2", region: "regionora",
  status: ready.
Service "sales_report1.dbpoolora.oradbcloud" has 2 instance(s). Affinity:
LOCALONLY
```

```

Instance "dbpoolora%1", name: "gdscat", db: "gdscat", region: "regionora",
status: ready.
Instance "dbpoolora%11", name: "gdscat2", db: "gdscat2", region: "regionora",
status: ready.
Service "sales_report2.dbpoolora.oradbcloud" has 2 instance(s). Affinity: ANYWHERE
Instance "dbpoolora%1", name: "gdscat", db: "gdscat", region: "regionora",
status: ready.
Instance "dbpoolora%11", name: "gdscat2", db: "gdscat2", region: "regionora",
status: ready.

```

Note

Affinity values can be `LOCALONLY` when the service locality is defined as `local_only`, `LOCALPREF` when the service locality is defined as `local_only` with the `region_failover` option enabled, and `ANYWHERE` when the service locality is defined as `anywhere`.

Display the status of `mtgly_report` service:

```
GDSCTL>services -service mtgly_report.sales.oradbcloud
```

Returns output similar to the following:

```

Service "mtgly_report.sales.oradbcloud" has 1 instance(s). Affinity:
ANYWHERE
Instance "sales%1", name: "debug", db: "debug", region: "eastcoast",
status: ready.

```

set dataguard_property

Dynamically updates the value of a specified property of a broker configuration or database.

Syntax

```

set dataguard_property {-shardspace name | -brokerconfig name | -shard name
|
-shardgroup name} [-reset] [-scope { configuration |
database}]
'property_name'=property_value

```

Options

Table 2-90 GDSCTL set dataguard_property Options

| Syntax | Description |
|---------------------------|-----------------------------------|
| -shardspace <i>name</i> | The name of the shardspace. |
| -brokerconfig <i>name</i> | Broker configuration identifier. |
| -shard <i>name</i> | The name of the shard. |
| -shardgroup <i>name</i> | The name of the shardgroup. |
| -reset | Resets property to default value. |

Table 2-90 (Cont.) GDSCTL set dataguard_property Options

| Syntax | Description |
|--------|---|
| -scope | Defines the scope for property: database or broker configuration. |

Usage Notes

A database property of a member of a Data Guard Broker configuration can be updated by using the `-SHARD` option. The `-SHARDGROUP` option allows users to update a database property of all databases in a specified shardgroup.

A property of the entire Data Guard Broker configuration can be updated by using the `-BROKERCONFIG` or `-SHARDSPACE` option.

See Oracle Data Guard Broker guide for the complete list of the Broker configuration and database properties.

If none of `-shardspace/-shardgroup/-shard/-brokerconfig` is specified, then the command will act on all broker configs or databases in the catalog.

If the user specifies `-shardspace`, then the command will only act on the broker configs / shards in that shardspace. Likewise with `-shard` and `-shardgroup`.

Specifying `-scope` configuration will override the default scope of 'database' for `-shard` and `-shardgroup`. Likewise, specifying `-scope database` will override the default scope of 'configuration' for `-shardspace` and `-brokerconfig`

Example

```
GDSCTL> set dataguard_property -shard db32 'archivelagtarget'=1200
```

```
set dataguard_property -shardgroup west 'archivelagtarget'=1200
```

```
set dataguard_property -brokerconfig conf_1 'logxptmode'=async
```

```
set dataguard_property -shard us 'logxptmode'=async
```

```
set dataguard_property -shardspace silver 'logxptmode'=async
```

set gsm

Sets the global service manager for the current session.

This command establishes to which global service manager the successive commands apply. The specified global service manager name is resolved in the `gsm.ora` configuration file.

Syntax

```
set gsm -gsm gsm_name
```

Options

Table 2-91 GDSCTL set gsm Options

| Syntax | Description |
|----------------------------|--|
| <code>-gsm gsm_name</code> | Specify the name of the global service manager to work with in the current session. If you do not specify a specific global service manager, then GDSCTL uses the default global service manager name of GSMORA. |

Usage Notes

You must run this command on the host where the global service manager that you want to set for the current session resides.

You must have the privileges of the user who started the global service manager to run this command.

Example

Set the global service manager for the current session to `gsm1`.

```
GDSCTL> set gsm -gsm gsm1
```

set inbound_connect_timeout

Sets the `INBOUND_CONNECT_TIMEOUT` listener parameter.

Syntax

```
set inbound_connect_timeout timeout_value
                               [-gsm gsm_name]
                               [-save_config | -config_only]
```

Options

Table 2-92 GDSCTL set inbound_connect_timeout Options

| Option | Description |
|----------------------------|---|
| <code>-config_only</code> | Update GSM.ORA only, without trying to connect to a running global service manager instance. |
| <code>-gsm gsm_name</code> | Specify the name of the global service manager that you want to start. If you do not specify a specific global service manager, then GDSCTL uses the current global service manager name for the session (specified with the command set gsm). |
| <code>-save_config</code> | Store configuration changes to GSM.ORA. |
| <code>timeout_value</code> | Specify in seconds the connection timeout value. |

Usage Notes

- You must run this command on the host where the global service manager for which you want to set the `INBOUND_CONNECT_TIMEOUT` listener parameter resides
- You must have the privileges of the user who started the global service manager to run this command
- By default, parameter values changes remain in effect until the global service manager is shut down.

Example

Set the `INBOUND_CONNECT_TIMEOUT` listener parameter for `mygsm` to time out in 60 seconds:

```
GDSCTL> set inbound_connect_timeout -gsm mygsm 60
```

set log_status

Sets the `LOG_STATUS` listener parameter.

Syntax

```
set log_status ON|OFF
           [-gsm gsm_name]
           [-save_config | -config_only]
```

Options

Table 2-93 GDSCTL set log_status Options

| Option | Description |
|---------------|---|
| ON OFF | Turns listener logging on or off. |
| -config_only | Update GSM.ORA only, without trying to connect to a running global service manager instance. |
| -gsm gsm_name | Specify the name of the global service manager that you want to start. If you do not specify a specific global service manager, then GDSCTL uses the current global service manager name for the session (specified with the command set gsm). |
| -save_config | Store configuration changes to GSM.ORA. |

Usage Notes

- You must run this command on the host where the global service manager for which you want to set the `LOG_STATUS` listener parameter resides
- You must have the privileges of the user who started the global service manager to run this command
- By default, parameter values changes remain in effect until the global service manager is shut down.

Example

Set the `LOG_STATUS` listener parameter to ON.

```
GDSCCTL> set log_status on -save_config
```

set outbound_connect_timeout

Sets the `OUTBOUND_CONNECT_TIMEOUT` listener parameter.

Syntax

```
set outbound_connect_timeout timeout_value
                               [-gsm gsm_name]
                               [-save_config | -config_only]
```

Options

Table 2-94 GDSCCTL set outbound_connect_timeout Options

| Option | Description |
|----------------------------|--|
| <code>timeout_value</code> | Specify in seconds the connection timeout value. |
| <code>-config_only</code> | Update <code>GSM.ORA</code> only, without trying to connect to a running global service manager instance. |
| <code>-gsm gsm_name</code> | Specify the name of the global service manager that you want to start. If you do not specify a specific global service manager, then GDSCCTL uses the current global service manager name for the session (specified with the command set gsm). |
| <code>-save_config</code> | Store configuration changes to <code>GSM.ORA</code> . |

Usage Notes

- You must run this command on the host where the global service manager for which you want to set the `OUTBOUND_CONNECT_TIMEOUT` listener parameter resides
- You must have the privileges of the user who started the global service manager to run this command
- By default, parameter values changes remain in effect until the global service manager is shut down.

Example

Set the `OUTBOUND_CONNECT_TIMEOUT` listener parameter for `mygsm` to time out in 60 seconds:

```
GDSCCTL> set outbound_connect_timeout -gsm mygsm 60
```

set trace_level

Sets the trace level for the listener associated with the specified global service manager.

Syntax

```
set trace_level [-gsm gsm_name]
                trace_level
```

Options

Table 2-95 GDSCTL set trace_level Options

| Option | Description |
|----------------------------|---|
| <code>-gsm gsm_name</code> | Specify the name of the global service manager. If the name is not specified, then GDSCTL uses the current global service manager name for the session (specified with the GDSCTL <code>set gsm</code> command). |
| <code>trace_level</code> | Specify the trace level for the global service manager listener. Valid values are USER - provides traces to identify user-induced error conditions ADMIN - provides traces to identify installation-specific problems SUPPORT - provides trace with troubleshooting information for Oracle Support Services OFF - provides no tracing |

Usage Notes

- You must run this command on the host where the global service manager for which you want to set the listener trace level resides.
- You must have the privileges of the user who started the global service manager to run this command.

Example

Set the trace level for all listeners associated with `mygsm` to ADMIN

```
GDSCTL> set trace_level -gsm mygsm ADMIN
```

set trc_level

Sets the TRC_LEVEL listener parameter.

Syntax

```
set trc_level trace_level
           [-gsm gsm_name]
           [-save_config | -config_only]
```

Options

Table 2-96 GDSCTL set trc_level Options

| Option | Description |
|----------------------|---|
| <i>trace_level</i> | Specify the trace level for the global service manager listener. Valid values are USER provides traces to identify user-induced error conditions ADMIN provides traces to identify installation-specific problems SUPPORT provides trace with troubleshooting information for Oracle Support Services OFF provides no tracing |
| -config_only | Update GSM.ORA only, without trying to connect to a running global service manager instance. |
| -gsm <i>gsm_name</i> | Specify the name of the global service manager that you want to start. If you do not specify a specific global service manager, then GDSCTL uses the current global service manager name for the session (specified with the command set gsm). |
| -save_config | Store configuration changes to GSM.ORA. |

Usage Notes

- You must run this command on the host where the global service manager for which you want to set the LOG_STATUS listener parameter resides
- You must have the privileges of the user who started the global service manager to run this command
- By default, parameter values changes remain in effect until the global service manager is shut down.

Example

Set the TRC_LEVEL listener parameter to SUPPORT.

```
GDSCTL> set trc_level support
```

show ddl

Shows DDL statements execution status.

Syntax

```
show ddl {[-ddl ddl_id] [-count cnt] | [-failed_only]}
        [-support]
        [-verbose]
```

Options

Table 2-97 GDSCTL show ddl Options

| Option | Description |
|--------------------|--|
| -count <i>cnt</i> | The maximum number of entries to display. |
| -ddl <i>ddl_id</i> | DDL numeric identifier. |
| -failed_only | Use this option to display only errored out statements. |
| -support | If specified, GDSCTL output displays additional support information. |
| -verbose | Enable verbose mode. |

Usage Notes

If `-DDL` and `-COUNT` are both unspecified, the command returns only the last 10 DDL statements.

If `-DDL` is specified but `-COUNT` is not, the command returns detailed information about the DDL statement. The `-COUNT` option defines the maximum number of DDLs to display.

Examples

```
GDSCTL> show ddl -count 20
```

Note

The `show ddl` command output might be truncated. You can run `SELECT ddl_text FROM gsmadmin_internal.ddl_requests` on the catalog to see the full text of the statements.

split chunk

Splits each of the specified chunks into two chunks with an equal number of records. After the split, the chunks remain in the same shard.

Syntax

```
split chunk -chunk chunk_id_list
            [-shardspace shard_space_list]
```

Options

Table 2-98 GDSCTL split chunk Options

| Option | Description |
|-----------------------------|--|
| -chunk <i>chunk_id_list</i> | Specify a comma-separated list of numeric chunk identifiers. |

Table 2-98 (Cont.) GDSCTL split chunk Options

| Option | Description |
|--|--|
| <code>-shardspace <i>shard_space_list</i></code> | Specify a list of shardspace names in which to split the specified chunks. |

Usage Notes

This command can only be used with system-managed sharding. For user-defined sharding, `ALTER TABLE` is used to split a partition of the root (parent) table.

Merging of chunks is not supported.

Examples

Split chunks 3, 4, and 5.

```
GDSCTL> split chunk -chunk 3,4,5
```

sql

Executes a SQL statement or a PL/SQL stored procedure against a sharded database.

Syntax

```
sql "sql_statement"
```

Options**Table 2-99 GDSCTL sql Options**

| Option | Description |
|-----------------------------------|--|
| <code><i>sql_statement</i></code> | Enter the SQL statement or PL/SQL stored procedure to be executed. Do not include a semi-colon (;) after the SQL statement to be executed. |

Usage Notes

This command can only be executed against a sharded GDS pool. The statements are executed in the GDS catalog database and are then broadcast to all shards in the pool. Since there can be only one sharded pool in a GDS configuration, all SQL statements executed on the catalog database are applied to this pool, if it exists.

Database objects created by this command in the catalog database are used as a schema of the sharded database and are not intended to store user data. The only exception is tables duplicated on all shards (reference tables) – they are populated with data in the catalog database.

SELECT statements are not allowed as the parameter.

The SQL statement or PL/SQL stored procedure to be executed must be enclosed in double quotation marks.

If the string that GDSCTL passes to PL/SQL contains a filename, then the filename must be enclosed in single quotation marks.

Do not include a semi-colon (;) after the SQL statement to be executed.

Examples

Using the `gdsctl sql` command.

```
GDSCTL> sql "CREATE TABLESPACE SET ts1 IN SHARDGROUP sgr1"
```

start gsm

Starts a specific global service manager.

Syntax

```
start gsm [-gsm gsm_name] [-validate_network]
```

Options

Table 2-100 GDSCTL start gsm Options

| Option | Description |
|-----------------------------------|---|
| <code>-gsm <i>gsm_name</i></code> | Specify the name of the global service manager that you want to start. If you do not specify a specific global service manager, then GDSCTL uses the current global service manager name for the session (specified with the command set gsm). |
| <code>-validate_network</code> | This flag enables several network validation checks, including checking network connectivity between hosts and checking VNCR entries are valid. |

Usage Notes

- You must run GDSCTL on the same host where the global service manager you want to start is located.
- You must have operating system privileges on the computer where you want to start the global service manager to run this command.

Example

Start the global service manager `gsm1` on the local host.

```
GDSCTL> start gsm -gsm gsm1
```

start observer

Starts specific services.

Syntax

```
start observer -database db_name
                [-timeout seconds]
```

Options

Table 2-101 GDSCTL start observer Options

| Option | Description |
|--------------------------|---|
| -database <i>db_name</i> | The name of the database. |
| -timeout <i>seconds</i> | The global service manager requests timeout in seconds. |

Usage Notes

TIMEOUT (default 15) represents the time between when the shard director/global service manager receives requests and starts the observer. See *Oracle Globally Distributed AI Database* for the automatic rules for choosing the right region for the shard director (global service manager) server to start the observer. If shard director servers are not running in this region, the observer is not started.

Example

```
GDSCTL> start observer -database mydb
```

start ru

Starts a specified replication unit.

Syntax

```
start ru -ru ru_id [-DATABASE db]
```

Options

Table 2-102 GDSCTL start ru Options

| Option | Description |
|-----------|---------------------------|
| -database | The name of the database. |
| -ru | Replication unit ID |

Usage Notes

If the database is not specified, the `start ru` command runs on all available replicas of the specified replication unit.

Examples

Add the shard to shardgroup GROUP1 of the DB11 database.

```
GDSCTL> start ru -ru 1 -database mydb1
```

start service

Starts specific services.

Syntax

```
start service [-gdspool gdspool_name]
              -service service_name
              [{-database db_name |
               -override [-pwd password] -connect connect_identifier}]
```

Options

Table 2-103 GDSCTL start service Options

| Option | Description |
|------------------------------------|---|
| -database <i>db_name</i> | Specify the name of the database on which you want to start the service. If you do not specify this option, then GDSCTL starts the services on all preferred databases. |
| -connect <i>connect_identifier</i> | Specify an Oracle Net connect descriptor or net service name that resolves to a connect descriptor. |
| -gdspool <i>gdspool_name</i> | Specify the name of the database pool in which the services that you want to start are located. If not specified and there is only one <i>gdspool</i> with access granted to the user, it is used as the default <i>gdspool</i> . |
| -override | This option causes the command to run without updating the global service manager catalog. You can use this option when the catalog database is unavailable. During normal operation, you should not use this option. |
| -pwd <i>password</i> | Specify the password of the GSMUSER in the specified database. |
| -service <i>service_name</i> | Specify a comma-delimited list of global service names. If you do not use this option, then GDSCTL starts all the services in the database pool. |

Usage Notes

You must connect to the catalog database as a user with the pool administrator privileges, using the command [connect](#) before running this command.

Before starting services which run on administrator-managed databases, they must be modified for those databases to stipulate which instances should run the service. Please refer to the `-modify_instances` parameter of the `modify service` command.

Example

Start the service `SALES_REPORT`, located in the `READERFARM` database pool.

```
GDSCTL> start service -gdspool readerfarm -service sales_report
```

① **See Also**

[#unique_142](#)

status

Displays the running status and runtime information for the global service manager.

Syntax

```
status [-gsm gsm_name] [-raw|-verbose|-support]
```

Options

Table 2-104 GDSCTL status Options

| Option | Description |
|----------------------|---|
| -gsm <i>gsm_name</i> | Specify the name of a global service manager to check. If the name is not specified, then GDSCTL uses the current global service manager name for the session (specified with the GDSCTL <code>set gsm</code> command). |
| -raw | If specified, GDSCTL output is presented in a raw non-parsed format. |
| -support | If specified, GDSCTL output displays additional information. |
| -verbose | Enable verbose mode. |

Example

```
GDSCTL> status
```

The command returns output similar to the following.

```
Alias MYGSM
Version 12.1.0.0.2
Start Date 03-JUL-2012 16:48:54
Trace Level support
Listener Log File /u01/ORACLE/mygsm/alert/log.xml
Listener Trace File /u01/ORACLE/mygsm/trace/ora_14816_47568108067776.trc
Endpoint summary (ADDRESS=(HOST=mymv.us.hq.com)(PORT=1523)(PROTOCOL=tcp))
GSMOCI Version 0.1.8
Mastership Y
Connected to GDS catalog Y
Process Id 14818
Number of reconnections 0
Pending tasks. Total 0
Tasks in process. Total 0
Regional Mastership TRUE
Total messages published 28599
```

```

Time Zone -07:00
Orphaned Buddy Regions:
None
GDS region regionora

```

status backup

View the detailed state on the scheduled backup jobs in the specified shards.

Syntax

```

status backup [-start_time t1] [-end_time t2]
              [-catpwd password]
              [-shard shard_list]
              [-READ_LOG rman_log]
              [-OUTPUT_LOG output_log]
              [-DELETE_LOG]
              [-DELETE_LOG_ONLY]

```

Options

Table 2-105 GDSCTL status backup Options

| Option | Description |
|--------------------------------|---|
| <code>-start_time time1</code> | The command lists run details of the automated backup jobs that started on or after this time. It must be specified in the format <code>YYYY-MM-DD HH:MM:SS[.FFF]</code> where <code>.FFF</code> is a fraction of a second in the precision of milliseconds. |
| <code>-end_time time2</code> | If specified, only backups usable to restore database control files to a specific restore point are listed. |
| <code>-catpwd password</code> | Password for user <code>GSMCATUSER</code> . Prompted if not specified. This password only needs to be specified once for this command in an entire <code>GDSCTL</code> session. |
| <code>-shard shard_list</code> | <code>shard_list</code> specifies a comma separated list of shard identifiers. They can be shard space, shard group or shard names. The default is <code>all shards</code> . |
| <code>-delete_log</code> | Delete the specified RMAN output file on the shard server. |
| <code>-delete_log_only</code> | Delete the specified RMAN output file on the shard server without reading it. |
| <code>-output_log</code> | The specified file must not exist. If the file is not specified with an absolute path, it will be created in the current working directory. The RMAN log will be saved into this file without being displayed in the console. |
| <code>-read_log</code> | Read the specified RMAN log from the specified shard and display it in the console or save it into the file specified with the option <code>-OUTPUT_LOG</code> . The RMAN log name usually comes from the output of this command when it is run without the option <code>-READ_LOG</code> . |

Examples

The following example shows the job state and all job run details from the sharded database (SDB) catalog and the primary shard `rdbmsb_cdb2_pdb1`.

```
GDSCTL> status backup -catpwd -shard catalog,rdbmsb_cdb2_pdb1;  
"GSMCATUSER" password:***
```

Retrieving scheduler backup job status for database "rdbms" ...

Jobs:

```
Incremental Level 0 backup job is enabled  
Job schedule start time: 2020-07-27 00:00:00.000 -0400  
Job repeat interval: freq=daily;interval=1  
Incremental Level 1 backup job is enabled  
Job schedule start time: 2020-07-27 00:00:00.000 -0400  
Job repeat interval: freq=minutely;interval=60  
Global restore point create job is enabled  
Job schedule start time: 2020-07-27 23:59:55.960 -0400  
Job repeat interval: freq=hourly
```

Run Details:

```
Incremental Level 1 backup job status: SUCCEEDED  
Job run actual start time: 2020-07-26 14:00:00.177 -0400  
Job run slave process ID: 9023  
Incremental Level 1 backup job status: SUCCEEDED  
Job run actual start time: 2020-07-26 22:00:01.305 -0400  
Job run slave process ID: 59526  
...  
Global restore point create job status: SUCCEEDED  
Job run actual start time: 2020-07-27 15:28:37.603 -0400  
Job run slave process ID: 44227  
...  
Global restore point create job status: SUCCEEDED  
Job run actual start time: 2020-07-27 17:28:38.251 -0400  
Job run slave process ID: 57611
```

Retrieving scheduler backup job status for database "rdbmsb_cdb2_pdb1" ...

Jobs:

```
Incremental Level 0 backup job is enabled  
Job schedule start time: 2020-07-28 00:00:00.000 -0400  
Job repeat interval: freq=daily;interval=1  
Incremental Level 1 backup job is enabled  
Job schedule start time: 2020-07-28 00:00:00.000 -0400  
Job repeat interval: freq=minutely;interval=60
```

Run Details:

```
Incremental Level 1 backup job status: SUCCEEDED  
Job run actual start time: 2020-07-26 14:00:00.485 -0400  
Job run slave process ID: 9056  
...  
Incremental Level 1 backup job status: SUCCEEDED  
Job run actual start time: 2020-07-27 14:33:42.702 -0400  
Job run slave process ID: 9056  
Incremental Level 0 backup job status: SUCCEEDED
```

Job run actual start time: 2020-07-27 00:00:01.469 -0400
Job run slave process ID: 75176

status database

Displays the runtime status of databases, such as registration information, services, and so on.

Syntax

```
{status database | databases} [-gsm gsm_name]
                               [-database db_name]
                               [-gdspool gdspool_name]
                               [-raw | -support | -verbose]
```

Options

Table 2-106 GDSCTL status database Options

| Option | Description |
|------------------------------|---|
| -database <i>db_name</i> | Specify the name of the database on which to check status |
| -gdspool <i>gdspool_name</i> | Specify the name of the database pool. If not specified and there is only one gdspool with access granted to the user, it is used as the default gdspool. |
| -gsm <i>gsm_name</i> | Specify the name of a global service manager to check. If the name is not specified, then GDSCTL uses the current global service manager name for the session (specified with the GDSCTL <code>set gsm</code> command). |
| -raw | If specified, GDSCTL output is presented in a raw, non-parsed format. |
| -support | If specified, GDSCTL output displays additional support information. |
| -verbose | Enable verbose output mode. |

Usage Notes

You must connect to the catalog database as a user with the pool administrator privileges, using the command [connect](#) before running this command.

This command requires a locally started global service manager. If `-gsm` is not specified for `STATUS DATABASE`, then the currently connected global service manager name is used by default.

Example

Display the status of all databases:

```
GDSCTL> status database
```

The `gdctl status database` command returns output similar to the following:

```
Database: "dbcat1" Registered: Y State: Ok ONS: N. Role: PRIMARY Instances: 1
Region: east
```

```

Service: "sales_svc2" Globally started: N Started: N
          Scan: Y Enabled: Y Preferred: Y
Service: "sales_svc1" Globally started: Y Started: Y
          Scan: N Enabled: Y Preferred: Y
Registered instances:
  sales%11
Database: "dbcat2" Registered: Y State: Ok ONS: N. Role: PRIMARY Instances: 1
Region: east
Service: "sales_svc2" Globally started: N Started: N
          Scan: Y Enabled: Y Preferred: Y
Service: "sales_svc1" Globally started: Y Started: Y
          Scan: N Enabled: Y Preferred: Y
Registered instances:
  sales%1

```

status gsm

Displays the status of a specific global service manager.

Syntax

```

status (gsm)? [-gsm gsm_name]
              [-raw | -verbose | -support]

```

Options

Table 2-107 GDSCTL status gsm Options

| Option | Description |
|----------------------------|---|
| <code>-gsm gsm_name</code> | Specify the name of a global service manager to check. If the name is not specified, then GDSCTL uses the current global service manager name for the session (specified with the GDSCTL <code>set gsm</code> command). |
| <code>-raw</code> | If specified, GDSCTL output is presented in a raw, non-parsed format. |
| <code>-support</code> | If specified, GDSCTL output displays additional support information. |
| <code>-verbose</code> | Enable verbose output mode. |

Usage Notes

You must run GDSCTL on the same host where the global service manager for which you want to display the status is located.

You must have operating system privileges on the computer where you want to display the global service manager status to run this command.

Example

Display status of mygsm:

```
GDSCTL> status gsm -gsm mygsm
```

The `gdsctl status gsm` command returns output similar to the following:

```
Alias MYGSM
Version 12.1.0.0.2
Start Date 03-JUL-2012 16:48:54
Trace Level support
Listener Log File /u01/ORACLE/mygsm/alert/log.xml
Listener Trace File /u01/ORACLE/mygsm/trace/ora_14816_47568108067776.trc
Endpoint summary (ADDRESS=(HOST=mymv.us.hq.com)(PORT=1523)(PROTOCOL=tcp))
GSMOCI Version 0.1.8
Mastership Y
Connected to GDS catalog Y
Process Id 14818
Number of reconnections 0
Pending tasks. Total 0
Tasks in process. Total 0
Regional Mastership TRUE
Total messages published 28599
Time Zone -07:00
Orphaned Buddy Regions:
None
GDS region regionora
```

status routing

Displays the runtime routing information status.

Syntax

```
status routing [-by_chunk | -by_instance] [-gsm
gsm_name]
[-raw|-verbose|-support]
```

Options

Table 2-108 GDSCTL status routing Options

| Option | Description |
|--------------|--|
| -by_chunk | Group routing table output by chunk. |
| -by_instance | Group routing table information by registered instance (default). |
| -gsm | GSM name |
| -raw | If specified, GDSCTL output is presented in a raw non-parsed format. |
| -support | If specified, GDSCTL output displays additional information. |
| -verbose | Enable verbose mode. |

Usage Notes

By default the registered chunks are grouped by instance

Example

```
GDSCTL> status routing
```

```
GDSCTL>
```

status ru

Displays runtime information about replication units for Oracle Globally Distributed AI Database native Raft replication.

Syntax

```
(STATUS REPLICATION | STATUS RU | RU)
    [-show_offline]
    [-savename]
    [-sort]
    [-gsm gsmname]
    [-catpwd pwd]
    [-wpwd wpwd]
    [-show_chunks]
    [-ru ru]
    [-leaders]
    [-database shard_name]
    [-show_errors [-all]]
    [-show_stats]
```

Options

Table 2-109 GDSCTL status ru Options

| Option | Description |
|---------------|---|
| -all | If not specified, only errors since last recovery are shown. |
| -catpwd | GSMCATUSER password. Both -CATPWD and -WPWD should be specified if GDSCTL doesn't share a HOME with the GSM (shard director). |
| -database | Shard name |
| -gsm | GSM (shard director) name |
| -leaders | Only leader information is displayed. |
| -ru | Replication unit ID |
| -show_chunks | Show chunk distribution across all replication units. |
| -show_errors | Show replication unit errors. |
| -show_offline | Show the list of offline (down) shards that have that RU. |
| -show_stats | Show usage statistics |
| -sort | Sort the output by replication unit ID. |

Table 2-109 (Cont.) GDSCTL status ru Options

| Option | Description |
|--------|-----------------|
| -wpwd | Wallet password |

Examples

```
GDSCTL> status ru
Replication units
```

```
-----
Database                               RU#  Role    Term  Log Index  Apply SCN
LWM SCN  On-disk SCN
-----  -----
-----  -----
cdbsh1_sh1                               1    Leader   2      315471
1      0      304531
cdbsh1_sh1                               2    Follower 1      456282    451835
452939    456611
cdbsh1_sh1                               3    Follower 2      262706    260479
261605    262709
cdbsh2_sh2                               2    Leader   1      456282
0      0      446475
cdbsh2_sh2                               1    Follower 2      315471    313342
14393    315479
cdbsh2_sh2                               3    Follower 2      262706    260479
261605    262709
cdbsh3_sh3                               3    Leader   2      262706
1      0      252741
cdbsh3_sh3                               1    Follower 2      315471    314395
314393    315477
cdbsh3_sh3                               2    Follower 1      456282    452941
452939    456583
```

```
GDSCTL> status ru -ru 2
Replication units
```

```
-----
Database                               RU#  Role    Term  Log Index  Apply LogIdx LWM
LogIdx On-disk LogIdx Status
-----  -----
-----  -----
den1b_cdb2_pdb1                         2    Leader   1    1      0
0      2      Ok
den1d_cdb4_pdb1                         2    Follower 1    1      0
0      1      Ok
```

```
den1c_cdb3_pdb1          2  Follower  1  1  0
0          1          Ok
```

```
GDSCTL> status ru -ru 2 -show_chunks
```

```
Chunks
```

```
-----
RU#          From      To
---          ----      --
2           8          8
```

```
Replication units
```

```
-----
Database          RU#  Role      Term Log Index Apply LogIdx LWM
LogIdx On-disk LogIdx Status
-----          ---  ----      ---- -
den1b_cdb2_pdb1   2    Leader    1   1   0
0          2          Ok
den1d_cdb4_pdb1   2    Follower  1   1   0
0          1          Ok
den1c_cdb3_pdb1   2    Follower  1   1   0
0          1          Ok
```

```
GDSCTL> ru -sort -show_chunks
```

```
Chunks
```

```
-----
RU#          From      To
---          ----      --
1           1          7
2           8          8
```

```
Replication units
```

```
-----
Database          RU#  Role      Term Log Index Status
-----          ---  ----      ---- -
den1b_cdb2_pdb1   1    Leader    1   3658  Ok
den1c_cdb3_pdb1   1    Follower  1   3658  Ok
den1d_cdb4_pdb1   1    Follower  1   3658  Ok
den1b_cdb2_pdb1   2    Leader    1   1     Ok
den1c_cdb3_pdb1   2    Follower  1   1     Ok
den1d_cdb4_pdb1   2    Follower  1   1     Ok
```

```
GDSCTL> ru -sort -show_errors
```

```
Replication units
```

```
-----
Database          RU#  Role      Term Log Index Status
-----          ---  ----      ---- -
den1b_cdb2_pdb1   1    Leader    1   3658  Ok
den1c_cdb3_pdb1   1    Follower  1   3658  Ok
den1d_cdb4_pdb1   1    Follower  1   3658  Ok
den1b_cdb2_pdb1   2    Leader    1   1     Ok
den1c_cdb3_pdb1   2    Follower  1   1     Ok
den1d_cdb4_pdb1   2    Follower  1   1     Ok
```

ERRORS

```
GDSCTL> ru -sort -database den1b_cdb2_pdb1
```

Replication units

| Database | RU# | Role | Term | Log | Index | Status |
|-----------------|-----|--------|------|-------|-------|--------|
| ----- | --- | ---- | ---- | ----- | ----- | ----- |
| den1b_cdb2_pdb1 | 1 | Leader | 1 | 3658 | | Ok |
| den1b_cdb2_pdb1 | 2 | Leader | 1 | 1 | | Ok |

status service

Displays the status of a specific service.

Syntax

```
{status service | services} [-gsm gsm_name]
                             [-service service_name]
                             [{-raw|-verbose|-support}]
```

Options

Table 2-110 GDSCTL status service Options

| Option | Description |
|------------------------------|--|
| -gsm <i>gsm_name</i> | Specify the name of a global service manager. If the name is not specified, then GDSCTL uses the current global service manager name for the session (specified with the GDSCTL <code>set gsm</code> command). |
| -raw | Used by oracle internal components. |
| -service <i>service_name</i> | Specify a comma-delimited list of global service names. If you do not specify any services, then GDSCTL displays the status of all services in the database pool. |
| -support | Display more detailed information concerning load balancing. |
| -verbose | Display extra information related to load balancing. |

Usage Notes

- You must connect to the catalog database as a user with the pool administrator privileges, using the command [connect](#) before running this command.
- This command is similar to [services](#).

Example

Display the status of service `sales_report1.sales.oradbcloud`:

```
GDSCTL> status service -service sales_report1.sales.oradbcloud
```

The `gdsctl status service` command returns output similar to the following:

```
Service "sales_report1.sales.oradbcloud" has 3 instance(s). Affinity: ANYWHERE
  Instance "sales%1", name: "dbcat2", db: "dbcat2", region: "east",
  status: ready.
  Instance "sales%11", name: "dbcat1", db: "dbcat1", region: "west",
  status: ready.
  Instance "sales%31", name: "dbcat3", db: "dbcat3", region: "east",
  status: ready.
```

stop gsm

Stops a specific global service manager.

Syntax

```
stop gsm [-gsm gsm_name]
```

Options

Table 2-111 GDSCTL stop gsm Options

| Option | Description |
|-----------------------------------|---|
| <code>-gsm <i>gsm_name</i></code> | Specify the name of a global service manager you want to stop. If you do not specify a specific global service manager, then GDSCTL uses the current global service manager name for the session (specified with the command set gsm). |

Usage Notes

- You must run GDSCTL on the same host where the global service manager that you want to stop is located.
- You must have operating system privileges on the computer where you want to start the global service manager to run this command.

Example

Stop the global service manager `gsm1` on the local host.

```
GDSCTL> stop gsm -gsm gsm1
```

stop ru

Stops a specified replication unit.

Syntax

```
stop ru -ru ru_id [-database db]
```

Options

Table 2-112 GDSCTL stop ru Options

| Option | Description |
|----------------------|---------------------------|
| -database | The name of the database. |
| -pwd <i>password</i> | Replication unit ID |

Usage Notes

If the database is not specified, the `stop ru` command runs on all available replicas of the specified replication unit.

Examples

```
GDSCTL> stop ru -ru 1 -database mydb1
```

stop service

Stops the specified global services.

Syntax

```
stop service [-gdspool gdspool_name]
             [-service service_name_list]
             [{-database db_name |
              -override -connect connect_identifier [-pwd password]}]
             [-force]
             [-drain_timeout time]
             [-stop_option {NONE|IMMEDIATE|TRANSACTIONAL}]
```

Options

Table 2-113 GDSCTL stop service Options

| Option | Description |
|------------------------------------|---|
| -connect <i>connect_identifier</i> | Specify an Oracle Net connect descriptor or net service name that resolves to a connect descriptor for the database (or shard). |
| -database <i>db_name</i> | Specify the name of the database on which you want to stop the service. If you do not specify this option, then GDSCTL stops the services on all databases on which the service is currently running. |
| -drain_timeout | Set drain time in seconds. |

Table 2-113 (Cont.) GDSCTL stop service Options

| Option | Description |
|---|---|
| <code>-force</code> | If you use this option, then GDSCTL disconnects all sessions when the service is stopped, requiring the sessions using the service to reconnect (potentially to a different instance). If you do not use this option, then the sessions that are connected to a database using this service remain connected, but new sessions cannot be established to the service. |
| <code>-gds pool <i>gds pool_name</i></code> | Specify the name of the GDS pool in which the service that you want to stop is located. If not specified and there is only one GDS pool with access granted to user, that GDS pool is used as the default GDS pool. |
| <code>-override</code> | This option causes the command to execute without updating the global service manager catalog. You can use this option when the catalog database is unavailable. During normal operation, you should not use this option. |
| <code>-pwd <i>password</i></code> | Specify the password of the GSMUSER in the specified database. |
| <code>-service <i>service_name</i></code> | Specify a comma-delimited list of global service names you want to stop. If you do not use this option, then GDSCTL stops all the services in the database pool. |
| <code>-stop_option</code> | Set the default stop option to NONE, IMMEDIATE, or TRANSACTIONAL |

Usage Notes

You must connect to the catalog database as a user with the pool administrator privileges, using the command [connect](#) before running this command.

If `-service` is not specified, all global services of GDS pool are stopped.

If `-database` is not specified, the global services are stopped on all of the databases.

If `-force` is specified, all sessions are disconnected, requiring the session using the global service to reconnect (potentially to another instance). If `-force` is not specified, then sessions already connected to this global service stay connected, but new sessions cannot be established to the global service.

If `-override` is specified, the command is executed without connecting to the GDS catalog. Use this option when the GDS catalog is unavailable. It is not recommended for use under normal operation.

Example

Stop the service SALES_REPORT, on all databases in the database pool READERFARM.

```
GDSCTL> stop service -gds pool readerfarm -service sales_report
```

① **See Also**

[#unique_143](#)

suspend services

This command allows users to block database on all GSM listeners.

This command allows users to block database on all GSM listeners. The purpose of this command is to isolate database that runs global services, but can not be accessed by applications.

Syntax

```
GDSCTL> suspend services -database target_db
```

Options

Table 2-114 GDSCTL suspend services Options

| Option | Description |
|------------------|----------------------------------|
| <i>target_db</i> | Specify the name of the database |

Example

In this scenario GSM won't be able to do failover, hence we need to block this database from GSM to initiate failover and guarantee that traffic won't be redirected to this database.

```
GDSCTL> suspend services -database target_db dbl
```

```
GDSCTL>
```

switchover ru

Switch leadership for the given replication unit to the specified database.

Syntax

```
switchover [ru|replication_unit] {-ru ru_id -database target_db | -rebalance}
[-timeout time]
```

Options

Table 2-115 GDSCTL switchover ru Options

| Option | Description |
|------------|---|
| -database | The name of the database. |
| -rebalance | Perform rebalancing of replication units across shards. |

Table 2-115 (Cont.) GDSCTL switchover ru Options

| Option | Description |
|----------|--|
| -ru | Replication unit ID |
| -timeout | Timeout of connection retention between FAN is sent to clients and chunk going read-only/down (seconds). |

Usage Notes

If the `REBALANCE` option is specified, an operation of rebalancing replication units and leadership responsibilities is distributed equally across the shards.

Examples

```
GDSCTL> switchover ru -ru 1 -database dba
```

sync brokerconfig (synchronize brokerconfig)

Synchronizes the Oracle Data Guard broker configuration in the global service manager with the configuration in the database pool. The `synchronize brokerconfig` command has the same options and usage.

Syntax

```
sync[hronize] brokerconfig [-gdspool gdspool_name]  
                           [-database db_name]
```

Options**Table 2-116 GDSCTL sync brokerconfig Options**

| Option | Description |
|------------------------------|---|
| -database <i>db_name</i> | Specify the name of a database in the database pool to use as a referential database , from which the configuration is queried. If you do not use this option, then GDSCTL uses the primary database as the referential database. If a primary database does not exist in the Oracle Data Guard broker configuration, then GDSCTL uses a random database from the pool as the referential database. |
| -gdspool <i>gdspool_name</i> | Specify the database pool to which the Oracle Data Guard broker configuration belongs. If not specified and there is only one <code>gdspool</code> with access granted to user, that <code>gdspool</code> is used as the default <code>gdspool</code> . If the specified database pool does not contain an Oracle Data Guard broker configuration, then GDSCTL returns an error. |

Usage Notes

You must connect to the catalog database as a user with the pool administrator privileges, using the command [connect](#) before running this command.

Example

Synchronize the Oracle Data Guard broker configuration in the database pool MYREADERFARM with the configuration stored in the Global Data Services catalog.

```
GDSCTL> sync brokerconfig -gdspool myreaderfarm
```

sync database (synchronize database)

Synchronizes attributes of global services and GDS related parameters of a GDS pool database with the contents of the GDS catalog. The `synchronize database` command has the same options and usage.

Syntax

```
sync[hronize] database [-gdspool gdspool_name]  
                        [-database database_name]
```

Options

Table 2-117 GDSCTL sync database Options

| Option | Description |
|---|--|
| <code>-database <i>database_name</i></code> | Specify the name of a database in the database pool to use as a <i>referential database</i> , from which the configuration is queried. |
| <code>-gdspool <i>gdspool_name</i></code> | Specify the GDS pool to which the database belongs. If not specified and there is only one GDS pool with access granted to user, it is used as the default GDS pool. |

Usage Notes

- If database has no GDS region assigned, an error is returned.
- If a GDS pool is specified and the database option is not specified, then each database in the pool is synchronized.
- Note that the GDS `sync database` command has the potential to cause services to restart.

Example

Synchronize a database in the default database pool with the database `mydb`.

```
GDSCTL> sync database -database mydb
```

sync ru

Synchronizes data of the specified replication unit on all shards, erases RAFT logs, and resets log index and term.

Syntax

```
sync[hronize] ru -ru ru_id [-database db]
```

Options

Table 2-118 GDSCTL sync ru Options

| Option | Description |
|--|---|
| <code>-connect connect_identifier</code> | Specify an Oracle Net connect descriptor or net service name that resolves to a connect descriptor for the database being added as the shard. |
| <code>-pwd password</code> | Enter the GSMUSER password. If not specified, the user is prompted for the password. |

Usage Notes

If a database is not specified for the `SYNC RU` command, a replica to synchronize with will be chosen based on the following criteria:

1. Pick the replica that was the the last leader.
2. If not available, pick the replica with greatest apply index.

Examples

```
GDSCTL> sync ru -ru 1 -database mydb1
```

sync schema (synchronize schema)

Allows common shared schemas across the existing databases to be retrieved. The command compares the schemas on all of the databases and retrieves those that are common.

Syntax

```
sync[hronize] schema [-schema schemalist [-retrieve_only] [-restart [-force]]  
| -apply [-skip_first] | -show [[-ddl ddlnum] [-count n] | [-failed_only]]]
```

Options

Table 2-119 GDSCTL sync schema Options

| Option | Description |
|---------------------|--|
| <code>-apply</code> | Specifies that the previously retrieved DDLs should be run in the catalog. |

Table 2-119 (Cont.) GDSCTL sync schema Options

| Option | Description |
|---------------------------|--|
| -count <i>n</i> | Specifies the maximum number of entries to show. |
| -ddl <i>ddlnum</i> | Specifies the DDL numeric identifier. |
| -failed_only | Shows only errored out statements. |
| -force | Forces sync without user confirmation. |
| -restart | Sync from the beginning, erasing schemas synced earlier. |
| -retrieve_only | Specifies that the DDLs of the common schemas should be retrieved only from the databases and stored in the catalog but not applied. |
| -schema <i>schemalist</i> | Specifies that only the listed schemas will be retrieved. Specify <i>all</i> to include all non-Oracle schemas. |
| -show | Shows DDL statements and their execution status. |
| -skip_first | Specifies that the first failed DDL statement is skipped. |

Usage Notes

This command is used only when the catalog is created for a federated database, which can be created by using option `-for_federated_database` of the `create shardcatalog` command. This option is mutually exclusive with `-sharding` parameter. The rest of the steps are similar to sharded database environment setup with user-defined sharding [`create shardcatalog`, `add gsm`, `add shardspace`, `add shard`, `deploy`]. After deployment is complete, the `sync schema` command can be run to import specified schemas from shards to the catalog.

The `sync ddl` command combines two operations:

1. Importing and applying schemas on the catalog.
2. Viewing the DDLs generated by combining schemas from shards.

The first operation is the default behavior and it requires a mandatory `-schema` parameter, which is list of schemas to import from shards. Note that `all` can be supplied to the `-schema` parameter to retrieve all non-Oracle schemas common to all shards and which do not exist on the catalog. This operation can be split into two steps using `-retrieve_only` and `-apply` options. The option `-retrieve_only` will retrieve schemas from the shards and generate the required DDL statements to be applied, but it does not execute these statements. To execute them at a later point, the `-apply` option is used. If, for some reason, a DDL execution fails, subsequent statements will not be executed as there could be dependencies on the failed DDL. When `-apply` is run again after fixing the issue, it will start from the first failed DDL statement and continue execution.

The second operation, `-show` is for examining DDL statements and their execution status.

Example

```
GDSCTL> sync schema -schema myschema
```

```
GDSCTL> sync schema -schema foo,bar
```

```
GDSCTL> sync schema -schema foo,"Bar"
```

```
GDSCTL> sync schema -schema all
```

validate backup

The validate backup command provides sharded database (SDB) backup validation.

Syntax

```
validate backup [-restorepoint restore_point_name] [-controlfile] [-header]
                [-catpwd password] [-shard shard_list] [-async]
```

Options

Table 2-120 GDSCTL validate backup Options

| Option | Description |
|---|--|
| -restorepoint <i>restore_point_name</i> | A restore point to verify the backups against. |
| -controlfile | If specified, only backups usable to restore database control files to a specific restore point are validated. |
| -header | If specified, it will only validate the backup file headers to determine whether the files on the media correspond to the metadata in the RMAN repository. |
| -catpwd <i>password</i> | Password for user GSMCATUSER. Prompted if not specified. This password only needs to be specified once for this command in an entire GDSCTL session. |
| -shard <i>shard_list</i> | <i>shard_list</i> specifies a comma separated list of shard identifiers. They can be shard space, shard group or shard names. The default is all shards. |
| -async | When specified, all tasks to configure the backup for the shards will run in background. By default, the task will run in foreground. The task for the SDB catalog database will always run in foreground regardless of this flag setting. |

Examples

The sharded database catalog (SC) database must be open, but the shard databases can be either mounted or open. If the backup validation is for database control files, the shards can be started nomount. The following example validates the backups of the control files from the SDB catalog database recoverable to restore point backup_before_db_maintenance.

```
GDSCTL> validate backup -shard catalog -controlfile -restorepoint
BACKUP_BEFORE_DB_MAINTENANCE
```

```
Validating backups for database "v1908" ...
executing command: SET until clause
```

```
Starting restore at 14-APR-20
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=201 device type=DISK
channel ORA_DISK_1: starting validation of datafile backup set
channel ORA_DISK_1: reading from backup piece /ade/b/3998875997/oracle/dbs/
V1908/autobackup/2020_04_14/o1_mf_s_1037669374_h9blkyc8_.bkp
channel ORA_DISK_1: piece handle=/ade/b/3998875997/oracle/dbs/V1908/
autobackup/2020_04_14/o1_mf_s_1037669374_h9blkyc8_.bkp tag=TAG20200414T012934
channel ORA_DISK_1: restored backup piece 1
```

```
channel ORA_DISK_1: validation complete, elapsed time: 00:00:01
Finished restore at 14-APR-20
Recovery Manager complete.
```

The next example validates the headers of the backups from shard v1908b_cdb2_pdb1 recoverable to restore point backup_before_db_maintenance.

```
GGDSCTL> validate backup -shard v1908b_cdb2_pdb1 -restorepoint
BACKUP_BEFORE_DB_MAINTENANCE -header
```

```
Validating backups for database "v1908b_cdb2_pdb1" ...
executing command: SET until clause
```

```
Starting restore at 14-APR-20
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=468 device type=DISK
allocated channel: ORA_DISK_2
channel ORA_DISK_2: SID=236 device type=DISK
```

List of Backup Sets

```
=====
```

| BS Key | Type | LV | Size | Device | Type | Elapsed Time | Completion Time |
|--------|------|----|---------|--------|------|--------------|-----------------|
| 2998 | Incr | 0 | 265.53M | DISK | | 00:00:06 | 14-APR-20 |

```

BP Key: 3009 Status: AVAILABLE Compressed: NO Tag:
BACKUP_BEFORE_DB_MAINTENANCE
Piece Name: /tmp/rman/backups/2/0sutl6oa
List of Datafiles in backup set 2998
Container ID: 3, PDB Name: CDB2_PDB1
File LV Type Ckp SCN Ckp Time Abs Fuz SCN Sparse Name
-----
11 0 Incr 2678401 14-APR-20 NO /ade/b/3998875997/
oracle/dbs/cdb2_pdb1_db.f

BS Key Type LV Size Device Type Elapsed Time Completion Time
-----
2999 Incr 0 191.61M DISK 00:00:04 14-APR-20
BP Key: 3010 Status: AVAILABLE Compressed: NO Tag:
BACKUP_BEFORE_DB_MAINTENANCE
Piece Name: /tmp/rman/backups/1/0tutl6oh
List of Datafiles in backup set 2999
Container ID: 3, PDB Name: CDB2_PDB1
File LV Type Ckp SCN Ckp Time Abs Fuz SCN Sparse Name
-----
12 0 Incr 2678425 14-APR-20 NO /ade/b/3998875997/
oracle/dbs/cdb2_pdb1_ax.f
13 0 Incr 2678425 14-APR-20 NO /ade/b/3998875997/
oracle/dbs/cdb2_pdb1_xdb.f
validation succeeded for backup piece
Finished restore at 14-APR-20
```

```
Recovery Manager complete.
```

validate catalog

Cross checks the Global Data Services catalog, global service manager run-time status, and pool databases, and reports inconsistencies and errors.

Syntax

```
validate [catalog]
        [-gsm gsm_name]
        [ {-config | -database db_name} ]
        [-catpwd cpwd]
        [-dbpwd dpwd]
```

Options

Table 2-121 GDSCTL validate catalog Options

| Option | Description |
|--------------------------|--|
| -catpwd <i>cpwd</i> | Provides the GSMCATUSER password, otherwise it is read from the local wallet file by default. |
| -config | Indicates that the validation should be performed on the Global Data Services catalog configuration only. |
| -database <i>db_name</i> | Indicates the name of the database for which the cross-check validation should be performed. |
| -dbpwd <i>dpwd</i> | Provides the pool database password directly if there is only one database in the pool, or if multiple databases in the pool share the same password. |
| -gsm <i>gsm_name</i> | Specify the global service manager name. If the name is not specified, then GDSCTL uses the current global service manager name for the session (specified with the command set gsm). |

Usage Notes

Because the execution of this command involves accessing all databases in a Global Data Services configuration, the GSMCATUSER password is required run it. The password is stored in the wallet of any global service manager that is part of the Global Data Services configuration. Therefore, if you run the command from the ORACLE_HOME of any of the global service managers, the password is automatically extracted from the wallet and does not have to be provided. Otherwise, you must provide the GSMCATUSER password using the -catpwd command option. Alternatively, if all databases in the Global Data Services configuration have the same GSMUSER password, you can specify the password instead of the GSMCATUSER password by using the -dbpwd option.

Example

Validate the catalog:

```
GDSCTL> validate
```

The output should be similar to the following:

```
Validation results:
VLD2: Region "regionora" does not have buddy region
VLD11: GDS pool "marketing" does not contain any databases
VLD12: GDS pool "marketing" does not contain any global services
VLD11: GDS pool "sales" does not contain any databases
VLD12: GDS pool "sales" does not contain any global services
VLD11: GDS pool "mkt" does not contain any databases
VLD12: GDS pool "mkt" does not contain any global services
```

validate

Cross checks the GDS catalog, global service manager run-time status, and databases from the GDS pool and reports any inconsistencies and errors.

Syntax

```
validate [catalog] [-gsm gsm]
                    [-config | -database db_name [-dbpwd sipwd]]
                    [-catpwd cpwd]
                    [-validate_network]
                    [-show_errors]
                    [-validate_network]
```

Options

Table 2-122 GDSCTL validate Options

| Option | Description |
|--------------------------|---|
| -catpwd <i>cpwd</i> | GSMCATUSER password. |
| -config | If specified, performs validation of GDS catalog configuration only. |
| -database <i>db_name</i> | Performs cross-check validation of the specified database. |
| -dbpwd <i>sipwd</i> | GSMUSER password. |
| -gsm <i>gsm</i> | Global service manager name |
| -validate_network | This flag enables several network validation checks, including checking network connectivity between hosts and checking VNCR entries are valid. |
| -show_errors | Show errors only. |

Usage Notes

If no options are specified, cross-checks are performed on the GDS catalog, database, and local global service manager.

Example

```
GDSCTL> validate catalog -catpwd cpwd -dbpwd sipwd
```

3

GDSCTL Commands Used For Oracle Globally Distributed AI Database

A subset of GDSCTL commands are applicable to an Oracle Globally Distributed AI Database configuration.

The following GDSCTL commands are commonly used in an Oracle Globally Distributed AI Database configuration:

- [add cdb](#)
- [add credential](#)
- [add file](#)
- [add gsm](#)
- [add invitednode \(add invitedsubnet\)](#)
- [add region](#)
- [add service](#)
- [add shard](#)
- [add shardgroup](#)
- [add shardspace](#)
- [alter move](#)
- [alter move](#)
- [alter task](#)
- [config](#)
- [config backup](#)
- [config cdb](#)
- [config chunks](#)
- [config credential](#)
- [config file](#)
- [config gsm](#)
- [config region](#)
- [config sdb](#)
- [config service](#)
- [config shard](#)
- [config shardgroup](#)
- [config shardspace](#)
- [config table family](#)
- [config task](#)

- [config vncr](#)
- [configure](#)
- [connect](#)
- [copy ru](#)
- [create restorepoint](#)
- [create shardcatalog](#)
- [delete backup](#)
- [delete catalog](#)
- [deploy](#)
- [disable backup](#)
- [disable service](#)
- [enable backup](#)
- [enable service](#)
- [list backup](#)
- [list restorepoint](#)
- [modify catalog](#)
- [modify cdb](#)
- [modify credential](#)
- [modify file](#)
- [modify gsm](#)
- [modify region](#)
- [modify service](#)
- [modify shard](#)
- [modify shardgroup](#)
- [modify shardspace](#)
- [move chunk](#)
- [move ru](#)
- [relocate chunk](#)
- [relocate service](#)
- [remove cdb](#)
- [remove credential](#)
- [remove file](#)
- [remove gsm](#)
- [remove invitednode \(remove invitedsubnet\)](#)
- [remove region](#)
- [remove ru](#)
- [remove service](#)
- [remove shard](#)

- [remove shardgroup](#)
- [remove shardspace](#)
- [restore backup](#)
- [services](#)
- [set gsm](#)
- [set inbound_connect_timeout](#)
- [set log_status](#)
- [set outbound_connect_timeout](#)
- [set trace_level](#)
- [split chunk](#)
- [sql](#)
- [start gsm](#)
- [start ru](#)
- [start service](#)
- [status backup](#)
- [status gsm](#)
- [status ru](#)
- [status service](#)
- [stop gsm](#)
- [stop ru](#)
- [stop service](#)
- [switchover ru](#)
- [sync ru](#)
- [sync schema \(synchronize schema\)](#)
- [validate backup](#)
- [validate catalog](#)

4

GDSCTL Commands Used For Global Data Services

A subset of GDSCTL commands are applicable to a Global Data Services (GDS) configuration.

The following GDSCTL commands are commonly used in a GDS configuration:

- [add brokerconfig](#)
- [add database](#)
- [add gdspool](#)
- [add gsm](#)
- [add invitednode \(add invitedsubnet\)](#)
- [add region](#)
- [add service](#)
- [config](#)
- [config database](#)
- [config gdspool](#)
- [config gsm](#)
- [config region](#)
- [config service](#)
- [config vncr](#)
- [configure](#)
- [connect](#)
- [create gdscatalog](#)
- [delete catalog](#)
- [disable service](#)
- [enable service](#)
- [export catalog](#)
- [import catalog](#)
- [modify catalog](#)
- [modify database](#)
- [modify gdspool](#)
- [modify gsm](#)
- [modify region](#)
- [modify service](#)
- [relocate service](#)

- [remove brokerconfig](#)
- [remove database](#)
- [remove gdspool](#)
- [remove gsm](#)
- [remove invitednode \(remove invitedsubnet\)](#)
- [remove region](#)
- [remove service](#)
- [services](#)
- [set gsm](#)
- [set inbound_connect_timeout](#)
- [set log_status](#)
- [set outbound_connect_timeout](#)
- [set trace_level](#)
- [start gsm](#)
- [start service](#)
- [status database](#)
- [status gsm](#)
- [status service](#)
- [stop gsm](#)
- [stop service](#)
- [sync brokerconfig \(synchronize brokerconfig\)](#)
- [sync database \(synchronize database\)](#)
- [sync schema \(synchronize schema\)](#)
- [validate catalog](#)

Glossary

Index

A

add cdb, [9](#)

D

databases, [61](#)

G

GDSCTL

commands

- add brokerconfig, [7](#)
- add cdb, [9](#)
- add credential, [10](#)
- add database, [10](#)
- add file, [12](#)
- add gdspool, [13](#)
- add gsm, [14](#)
- add invitednode (add invitedsubnet), [16](#)
- add region, [17](#)
- add service, [18](#)
- add shard, [25](#)
- add shardgroup, [28](#)
- add shardspace, [29](#)
- alter move, [30](#)
- alter task, [32](#)
- config, [33](#)
- config backup, [34](#)
- config cdb, [36](#)
- config chunks, [37](#)
- config credential, [38](#)
- config database, [39](#)
- config file, [40](#)
- config gdspool, [41](#)
- config gsm, [42](#)
- config sdb, [43](#)
- config service, [44](#)
- config shard, [46](#)
- config shardgroup, [47](#)
- config shardspace, [48](#)
- config table family, [49](#)
- config task, [49](#)
- config vncr, [50](#)
- connect, [52](#)
- copy ru, [53](#)

GDSCTL (continued)

commands (continued)

- create catalog, [54](#)
- create restorepoint, [56](#)
- create shardcatalog, [57](#)
- databases, [61](#), [128](#)
- delete backup, [62](#)
- delete catalog, [63](#)
- deploy, [64](#)
- disable backup, [65](#)
- disable service, [65](#)
- enable backup, [66](#)
- enable service, [67](#)
- exit, [68](#), [93](#)
- help, [69](#)
- list backup, [70](#)
- list restorepoint, [72](#)
- modify catalog, [73](#)
- modify cdb, [75](#)
- modify credential, [76](#)
- modify database, [77](#)
- modify file, [78](#)
- modify gdspool, [78](#)
- modify gsm, [79](#)
- modify region, [81](#)
- modify service, [81](#)
- modify shard, [89](#)
- modify shardgroup, [90](#)
- modify shardspace, [91](#)
- move chunk, [92](#)
- move ru, [93](#)
- quit, [68](#), [93](#)
- recover shard, [93](#)
- relocate chunk, [94](#)
- relocate service, [95](#)
- remove brokerconfig, [96](#)
- remove cdb, [97](#)
- remove credential, [98](#)
- remove database, [99](#)
- remove file, [100](#)
- remove gdspool, [100](#)
- remove gsm, [101](#)
- remove region, [102](#)
- remove ru, [103](#)
- remove service, [104](#)
- remove shard, [104](#)

GDSCTL (*continued*)

- commands (*continued*)
- remove shardgroup, [105](#)
- remove shardspace, [106](#)
- remove vncr, [102](#)
- restore backup, [106](#)
- resume services, [109](#)
- rman, [109](#)
- run backup, [111](#)
- services, [112](#)
- set gsm, [113](#), [114](#)
- set inbound_connect_timeout, [115](#)
- set log_status, [116](#)
- set outbound_connect_timeout, [117](#)
- set trace_level, [117](#)
- set trc_level, [118](#)
- show ddl, [119](#)
- split chunk, [120](#)
- sql, [121](#)
- start gsm, [122](#)
- start observer, [122](#)
- start ru, [123](#)
- start service, [124](#)
- status, [125](#), [129](#)
- status backup, [126](#)
- status database, [128](#)
- status gsm, [129](#)
- status routing, [130](#)
- status ru, [131](#)
- status service, [109](#), [112](#), [130](#), [134](#), [138](#)
- stop gsm, [135](#)
- stop ru, [135](#)
- stop service, [136](#)
- suspend services, [138](#)
- switchover ru, [138](#)
- sync brokerconfig, [139](#)
- sync database, [140](#)
- sync ru, [141](#)
- sync schema, [141](#)
- synchronize brokerconfig, [139](#)
- synchronize database, [140](#)
- synchronize schema, [141](#)
- validate, [146](#)
- validate backup, [143](#)

GDSCTL (*continued*)

- commands (*continued*)
- validate catalog, [145](#)
- for Global Data Services, [1](#)
- for Oracle Globally Distributed Database, [1](#)
- reference, [1](#)

M

- modify gdspool, [78](#)
- modify gsm, [79](#)
- modify region, [81](#)
- modify service, [81](#)
- modify shardgroup, [90](#)

R

- remove brokerconfig, [96](#)
- remove database, [99](#)
- remove gdspool, [100](#)
- remove gsm, [101](#)
- remove region, [102](#)
- remove service, [104](#)
- remove vncr, [102](#)

S

- services
 - managing, [81](#)
- set dataguard_property, [113](#)
- set gsm, [114](#)
- start gsm, [122](#)
- start service, [124](#)
- status database, [128](#)
- status service, [134](#)
- stop gsm, [135](#)
- sync database, [140](#)
- sync schema, [141](#)

V

- validate catalog, [145](#)