

Oracle Maximum  
Availability Architecture

An Oracle White Paper  
March 2014

## Oracle Data Guard Broker

# Best Practices for Configuring Redo Transport for Data Guard and Active Data Guard 12c

Introduction .....	1
Broker Redo Transport Configuration Properties .....	3
Planning for Redo Transport Changes.....	4
Example 1: Primary Database and Standby Database .....	5
Example 2: A Basic Active Data Guard Far Sync Configuration.....	6
Example 3: A More Advanced Far Sync Configuration .....	8
Example 4: A Basic Cascaded Physical Standby Configuration.....	9
Example 5: A More Advanced Cascaded Configuration .....	11
Conclusion .....	12

## Introduction

Oracle Data Guard broker with Oracle Database 12c introduces several new features that automate management and configuration of sophisticated high availability and disaster recovery environments:

- Cascaded redo transport (including Active Data Guard Real-Time Cascade) is supported for broker managed physical standby configurations.
- Active Data Guard Far Sync can be used to offload redo transport processing from the primary database and enable zero-data-loss failovers to remote standby databases situated far beyond typical `SYNC` transport distances.
- A new Data Guard broker property, `RedoRoutes`, is provided to facilitate redo transport configuration for advanced environments.

This paper provides examples that illustrate Data Guard broker DGMGRL commands and best practices used to manage and configure a variety of Active Data Guard configurations that make use of these new features.

When initially created, a Data Guard broker configuration defaults to the Maximum Performance data protection mode and the primary database sends redo to each configuration member using `ASYNC` transport. The examples that follow begin with this initial default configuration and show the DGMGRL commands required to convert to the no-data-loss Maximum Availability data protection mode. Several of the examples incorporate Active Data Guard Far Sync or Active Data Guard real-time cascading (real-time cascade enables a standby database to forward redo directly to other standby databases as soon as it is received, using asynchronous redo transport). For all examples, the Data Guard broker properties are configured to ensure the desired redo transport method is utilized regardless of which database functions in the primary role following a Data Guard role transition (switchover or failover). To illustrate how the broker uses the specified property values to automatically

reconfigure redo transport after a role transition, some examples also include configuration displays after one or more switchovers.

Content from the Oracle Data Guard 12c documentation has been included in portions of this paper to provide relevant background or technical information. For additional information about how to configure and manage Oracle Data Guard broker configurations and how to initially create and configure standby databases and far sync instances, refer to the Oracle Data Guard 12c documentation available through the Oracle Technology Network.<sup>1</sup>

Note that additional configuration steps might be required for a production deployment. In addition to the Oracle Data Guard documentation, the Oracle Maximum Availability Architecture (MAA) provides additional best practices for deploying and maintaining Oracle database high availability and disaster recovery environments.<sup>2</sup>

NOTE: Use of Active Data Guard Far Sync or Real-Time Cascade requires that the primary and standby databases be licensed for Active Data Guard.

---

<sup>1</sup> Data Guard documentation: <http://www.oracle.com/technetwork/database/features/availability/data-guard-documentation-152848.html>

<sup>2</sup> Oracle MAA Best Practices: <http://www.oracle.com/technetwork/database/features/availability/maa-096107.html>

## Broker Redo Transport Configuration Properties

Redo transport services are managed by setting appropriate values for one or more Data Guard broker configuration properties. The most commonly used of these properties are:

- `DGConnectIdentifier`
- `LogXptMode`
- `NetTimeout`
- `RedoCompression`
- `RedoRoutes` (new in Oracle Database 12c)
- `ReopenSecs`

The actual redo transport configuration, which is accomplished by setting appropriate values for `LOG_ARCHIVE_DEST_n` initialization parameters, is carried out by the broker on the each configuration member that ships redo to other configuration members. If changing a broker redo transport property requires changes to any `LOG_ARCHIVE_DEST_n` initialization parameter attributes, the broker forces a log switch on each primary database thread so that the new setting is adopted immediately. To facilitate future role transitions, the `RedoRoutes` property should be preconfigured for all members of a Data Guard broker configuration.

For very simple configurations, the default values for these properties are appropriate. The examples in this paper show how to assign appropriate values to the `RedoRoutes` property for a variety of typical configurations. For information on how to assign values to other broker redo transport properties refer to the Data Guard broker documentation.

For Data Guard broker with Oracle Database 11g, the redo transport mode (`SYNC` or `ASYNC`) is specified by setting the `LogXptMode` property on the redo recipient (standby database). The broker then uses this information to configure a `LOG_ARCHIVE_DEST_n` initialization parameter on the redo source (primary database) that includes the specified redo transport mode.

Oracle Data Guard broker with Oracle Database release 12c adds `FASTSYNC` to the already supported `SYNC` and `ASYNC` redo transport modes and allows these modes to be specified in the `RedoRoutes` property value to facilitate redo transport configuration for complex configurations. The `RedoRoutes` property is set on any configuration member that might function in a role that ships redo to other configuration members (primary database, cascading physical standby, or far sync instance). The broker uses the `RedoRoutes` property values to configure `LOG_ARCHIVE_DEST_n` initialization parameters on each of these configuration members that include the specified destinations redo transport modes. Although it is possible to set values for both `LogXptMode` and `RedoRoutes`, it is recommended, for ease in ensuring that the intended log transport mode is implemented, that only one of these two properties be used exclusively within any given configuration. Because the `RedoRoutes` property allows the redo transport mode to vary based upon which database is currently in the primary role, it provides greater control over how redo transport is configured and hence is the preferred way to configure redo transport when using Data Guard broker 12c. Note that a redo

transport mode that is specified in the `RedoRoutes` property overrides the value specified for the `LogXptMode` property.

The valid values for `RedoRoutes` are strings consisting of one or more redo routing rules, each rule enclosed within a pair of parentheses, of the form `(redo_source_1 : redo_destination_1) [ , (redo_source_n : redo_destination_n) ]`. The `redo_source` field of a redo routing rule must contain the keyword `LOCAL` (an alias for the local database name) or a comma-separated list of `DB_UNIQUE_NAME` values. The `LOCAL` keyword cannot be used at a far sync instance and a database cannot be specified as a redo source in more than one redo routing rule defined at a given database, either explicitly or implicitly through use of the `LOCAL` keyword. The `redo_destination` field must contain the keyword `ALL` (an alias for all possible destinations in the configuration) or a comma-separated list of `DB_UNIQUE_NAME` values, each of which can be followed by an optional redo transport mode attribute (`ASYNC`, `SYNC`, or `FASTSYNC`). If the optional redo transport attribute is not specified, then the redo transport mode used will be the one specified by the `LogXptMode` property for the destination (which defaults to `ASYNC`). The following additional usage notes apply:

- The `RedoRoutes` property has a default value of `NULL`, which is treated as `(LOCAL : ALL ASYNC)`. For the primary database, this means that it will ship redo data to every member of the configuration.
- A redo routing rule is active if its redo source field includes the current primary database. If a rule is active, primary database redo is sent by the database at which the rule is defined to each destination specified in the redo destination field of the active rule.
- After a role transition, the broker automatically reconfigures the `LOG_ARCHIVE_DEST_n` initialization parameters for each configuration member to conform to the active redo routing rules.
- The `ASYNC` redo transport attribute must be explicitly specified for a cascaded destination to enable real-time cascading to that destination.
- The `RedoRoutes` property cannot be set on a logical (SQL Apply) or snapshot standby database.
- When performing a database rolling upgrade using a transient logical standby, any necessary updates to database log archive destination parameters must be performed manually because the broker is temporarily disabled during this process.

NOTE: The examples make use of the `LOCAL` and `ALL` keywords where appropriate; there may be alternative ways to achieve the same redo transport configuration, for example, by using specific database `DB_UNIQUE_NAME` values instead of these keywords.

## Planning for Redo Transport Changes

Altering the values of the various Data Guard broker redo transport related configuration properties can temporarily disrupt the flow of redo from the primary database to the other configuration members. The same also may be true immediately after performing a role transition. For each of these

cases, transient warning and error messages may appear in the configuration display during the interval while the old redo transport is being dismantled and the new redo transport is becoming fully active.

For example, in the examples that follow, immediately after setting new values for the broker RedoRoutes property or immediately after performing a switchover, the broker configuration status may briefly include one or more of the following messages:

```
ORA-16629: database reports a different protection level from the protection mode
ORA-16687: this database receives redo data from multiple sources
ORA-16778: redo transport error for one or more databases
ORA-16857: standby disconnected from redo source for longer than specified threshold
ORA-16858: last communication time from redo source could not be determined
```

If any of these messages persist, verify that redo transport has been configured correctly. Also, be sure standby redo logs are correctly configured for all configuration members so that redo from the primary database can be shipped to and applied on all standby databases in as close to real time as possible.

The occurrence and duration of any transient disruption while reconfiguring redo transport can be minimized by planning in advance the specific sequence of commands required to complete the desired changes. When working out the sequence in which the redo transport changes will be implemented, note that it is preferable for a downstream target to temporarily receive redo from multiple sources than to not receive any redo at all while changes are being made.

## Example 1: Primary Database and Standby Database

This is the most basic Data Guard configuration consisting of a primary database db01 and a physical standby database db02. For purposes of the Maximum Availability protection mode example, it is assumed that the two databases are situated such that SYNC redo transport between the local and remote sites is feasible.

```
DGMGRL> show configuration

Configuration - config

Protection Mode: MaxPerformance
Databases:
db01 - Primary database
db02 - Physical standby database

Fast-Start Failover: DISABLED

Configuration Status:
SUCCESS

DGMGRL> edit database db01 set property redoroutes='(LOCAL : db02 SYNC)';
Property "redoroutes" updated
DGMGRL> edit database db02 set property redoroutes='(LOCAL : db01 SYNC)';
Property "redoroutes" updated
DGMGRL> edit configuration set protection mode as maxavailability;
Succeeded.
DGMGRL> show configuration;

Configuration - config
```

```
Protection Mode: MaxAvailability
Databases:
db01 - Primary database
      db02 - Physical standby database

Fast-Start Failover: DISABLED

Configuration Status:
SUCCESS
```

This example is equivalent to setting the `LogXptMode` property to `SYNC` for each database in the configuration. Note that the remaining examples require setting values for the `RedoRoutes` property and cannot be configured using the `LogXptMode` property alone.

## Example 2: A Basic Active Data Guard Far Sync Configuration

Active Data Guard Far Sync uses a new kind of redo destination (a far sync instance) that accepts redo from the primary database and then ships that redo to other members of the Data Guard configuration. A far sync instance is comprised only of a control file, standby redo log files, and archive log files (no data files, no media recovery). Active Data Guard Far Sync enables zero data loss failover to a remote standby database located hundreds or thousands of miles away, without primary database performance being affected by WAN network latency. Far Sync when used in combination with the Oracle Advanced Compression Option also enables off-host redo transport compression, conserving WAN bandwidth without affecting primary database performance.

Because a far sync instance has no data files, it cannot assume the primary role. The primary and standby databases in this example are each located within the feasible `SYNC` redo transport range of the far sync instance but are themselves separated by more than this distance. The far sync instance `fs01` in this configuration makes a no-data-loss failover possible between databases `db01` and `db02` even though, with a single far sync, the databases can be located apart at up to twice the typical distance feasible for `SYNC` transport. Because the resource and management requirements for far sync instances are less than those for cascading physical standby databases, it may be possible to locate a far sync instance in an environment provisioned with fewer resources than a typical data center.

```
DGMGRL> show configuration;

Configuration - config

Protection Mode: MaxPerformance
Databases:
db01 - Primary database
      db02 - Physical standby database
      fs01 - Far Sync (inactive)

Fast-Start Failover: DISABLED

Configuration Status:
SUCCESS

DGMGRL> edit database db01 set property redoroutes='(LOCAL : fs01 SYNC)';
Property "redoroutes" updated
DGMGRL> edit database db02 set property redoroutes='(LOCAL : fs01 SYNC)';
Property "redoroutes" updated
```

```
DGMGRL> edit far_sync fs01 set property redoroutes='(LOCAL : db01, db02)';
Error: ORA-16689: illegal sender specified for RedoRoutes property
```

Failed.

NOTE: An attempt to use the LOCAL keyword in the RedoRoutes property value for a far sync instance will generate an error.

```
DGMGRL> edit far_sync fs01 set property redoroutes='(db01 : db02)(db02 :
db01)';
```

```
Property "redoroutes" updated
```

```
DGMGRL> edit configuration set protection mode as maxavailability;
```

```
Succeeded.
```

```
DGMGRL> show configuration;
```

```
Configuration - config
```

```
Protection Mode: MaxAvailability
Databases:
db01 - Primary database
fs01 - Far Sync
db02 - Physical standby database
```

```
Fast-Start Failover: DISABLED
```

```
Configuration Status:
```

```
SUCCESS
```

```
DGMGRL> switchover to fs01;
```

```
Performing switchover NOW, please wait...
```

```
Error: ORA-16831: operation disallowed on this standby database type
```

Failed.

Unable to switchover, primary database is still "db01"

NOTE: An attempt to perform a switchover to a far sync instance will generate an error.

```
DGMGRL> switchover to db02;
```

```
Performing switchover NOW, please wait...
```

```
New primary database "db02" is opening...
```

```
Operation requires shutdown of instance "db01" on database "db01"
```

```
Shutting down instance "db01"...
```

```
ORACLE instance shut down.
```

```
Operation requires startup of instance "db01" on database "db01"
```

```
Starting instance "db01"...
```

```
ORACLE instance started.
```

```
Database mounted.
```

```
Database opened.
```

```
Switchover succeeded, new primary is "db02"
```

```
DGMGRL> show configuration;
```

```
Configuration - config
```

```
Protection Mode: MaxAvailability
Databases:
db02 - Primary database
fs01 - Far Sync
```

```

db01 - Physical standby database

Fast-Start Failover: DISABLED

Configuration Status:
SUCCESS

```

### Example 3: A More Advanced Far Sync Configuration

This example illustrates the use of local far sync instances at both the local and remote locations to enable no-data-loss failovers in either direction between two widely separated databases. The initial primary database db01 is located within SYNC transport range of far sync instance fs01. The initial standby database db02 is located within SYNC transport range of far sync instance fs02. Databases db01 and db02 are widely separated (from hundreds to thousands of miles) beyond the normally feasible SYNC transport range. If there is a failure at the primary database site, the local far sync instance and remote standby coordinate a final redo shipment from the far sync instance to the standby to ensure any remaining redo not yet available to the standby database is received and applied. In this way, a successful no-data-loss failover to the standby is possible even when the standby is widely separated from the primary database. Note that the ability to perform a no-data-loss failover in the event of a primary database failure is enhanced when each far sync instance is not immediately co-located with the database from which it receives redo (so that a local area disaster has only a minimal chance of affecting both the database and its local far sync instance).

```

DGMGRL> show configuration;

Configuration - config

Protection Mode: MaxPerformance
Databases:
db01 - Primary database
db02 - Physical standby database
fs01 - Far Sync (inactive)
fs02 - Far Sync (inactive)

Fast-Start Failover: DISABLED

Configuration Status:
SUCCESS

DGMGRL> edit database db01 set property redoroutes='(LOCAL : fs01 SYNC)';
Property "redoroutes" updated
DGMGRL> edit database db02 set property redoroutes='(LOCAL : fs02 SYNC)';
Property "redoroutes" updated
DGMGRL> edit far_sync fs01 set property redoroutes='(db01 : db02)';
Property "redoroutes" updated
DGMGRL> edit far_sync fs02 set property redoroutes='(db02 : db01)';
Property "redoroutes" updated
DGMGRL> edit configuration set protection mode as maxavailability;
Succeeded.
DGMGRL> show configuration;

Configuration - config

Protection Mode: MaxAvailability
Databases:
db01 - Primary database

```

```
fs01 - Far Sync
db02 - Physical standby database
fs02 - Far Sync (inactive)

Fast-Start Failover: DISABLED

Configuration Status:
SUCCESS

DGMGRL> switchover to db02;
Performing switchover NOW, please wait...
New primary database "db02" is opening...
Operation requires shutdown of instance "db01" on database "db01"
Shutting down instance "db01"...
ORACLE instance shut down.
Operation requires startup of instance "db01" on database "db01"
Starting instance "db01"...
ORACLE instance started.
Database mounted.
Database opened.
Switchover succeeded, new primary is "db02"
DGMGRL> show configuration;

Configuration - config

Protection Mode: MaxAvailability
Databases:
db02 - Primary database
fs02 - Far Sync
db01 - Physical standby database
fs01 - Far Sync (inactive)

Fast-Start Failover: DISABLED

Configuration Status:
SUCCESS
```

## Example 4: A Basic Cascaded Physical Standby Configuration

This example parallels *Example 2* (with the far sync instance in that example being replaced by a cascading physical standby database). The three databases in this configuration are located such that the initial primary database db01 is located within SYNC transport range of physical standby database db02 and physical standby database db02 is also within SYNC transport range of physical standby database db03. The databases db01 and db03 are located too far apart to support SYNC transport directly between them, so a Maximum Availability configuration with real time cascading is implemented. To preserve the protection mode, if db02 is in the primary role, it must ship redo directly to db01 and db03 using SYNC transport. When db01 or db03 are in the primary role, db02 is configured to ship redo to the remaining database using real time cascading. In this case, the furthest downstream standby database receives redo via ASYNC transport (using Active Data Guard Real-Time Cascade with Oracle Database 12c).

```
DGMGRL> show configuration

Configuration - config

Protection Mode: MaxPerformance
Databases:
```

```
db01 - Primary database
db02 - Physical standby database
db03 - Physical standby database
```

```
Fast-Start Failover: DISABLED
```

```
Configuration Status:
SUCCESS
```

```
DGMGRL> edit database db01 set property redoroutes='(LOCAL : db02 SYNC)';
Property "redoroutes" updated
DGMGRL> edit database db02 set property redoroutes='(LOCAL : db01 SYNC, db03
SYNC)(db01 : db03 ASYNC)(db03 : db01 ASYNC)';
Property "redoroutes" updated
DGMGRL> edit database db03 set property redoroutes='(LOCAL : db02 SYNC)';
Property "redoroutes" updated
DGMGRL> edit configuration set protection mode as maxavailability;
Succeeded.
DGMGRL> show configuration;
```

```
Configuration - config
```

```
Protection Mode: MaxAvailability
Databases:
db01 - Primary database
db02 - Physical standby database
db03 - Physical standby database (receiving current redo)
```

```
Fast-Start Failover: DISABLED
```

```
Configuration Status:
SUCCESS
```

NOTE: The *receiving current redo* clause is used to indicate that Oracle Active Data Guard real-time cascading is enabled between databases db02 and db03.

```
DGMGRL> switchover to db03;
Performing switchover NOW, please wait...
New primary database "db03" is opening...
Operation requires shutdown of instance "db01" on database "db01"
Shutting down instance "db01"...
ORACLE instance shut down.
Operation requires startup of instance "db01" on database "db01"
Starting instance "db01"...
ORACLE instance started.
Database mounted.
Database opened.
Switchover succeeded, new primary is "db03"
DGMGRL> show configuration
```

```
Configuration - config
```

```
Protection Mode: MaxAvailability
Databases:
db03 - Primary database
db02 - Physical standby database
db01 - Physical standby database (receiving current redo)
```

```
Fast-Start Failover: DISABLED
```

```
Configuration Status:
SUCCESS
```

An advantage to using a cascading standby in this example (instead of a far sync) is that if the current primary database fails, the new primary is still able to maintain no-data-loss protection.

## Example 5: A More Advanced Cascaded Configuration

This example assumes two data centers each with two databases. The first data center hosts databases db01 and db02 and the second data center hosts databases db03 and db04. When db01 is in the primary role, it ships redo using SYNC transport to database db02, which in turn uses real-time cascading to ship redo to databases db03 and db04. When db02 is in the primary role, it ships redo using SYNC transport to database db01, which in turn uses real-time cascading to ship redo to databases db03 and db04. Redo transport is configured in a similar manner when a database in the second datacenter is in the primary role. In each case, the local standby provides HA, facilitates rolling hardware and software upgrades, and offloads redo shipping from the primary database, while the databases in the remote datacenter provide DR protection and enable faster reporting at the remote location.

```
DGMGRL> show configuration

Configuration - config

Protection Mode: MaxPerformance
Databases:
db01 - Primary database
db02 - Physical standby database
db03 - Physical standby database
db04 - Physical standby database

Fast-Start Failover: DISABLED

Configuration Status:
SUCCESS

DGMGRL> edit database db01 set property redoroutes='(LOCAL : db02 SYNC)(db02 :
db03 ASYNC, db04 ASYNC)';
Property "redoroutes" updated
DGMGRL> edit database db02 set property redoroutes='(LOCAL : db01 SYNC)(db01 :
db03 ASYNC, db04 ASYNC)';
Property "redoroutes" updated
DGMGRL> edit database db03 set property redoroutes='(LOCAL : db04 SYNC)(db04 :
db01 ASYNC, db02 ASYNC)';
Property "redoroutes" updated
DGMGRL> edit database db04 set property redoroutes='(LOCAL : db03 SYNC)(db03 :
db01 ASYNC, db02 ASYNC)';
Property "redoroutes" updated
DGMGRL> edit configuration set protection mode as maxavailability;
Succeeded.
DGMGRL> show configuration;

Configuration - config

Protection Mode: MaxAvailability
Databases:
db01 - Primary database
db02 - Physical standby database
db03 - Physical standby database (receiving current redo)
db04 - Physical standby database (receiving current redo)

Fast-Start Failover: DISABLED
```

```
Configuration Status:
SUCCESS

DGMGRL> switchover to db04
Performing switchover NOW, please wait...
New primary database "db04" is opening...
Operation requires shutdown of instance "db01" on database "db01"
Shutting down instance "db01"...
ORACLE instance shut down.
Operation requires startup of instance "db01" on database "db01"
Starting instance "db01"...
ORACLE instance started.
Database mounted.
Database opened.
Switchover succeeded, new primary is "db04"
DGMGRL> show configuration;

Configuration - config

Protection Mode: MaxAvailability
Databases:
db04 - Primary database
db03 - Physical standby database
db01 - Physical standby database (receiving current redo)
db02 - Physical standby database (receiving current redo)

Fast-Start Failover: DISABLED

Configuration Status:
SUCCESS
```

## Conclusion

The new Data Guard broker RedoRoutes property enables you to create sophisticated redo transport topologies that use Active Data Guard Far Sync or Real-Time Cascade such that remote destinations automatically use the desired redo transport method based upon whichever database is currently operating in the primary role. The new redo transport features with Oracle database 12c enable easy configuration and management of sophisticated redo transport topologies that were not possible in prior releases.



Best Practices for Configuring Redo Transport  
for Data Guard and Active Data Guard 12c  
February 2014

Author: Laurence Clarke  
Contributors: Larry Carpenter, Nitin Karkhanis,  
Robert McGuiirk, Joseph Meeks

Oracle Corporation  
World Headquarters  
500 Oracle Parkway  
Redwood Shores, CA 94065  
U.S.A.

Worldwide Inquiries:  
Phone: +1.650.506.7000  
Fax: +1.650.506.7200

[www.oracle.com](http://www.oracle.com)



Oracle is committed to developing practices and products that help protect the environment

Copyright © 2014, Oracle and/or its affiliates. All rights reserved. This document is provided for information purposes only and the contents hereof are subject to change without notice. This document is not warranted to be error-free, nor subject to any other warranties or conditions, whether expressed orally or implied in law, including implied warranties and conditions of merchantability or fitness for a particular purpose. We specifically disclaim any liability with respect to this document and no contractual obligations are formed either directly or indirectly by this document. This document may not be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without our prior written permission.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. UNIX is a registered trademark licensed through X/Open Company, Ltd. 1010

**Hardware and Software, Engineered to Work Together**